ibi™ WebFOCUS®

ibi™ WebFOCUS® HTML5 (JSCHART) リファレンス

バージョン 9.0.0 以降 | January 2024

Copyright © 2023. Cloud Software Group, Inc. All Rights Reserved.

目次

1. TIBCO WebFOCUS HTML5 グラフの概要 15

TIBCO WebFOCUS グラフの概要 15

グラフタイプ 17

HTML5 グラフの作成 18

TIBCO WebFOCUS GRAPH FILE コマンドの概要 28

TIBCO WebFOCUS HTML5 GRAPH リクエストの構造 28

GRAPH と TABLE の類似点 34

GRAPH と TABLE の相違点 35

サンプルファイルの生成 36

2. グラフタイプの理解 39

グラフのコンポーネント 39

グラフタイプの制御 40

LOOKGRAPH によるグラフタイプの制御 40

グラフタイプの注意点およびサンプルグラフ 48

ヒストグラム 48

パラボックス 49

パレートグラフ 51

極グラフ 52

レーダーグラフ 54

3D 面グラフ 55

滝型グラフ 57

ブレットグラフ 58

スパークライングラフ 59

3. HTML5 グラフの JSON プロパティの概要 61

TIBCO WebFOCUS HTML5 JSON 構文の基礎 61

色およびグラデーション 63

グラデーションの定義 66

SVG 色パターン 76

フォントの定義 83

Google フォントの使用.....84
数値のフォーマット設定.....86
数値フォーマットの自動設定.....86
JSON オブジェクトによる数値フォーマット設定.....88
フォーマット文字列による数値フォーマット設定.....91
関数による数値フォーマット設定.....97
ツールヒントおよびラベルの数値フォーマット制御.....98
文字列内での HTML コードの使用.....101
データ定義.....102
グラフのプロパティ値の設定.....102
ラベル値を返すコールバック関数の定義.....102
グラフテンプレートエンジンを使用したツールヒント、タイトル、データラベルのカスタマイズ.....109
グラフタイトル、サブタイトル、脚注での series_label マクロの使用.....116
メタデータ数値フォーマットのデータラベルおよびツールヒントへの適用.....118

4. TIBCO WebFOCUS グラフ構文.....123

グラフ属性構文の概念.....123
属性カテゴリへのフィールドの割り当て方法についての理解.....124
サーバ生成プロパティの上書き.....128
マトリックス配列での複数グラフの表示.....131
マトリックスマーカーグラフ.....132
データグリッドの生成.....134
バブルグラフまたは散布図への詳細レベルの追加.....136
色フィールドおよびサイズフィールドの追加.....138
スライダコントロールを使用した時間推移のアニメーション化.....143
ツールヒントへのフィールドの追加.....145
ドリルダウンリンク付き画像の見出しへの追加.....148
グラフのフィールド値への色のバインド.....150
グラフ属性構文のグラフタイプ.....153
グラフ属性構文を使用したグラフの例.....155

グラフ属性構文を使用した面グラフ....156
グラフ属性構文を使用した棒グラフ....157
グラフ属性構文を使用したボックスプロット....159
グラフ属性構文を使用したバブルグラフ....161
グラフ属性構文を使用したバブルマップ....162
グラフ属性構文を使用したコロプレスマップ....166
グラフ属性構文を使用したデータグリッド....168
グラフ属性構文を使用したじょうごグラフ....170
グラフ属性構文を使用したメータグラフ....171
グラフ属性構文を使用したヒートマップ....172
グラフ属性構文を使用した折れ線グラフ....174
グラフ属性構文を使用したマトリックスマーカーグラフ....175
グラフ属性構文を使用したマリメッコチャート....177
グラフ属性構文を使用した円グラフ....179
グラフ属性構文を使用した散布図....181
グラフ属性構文を使用した統計ボックスプロット....182
グラフ属性構文を使用したストリームグラフ....186
グラフ属性構文を使用したタグクラウド....187
グラフ属性構文を使用したツリーマップ....189
グラフ方向の指定 191
棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定 193
絶対棒グラフのインセット幅の制御....195
グラフデータの属性カテゴリへの割り当て 197
カテゴリ内のフィールド順序の割り当て....201
複数 X 軸および複数 Y 軸の指定....203
複数の属性カテゴリへのフィールドの割り当て....205
ソートフィールドに対する属性カテゴリの割り当て順序 207
グラフタイプ別のサポート可能な属性カテゴリの概要 208

5. グラフ全体のプロパティ 215

グラフ全体のプロパティの概要 215

グラフ境界線のフォーマット設定 216
JavaScript エラーの処理方法の定義 219
グラフフレームのフォーマット設定 219
1 行に表示するグラフ数の制御 223
グラフに描画するデータ範囲の定義 225
グラフへの奥行きの適用 227
描画領域への色またはグラデーションの適用 229
グループラベルの定義 232
ライザ、マーカー、円項目へのベベル効果の適用 234
連続するライザへの濃淡効果の適用 239
3D グラフのライザ間隔の制御 241
シリーズとグループの位置の入れ替え 244
シリーズ、グループ、ラベルの位置の入れ替え 246

6. グラフタイトルのプロパティ 251

グラフタイトルのプロパティの概要 251
グラフ脚注のフォーマット設定 252
グラフサブタイトルのフォーマット設定 256
グラフタイトルのフォーマット設定 260

7. 凡例のプロパティ 269

凡例のプロパティの概要 269
凡例領域の背景色の定義 271
凡例ラベルのフォーマット設定 274
円グラフ凡例ラベルでのグラフマクロの使用 276
凡例領域の枠線の生成 277
凡例マーカーの位置の制御 279
凡例マーカーのサイズの制御 282
折れ線グラフの凡例マーカーとシリーズマーカーの一致 284
凡例領域の最大パーセントの設定 286
凡例の方向の制御 288
凡例の位置の制御 290

凡例項目の順序の入れ替え 295
スクロール凡例の生成 298
ドッキング凡例の生成 301
凡例タイトルのフォーマット設定 306
凡例領域の表示の制御 308
サイズ凡例のフォーマット設定 309

8. 軸のプロパティ 315

軸のプロパティの概要 315
軸リストの作成 323
軸の罫線間のセグメントを交互に塗りつぶし 325
ライザ幅の制御 327
積み上げ棒グラフのライザのソート 334
数値軸ベースラインの外観の制御 337
対数目盛りの有効化 339
軸線のフォーマット設定 342
色バンドの定義 344
主罫線、目盛り、ラベルの個数の制御 349
Y 軸主罫線と Y2 軸主罫線の同期 354
数値軸の方向の制御 356
軸ラベルのフォーマット設定 358
主罫線と目盛りのフォーマット設定 367
数値軸での最小値および最大値の定義 373
複数 Y 軸および分割 Y 軸グラフの最大値と最小値の同期 376
数値軸の最小値ラベルと最大値ラベルの置換 379
棒グラフの数値軸のゼロ値の表示 381
補助罫線と目盛りのフォーマット設定 383
X 軸のラベルレイアウトの制御 387
X 軸スクロールのフォーマット設定 390
横方向スクロールバーの初期位置の設定 393
軸モードの設定 397

数値軸ラベルのフォーマットの制御 399
軸のデフォルト表示位置の入れ替え 399
軸タイトルのフォーマット設定 402
Y 軸または Y2 軸のタイトル位置の定義.....408
Y2 軸タイトル表示の反転..... 410
時間軸のフォーマット設定 412
9. シリーズ固有のプロパティ 427
シリーズ固有のプロパティの概要 427
特定シリーズの選択 430
特定グループの選択 434
シリーズライザの境界線の定義 436
シリーズライザの色の定義 442
データテキストラベルの表示およびフォーマット設定 445
円グラフからの円項目の削除 461
円グラフからの円項目の分離 463
特定シリーズへのラベルの割り当て 466
シリーズマーカーのサイズ、境界線、色、形状、回転の定義 467
棒グラフ、折れ線グラフ、面グラフのシリーズライザ形状の定義 482
散布図でのマーカーの接続 484
シリーズライザおよびマーカーでの塗りつぶしパターン名の使用 486
棒グラフのライザ角形状の曲線化 489
ライザのツールヒントの定義または非表示 491
バブルグラフおよび散布図でのシリーズ傾向線の描画 499
棒グラフ、折れ線グラフ、面グラフでのシリーズ傾向線の描画 505
特定シリーズの表示の制御 508
軸へのシリーズの割り当て 510
10. グラフ固有のプロパティ 513
棒グラフ、折れ線グラフ、面グラフのプロパティ (blaProperties) 514
棒グラフ、折れ線グラフ、面グラフのシリーズレイアウトの制御..... 514
棒グラフ、折れ線グラフ、面グラフのグラフ方向の指定..... 516

棒グラフ、折れ線グラフ、面グラフのデータポイント接続方法の制御. 518
棒グラフ、折れ線グラフ、面グラフの塗りつぶし効果の制御. 522
棒グラフ、折れ線グラフ、面グラフのライザグループ間隔の制御. 526
棒ライザの角の丸みの制御. 528
棒グラフ、折れ線グラフ、面グラフの複数 Y 軸表示の制御. 530
積み上げ棒グラフ、折れ線グラフ、面グラフの合計ラベルの制御. 532
棒、折れ線、面グラフでの特定データラベルの表示. 535
複合グラフのプロパティの指定. 537
棒グラフ、折れ線グラフ、面グラフでの Null 値表示の制御. 540
ボックスプロットおよび統計ボックスプロットのプロパティ (boxPlotProperties) 546
ボックスプロットのハット幅の指定. 547
ボックスプロットのハットの外観制御. 550
ボックスプロットのノッチの外観制御. 552
ボックスプロットの中央値線の外観制御. 554
ボックスプロットの接続線の外観制御. 557
ボックスプロットの平均値マーカーの外観制御. 559
ブレットグラフのプロパティ (bulletProperties) 562
データグリッドのプロパティ (dataGridProperties) 564
データグリッド内の 1 行おきに異なる背景色の定義. 568
データグリッドのセルのフォーマット設定. 569
データグリッドの行見出しのフォーマット設定. 581
データグリッドの列見出しのフォーマット設定. 596
データグリッドの列合計の生成とフォーマット設定. 616
データグリッドのスクロールバーの生成とフォーマット設定. 621
じょうごグラフのプロパティ (funnelProperties) 624
じょうご上端幅の制御. 625
じょうご下端幅の制御. 626
じょうごグラフのレイヤ間隔の制御. 627
じょうごグラフのグループラベルのフォーマット設定. 629
メータグラフのプロパティ (gaugeProperties) 631

メータグラフのレイアウト制御. 633
メータ軸の色バンドの生成. 636
メータグラフの主罫線目盛り長さの制御. 638
メータグラフの補助罫線目盛り長さの制御. 640
メータグラフの軸幅の制御. 642
メータ面の内側塗りつぶし色の制御. 643
メータグラフのグループラベルのフォーマット設定. 645
メータ針の回転軸部のフォーマット設定. 647
メータグラフの外側境界のフォーマット設定. 649
メータグラフの軸の開始角度と終了角度の制御. 652
シンプルメータグラフのセカンダリリング塗りつぶし色の制御. 654
メータグラフの合計ラベルのフォーマット設定. 656
ヒストグラムのプロパティ (histogramProperties) 658
マトリックスグラフのプロパティ (matrixProperties) 660
マトリックスの各セルに表示されるグラフタイプの制御. 662
マトリックス列グラフのファセットモード. 664
マトリックス行ラベルのフォーマット設定. 666
マトリックス列ラベルのフォーマット設定. 668
マトリックス行見出しのフォーマット設定. 670
マトリックス列見出しのフォーマット設定. 672
マトリックスセル境界線のフォーマット設定. 674
マトリックスセルのセルパディングの制御. 676
マトリックスグラフの最小セルサイズの制御. 679
マトリックスグラフの数値軸範囲の設定. 681
パラボックスのプロパティ (paraboxProperties) 687
円グラフのプロパティ (pieProperties) 689
円項目クリック時のアニメーションの制御. 690
リンググラフの描画 (中央に穴の開いた円グラフ). 692
リンググラフ円項目端部への丸みの適用. 695
円グラフのラベルのフォーマット設定. 696

同心円グラフの生成. 699
指定した値未満の円項目の統合. 700
円グラフの回転. 705
円グラフの傾斜. 707
円グラフの円項目のソート. 708
円グラフの合計ラベルのフォーマット設定. 712
極グラフのプロパティ (polarProperties) 714
タグクラウドのプロパティ (tagloudProperties) 719
3D グラフのプロパティ (threedProperties) 722
ツリーマップのプロパティ (treemapProperties) 725
ツリーマップセルのフォントの拡大縮小. 727
ツリーマップのヘッダのフォーマット設定. 728
ツリーマップのセル境界線のフォーマット設定. 732
滝型グラフのプロパティ (waterfallProperties) 734
滝型グラフでの合計ライザの追加. 735
合計ライザへのカスタムラベルの追加. 738
滝型グラフの接続線のフォーマット設定. 739
滝型グラフでの負の値のライザ色の制御. 741
滝型グラフの「その他」ライザ色の指定. 744
「その他」ライザの指定. 746
滝型グラフでの正の値のライザ色の指定. 748
中間合計値の指定. 750
滝型グラフの中間合計ライザ色および合計ライザ色の指定. 752
中間合計ライザの軸ラベルおよびデータラベルプロパティのフォーマット設定. 754
滝型グラフでの最初および最後のライザのフォーマット設定. 758

11. グラフアクセシビリティのプロパティ 763

アクセシビリティプロパティの概要 763
アクセシビリティのタイトルおよび説明の定義 763
グラフのライザおよびマーカーでの Tab キーナビゲーションの有効化 765

12. その他の機能 769

アニメーション (introAnimation) 769
モーフアニメーションの継続時間 (morphAnimation) 770
データページスライダの速度制御 770
JSON を使用したグラフイベント登録 771
グラフ上の注釈 775
PDF、Excel、PowerPoint、PNG フォーマットで使用する HTML5 グラフの作成 794
色モード (colorMode) 802
色モードの色の定義. 809
色スケールの定義 (colorScale) 812
色スケールの色モードの定義. 819
ピクトグラムのサポート 827
エラーバーの描画 (errorBars) 838
HTML ツールヒントの生成 (htmlToolTip) 842
グラフでのユーザ操作の定義 (interaction) 853
マウスオーバーインジケータのフォーマット設定 (mouseOverIndicator) 859
参照線の描画 (referenceLines) 863
バブルグラフおよび散布図での傾向線の描画 (trendline) 867

13. マップサポート 871

マップサポートの概要 872
マップグラフの TIBCO WebFOCUS パラメータおよび属性カテゴリ 872
マッププロパティの概要 875
マップレイヤの理解. 875
ESRI マップグラフの作成 876
TIBCO WebFOCUS Reporting Server 構文による ESRI マップグラフの生成. 878
TIBCO WebFOCUS Rorting Server 構文による ESRI マップレイヤの定義. 878
TIBCO WebFOCUS Reporting Server の地理構成ファイル. 893
グラフエンジン構文による ESRI マップグラフの生成. 906
グラフエンジン構文による区切りファイルのサポート. 928
グラフエンジン構文による Geojson ファイルのサポート. 935
ESRI マップ出力のカスタマイズ. 939

Leaflet マップグラフの生成 978
初期位置の定義 995
マップの不透明度の調整 998
マップビューア内外の要素の調整 998
マップの縮尺コントロールおよびレイヤコントロールの調整 998
マップグラフでのグラフプロパティの統合 1000
コロプレスマップでの色スケールの調整 1000
Leaflet マップグラフでの Hover 色の調整 1002
バブルマップでのマーカーの色とサイズの調整 1003
コロプレスマップの色およびツールヒントの定義 1007
バブルマップでのデータテキストラベルの表示 1012
コロプレスマップでのデータテキストラベルとしての地名表示 1016
14. ユーザ独自グラフタイプのグラフライブラリへの追加 1021
グラフ拡張機能の概要 1021
グラフ拡張機能の作成 1022
グラフ拡張機能 API の使用 1023
グラフの描画 1023
グラフ拡張機能の構成 1028
グラフエンジン構成の config オブジェクト作成 1028
グラフインターフェースの構成 1030
拡張機能のデータアクセス 1036
拡張機能のバケットの定義と使用 1037
拡張機能の部分データおよび NULL データの処理 1040
グラフ拡張機能のインストール 1041
ibi WebFOCUS リクエストでの拡張機能の使用 1043
A. TIBCO WebFOCUS グラフのパラメータ 1049
JSON chartType プロパティによるグラフタイプの制御 1049
複数グラフの作成 1051
SET パラメータによるグラフのカスタマイズ 1055
B. リクエストのグラフ属性構文への変換 1061

変換規則の概要 1061
棒グラフ、折れ線グラフ、面グラフのリクエストのグラフ属性構文への変換 1062
ボックスプロットリクエストのグラフ属性構文への変換 1072
バブルグラフリクエストのグラフ属性構文への変換 1076
じょうごグラフリクエストのグラフ属性構文への変換 1080
マップグラフリクエストのグラフ属性構文への変換 1084
マリメッコチャートリクエストのグラフ属性構文への変換 1104
円グラフリクエストのグラフ属性構文への変換 1109
散布図リクエストのグラフ属性構文への変換 1116
ブロック地図リクエストのヒートマップリクエストへの変換 1128
ストリームグラフリクエストのグラフ属性構文への変換 1133
タグクラウドリクエストのグラフ属性構文への変換 1136
ツリーマップリクエストのグラフ属性構文への変換 1140
ネストされた X 軸を含むリクエストのグラフ属性構文への変換 1145

C. JSON プロパティの変更累積リスト 1149

WebFOCUS バージョン 8.2 での JSON プロパティの変更 1149
WebFOCUS バージョン 8.1 での JSON プロパティの変更 1149

Legal and Third-Party Notices 1151

1

TIBCO WebFOCUS HTML5 グラフの概要

JavaScript グラフエンジン (FORMAT JSCHART) は、TIBCO WebFOCUS GRAPH リクエストおよび拡張オプションの JSON (JavaScript Object Notion) 定義を使用して、Web ブラウザ (HTML 環境) でグラフを描画します。JSON オブジェクトでは、グラフおよびグラフオブジェクトの外観を調整するグラフプロパティを定義します。

トピックス

- TIBCO WebFOCUS グラフの概要
- グラフタイプ
- HTML5 グラフの作成
- TIBCO WebFOCUS GRAPH FILE コマンドの概要
- サンプルファイルの生成

TIBCO WebFOCUS グラフの概要

グラフを作成すると、色、サイズ、位置などの視覚的要素を使用して情報がグラフィカルに表現され、メジャー (集計する数値フィールド) とディメンション (カテゴリ) の関係を把握できるとともに、傾向や異常値の識別が容易になります。WebFOCUS では、ユーザが選択した出力フォーマットで、さまざまなグラフを作成することができます。

WebFOCUS でグラフを作成する方法の 1 つとして、TIBCO WebFOCUS Reporting Server で生成するグラフがあります。この方法で生成されたグラフは、ビットマップイメージ (例、.png、.gif、.jpg) としてブラウザに送信されるか、PDF ドキュメントに埋め込まれたベクタフォーマットで出力されます。

一方、HTML5 グラフは、ブラウザで直接実行される JavaScript コードとして作成されるため、高度なインタラクティブ機能を備えたグラフが生成されます。これらのグラフは、現在のすべての Web 標準に準拠するとともに、すべての最新ブラウザでサポートされます。ブラウザおよびサポートされるバージョンの一覧は、『ibi™ WebFOCUS® リリースノート/新機能ガイド』を参照してください。

また、HTML5 グラフを作成して、PDF、Excel、PowerPoint、.PNG フォーマットの出力ファイルで使用することもできます。これらのフォーマットで表示されるグラフは、ブラウザで表示される HTML5 グラフ出力の静的イメージです。詳細は、794 ページの 「 PDF、Excel、PowerPoint、PNG フォーマットで使用する HTML5 グラフの作成 」 を参照してください。

グラフを作成する手順は、TIBCO WebFOCUS ツールを使用して作業領域に要素をドラッグし、ツールバーやリボンからコマンドを選択するという方法ですが、このプロセスの基本は、使い慣れた、分かりやすい WebFOCUS 言語を使用する点にあります。HTML5 グラフでは、これらの簡単な WebFOCUS コマンド以外に、出力結果を詳細にカスタマイズするための JSON API もサポートされます。複数のグラフを関連付けることもできます。

また、WebFOCUS HTML5 グラフ製品では、マップがサポートされます。地図上の領域が色分けされたコロプレスマップを作成することも、地図上で関連する領域にサイズの異なるバブルを表示するプロポーショナルシンボルマップを作成することもできます。

WebFOCUS 言語を使用して HTML5 グラフを作成するには、リクエストに PCHOLD FORMAT JSCHART コマンドを含めます。

また、WebFOCUS スタイルシートコマンドを使用して、属性カテゴリにグラフオブジェクトを配置する強力なグラフ属性構文も備えており、有意義なグラフ生成のための追加オプションやグラフタイプを提供します。グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

注意：HTML5 グラフの生成には、グラフ属性構文を使用してください。生成するグラフタイプで使用可能な場合は、常にこの構文を使用します。

WebFOCUS グラフリクエストで参照するデータソースは、JOIN で結合されたデータソースにすることも、クロスリファレンスされたデータソースにすることもできます。また、すべてのオペレーティング環境で、データソースを Hyperstage キャッシュに格納することも、サポートされている任意の DBMS に格納することもできます。リクエストにダイアログマネージャの変数およびコマンドを使用すると、実行時にユーザ選択値に基づいてカスタマイズが可能なグラフを作成することができます。さらに、WebFOCUS グラフでは、さまざまな WebFOCUS スタイルオプションがサポートされます。

リクエストで参照するデータソースには、シノニムと呼ばれる WebFOCUS 定義が必要です。シノニムは、データの内容を定義したマスターファイルと、ほとんどの場合、データへのアクセス方法を定義したアクセスファイルで構成されます。シノニムエディタを使用すると、ほとんどのタイプのデータソースのシノニムを自動的に作成することができます。シノニム作成についての詳細は、『ibi™ WebFOCUS® メタデータリファレンス』を参照してください。

JSON オブジェクトでは、高度なグラフプロパティの値を定義します。JSCHART フォーマットでは、標準の JSON オブジェクトを使用することも、一部のプロパティのみが含まれた JSON オブジェクトを使用することもできます。これらのプロパティ値は、デフォルト値または既存の設定値の代わりに使用されます。

グラフタイプ

ここでは、JSCHART フォーマットを使用して作成可能なグラフタイプの代表的なものについて説明します。

- **折れ線グラフ** 折れ線グラフは、時間の経過とともに変化する数値データの動向や傾向を強調する際に役立ちます。これは、過去の状態を観察したり補間したりすることによって、特定の点の変動を追跡できるためです。高低のレベル、遅速の動き、安定に向けての傾向は、折れ線グラフが適するトレンドタイプです。

  また、折れ線グラフに複数の目盛りを使用して、異なる期間においての同一データ値または複数データ値を比較することもできます。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

- **棒グラフ** 棒グラフは、スケールを基準にした長方形のブロックを表示することによって、数値データを表します。棒の長さは、各データの値または数量に相当します。棒の長さを対比することにより、データ間の比較結果を明確なイメージとして頭の中に描くことができます。数値データを表示してデータの分布状況について説明する場合は、棒グラフが適しています。縦方向の棒グラフだけでなく、横方向の棒グラフも作成することができます。

- **円グラフ** 円グラフは、特定のデータが全体の中でどの程度の割合を占めるかを強調します。データ全体が比較的大規模なデータセットで構成される場合に、円グラフが適しています。変数が多すぎると、円が小さいセグメントに分割されるため、判読が困難になります。視覚的なコントラストを付けるには、個々のセグメントに色またはテクスチャを使用します。

- **散布図** 散布図は、2 つの異なる数値メジャーの関係を示します。散布図を使用すると、傾向、密集度、異常値を感覚的に把握できるため、さらに調査を要する箇所の特定に役立ちます。

- **面グラフ** 面グラフは、折れ線グラフに類似していますが、データ線とゼロの線 (軸線) との間の領域に色またはテクスチャが適用されている点が異なります。面グラフでは、1 つのデータの上に別のデータを積み上げることができます。積み上げにより、あるデータシリーズが 2 つ目のデータシリーズにどのように近づくか、追い越すかを表示し、データ間の関係を強調することができます。

- **3D グラフ** 3D グラフを使用すると、グラフに奥行きが追加されます。グラフを立体的に表示することで、複数のデータセットの傾向が認識しやすくなります。また、3D グラフを作成することで、グラフがインパクトのある外観になります。
- **極グラフ** 極グラフは、散布図を円形にしたグラフです。各ポイントは、角度およびグラフ中心からの距離で定義されます。
- **レーダーグラフ** レーダーグラフは、折れ線グラフまたは面グラフを円形にしたグラフです。年月のような周期データについて説明する場合は、レーダーグラフが適しています。
- **株価グラフ** 株価グラフは、金融商品が時間の経過ともに変化する傾向や変動量を表します。
- **バブルグラフ** バブルグラフは散布図を拡張したグラフで、3 つ目のメジャー値に比例して各マーカーのサイズが変化します。
- **ブロック地図** ブロック地図は、ヒートマップとも呼ばれ、テーブルの各セルの色がセル内の値に応じて変化します。
- **その他のグラフ** 上記のグラフタイプ以外に、LOOKGRAPH パラメータで指定可能なグラフタイプが多数あります。たとえば、滝型グラフは、初期値に正の値または負の値を連続的に加算することで、累積結果を視覚的に表現するグラフです。
- **HTML5 固有のグラフ** 一部のグラフタイプは、FORMAT JSCHART を指定した場合にのみ使用できます。これらのグラフタイプには、マップグラフ (さまざまな値グループを色分けしたコロプレスマップ、サイズの異なるバブルを識別子として使用するプロポーショナルシンボルマップ)、タグクラウド (発生頻度に比例したサイズで単語を表示)、ツリーマップ (階層データをネストされた長方形の集合体として表示) があります。

作成可能なグラフタイプのリスト、および各グラフタイプの生成に使用する構文についての詳細は、40 ページの 「 グラフタイプの制御 」 および 123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

HTML5 グラフの作成

HTML5 グラフを生成するには、GRAPH FILE リクエストに次のコマンドを追加します。

```
ON GRAPH PCHOLD FORMAT JSCHART
```

ON GRAPH PCHOLD FORMAT JSCHART コマンドまたは ON GRAPH SET GRAPHENGINE JSCHART コマンドが発行されていない場合は、レガシーグラフが作成されます。

WebFOCUS スタイルコマンド、JSON オブジェクトのライブラリ、レガシーグラフ API コールを使用して、ユーザのニーズや基準に準拠するよう出力結果をカスタマイズすることができます。

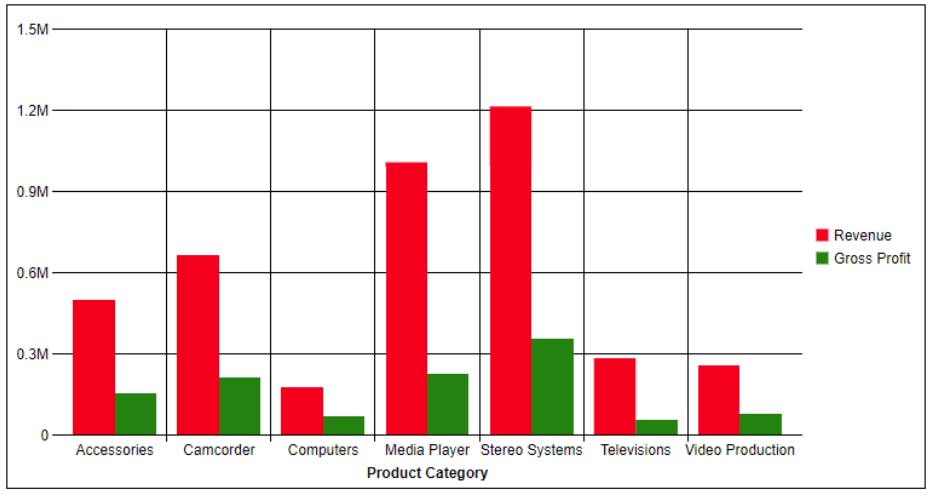
JSON プロパティを指定しない場合は、デフォルトのプロパティが使用されます。WebFOCUS スタイルシートを使用するよう指定しない場合は、デフォルトのスタイルが使用されます。

例　HTML5 縦棒グラフの作成

次のリクエストは、WF_RETAIL_LITE データソースを使用して、HTML5 縦棒グラフを作成します。このリクエストではスタイルシートおよび JSON プロパティが指定されていないため、デフォルトのプロパティが使用されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=Y-AXIS,$
ENDSTYLE
END
```

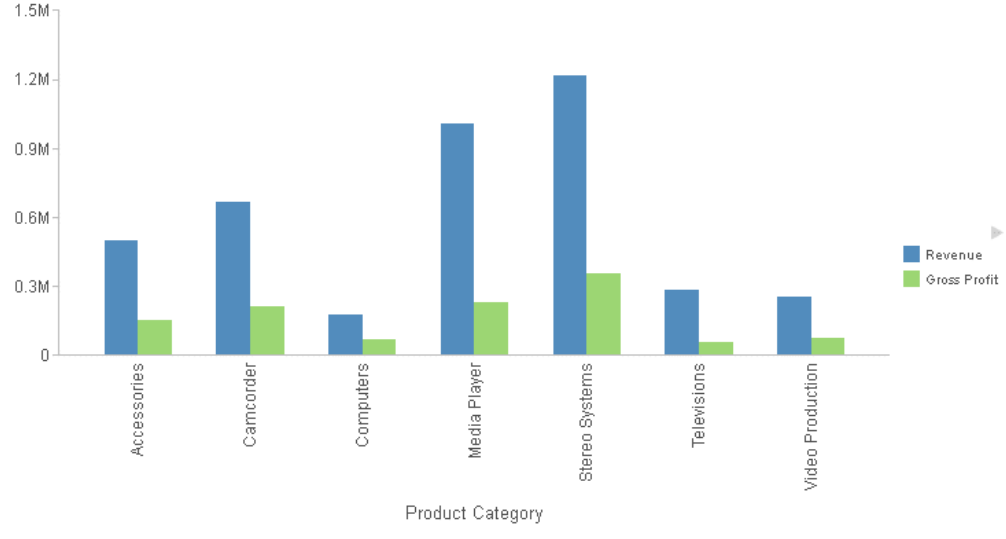
出力結果は次のとおりです。



次のリクエストでは、スタイルシートを適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=Y-AXIS,$
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　コンテナに格納する HTML5 グラフ出力のサイズ変更

SET AUTOFIT コマンドを使用して、HTML5 グラフが配置先のコンテナ内に収まるようにグラフ出力のサイズを自動的に変更することができます。ON 設定を使用して幅と高さの両方を自動的に変更することも、WIDTH 設定を使用して幅のみを自動的に変更することもできます。

```
ON GRAPH SET AUTOFIT {OFF|ON|RESIZE|WIDTH}
```

説明

OFF

HAXIS および VAXIS パラメータで指定されたサイズに固定されます。

ON

コンテナ内に収まるように、HTML5 グラフ出力の高さと幅が常に自動的に変更されます。

RESIZE

最初は HAXIS および VAXIS パラメータで指定されたサイズに固定されますが、コンテナのサイズが変更された場合はグラフ出力のサイズも変更されます。

WIDTH

コンテナ内に収まるように、グラフ出力の幅のみが自動的に変更され、高さは自動的に変更されません。

注意

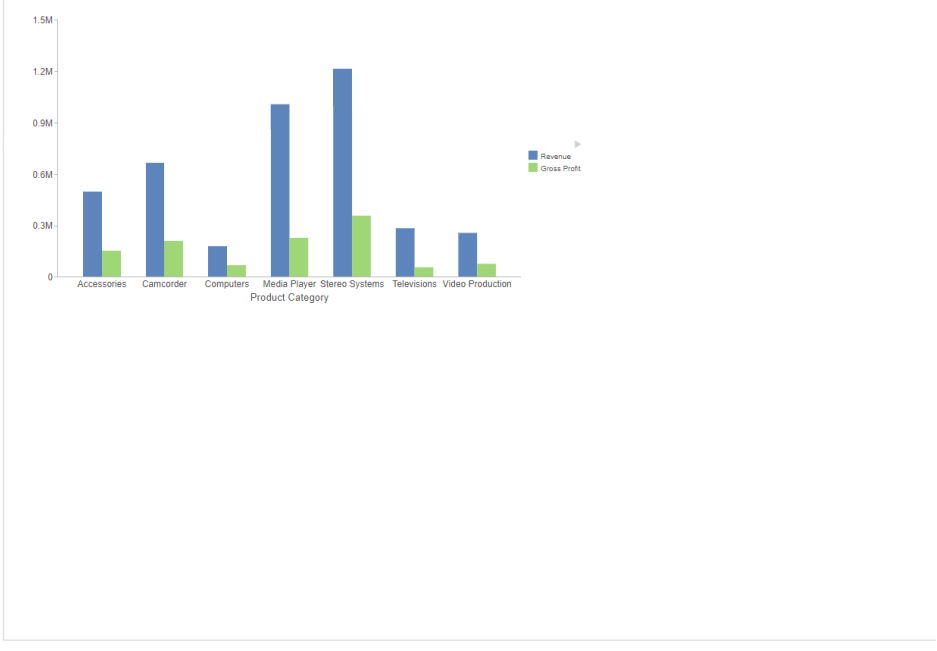
- AUTOFIT を有効にした場合、GRAPH FILE リクエストの出力結果は描画領域全体が収まるようサイズ変更されます。そのため、プロシジャに複数の GRAPH FILE リクエストが含まれている場合は、AUTOFIT パラメータを OFF に設定します。
- 従来の GMERGE 構文を使用した複数グラフリクエストでは、AUTOFIT ON は無視されます。複数グラフリクエストについての詳細は、1051 ページの「 複数グラフの作成 」を参照してください。
- ソートフィールドを page 属性カテゴリに割り当てて複数グラフを生成する場合は、AUTOFIT ON および AUTOFIT-MULTI ON の両方を設定する必要があります。詳細は、24 ページの 「 コンテナ内の複数グラフのサイズ調整 」 を参照してください。
- プロシジャに複数の GRAPH FILE リクエストが含まれ、 SET AUTOFIT=ON に設定されている場合、出力結果には最後のグラフのみが表示されます。

例 SET AUTOFIT の使用

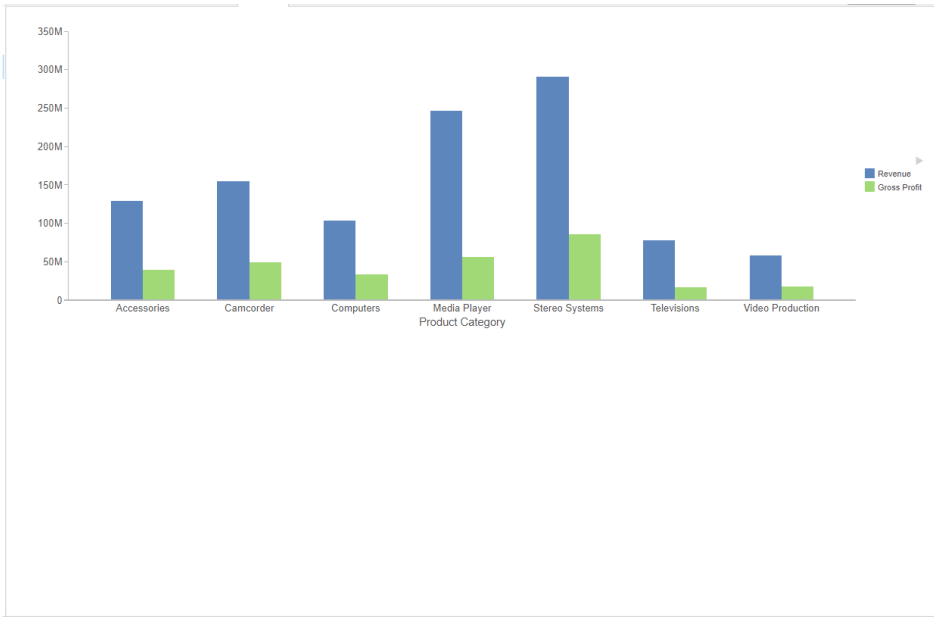
次のリクエストは、縦棒グラフを生成します。AUTOFIT パラメータは OFF に設定されています。

```
SET AUTOFIT = OFF
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN = PRODUCT_CATEGORY, BUCKET = x-axis,$
TYPE=DATA, COLUMN = REVENUE_US, BUCKET = y-axis,$
TYPE=DATA, COLUMN = GROSS_PROFIT_US, BUCKET = y-axis,$
ENDSTYLE
END
```

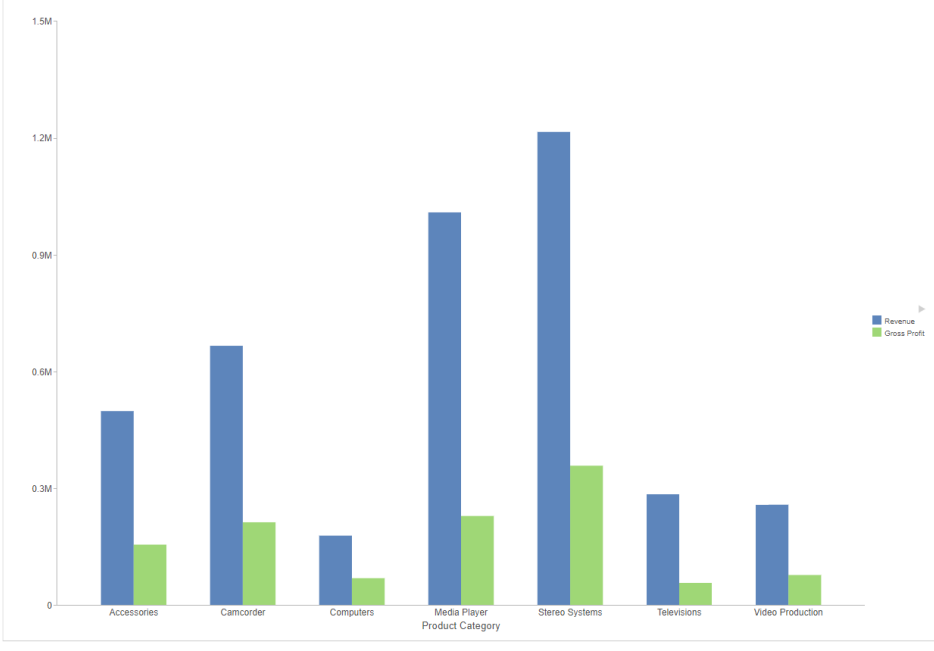
下図のように、グラフサイズは、HAXIS および VAXIS パラメータのデフォルト設定で制御されます。



AUTOFIT パラメータ値を WIDTH に変更すると、下図のように、グラフ幅はページ全体を使用して描画されますが、高さは自動的に変更されません (VAXIS が有効)。



AUTOFIT パラメータ値を ON に変更すると、下図のように、グラフがページ全体に描画され、HAXIS および VAXIS パラメータ値は無視されます。



コンテナのサイズを変更した場合、グラフのサイズも変更されます。

AUTOFIT 設定を RESIZE に変更すると、初期状態では OFF 設定の場合と同様にグラフが描画されますが、コンテナが拡大または縮小されると、グラフのサイズも変更されます。

構文　コンテナ内の複数グラフのサイズ調整

SET AUTOFIT コマンドは、HTML5 グラフが配置先のコンテナ内に収まるようにグラフ出力のサイズを自動的に変更するかどうかを制御します。新しい 2 つの SET コマンドが、複数のグラフを生成するリクエストに対して AUTOFIT を実行します。これらのコマンドを実行するには、AUTOFIT の有効化も必要です。

- SET AUTOFIT-MULTI は、コンテナ内の複数グラフのサイズを自動的に変更するかどうかを定義します。
- SET AUTOFIT-MIN-PCT は、各グラフの最小高さを定義します。

構文は次のとおりです。

```
ON GRAPH SET AUTOFIT-MULTI {ON|OFF}
ON GRAPH SET AUTOFIT-MIN-PCT pct
```

説明

ON
: コンテナ内の複数グラフの自動サイズ調整を有効にします。AUTOFIT も、ON または RESIZE に設定する必要があります。

OFF
: 複数グラフの自動サイズ調整を無効にします。

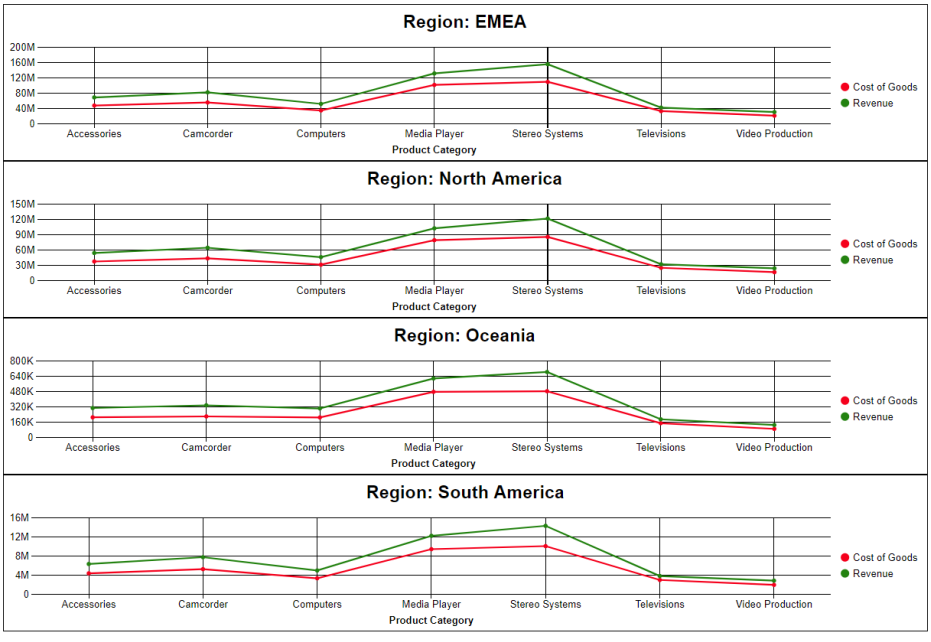
PCT
: 各グラフの最小高さを、VAXIS 値のパーセントで指定します。AUTOFIT-MIN-PCT のデフォルト値は 100 です。この最小高さで、コンテナ内にグラフがすべて縦方向に収まらない場合、スクロールバーが追加されたウィンドウが生成されます。

例　コンテナ内の複数グラフの自動サイズ調整

次のリクエストは、BUSINESS_REGION フィールドの各値についてグラフを生成します。AUTOFIT および AUTOFIT-MULTI はともに ON に設定され、AUTOFIT-MIN-PCT は VAXIS 値の 50% に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"Region: <BUSINESS_REGION"
SUM COGS_US REVENUE_US
BY BUSINESS_REGION NOPRINT
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET VAXIS 200
ON GRAPH SET AUTOFIT ON
ON GRAPH SET AUTOFIT-MULTI ON
ON GRAPH SET AUTOFIT-MIN-PCT 50
ON GRAPH SET STYLE *
type=heading, size=16, style=bold, $
TYPE=DATA, COLUMN=N1, BUCKET=page, $
TYPE=DATA, COLUMN=N2, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
END
```

出力結果は次のとおりです。4 つのグラフすべてがウィンドウに収まり、スクロールバーは生成されません。



参照　棒グラフ、折れ線グラフ、面グラフのスクロールバー生成

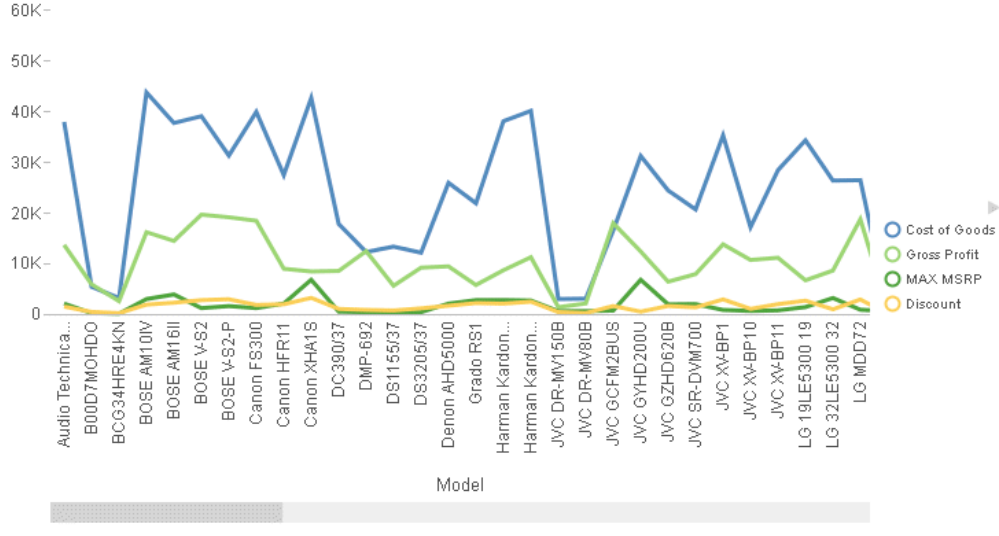
棒グラフ、折れ線グラフ、面グラフのグループエントリ数が多い場合は、スクロールバーが生成され、スクロールすることによってすべてのグループを表示することができます。

例　　折れ線グラフのスクロールバー生成

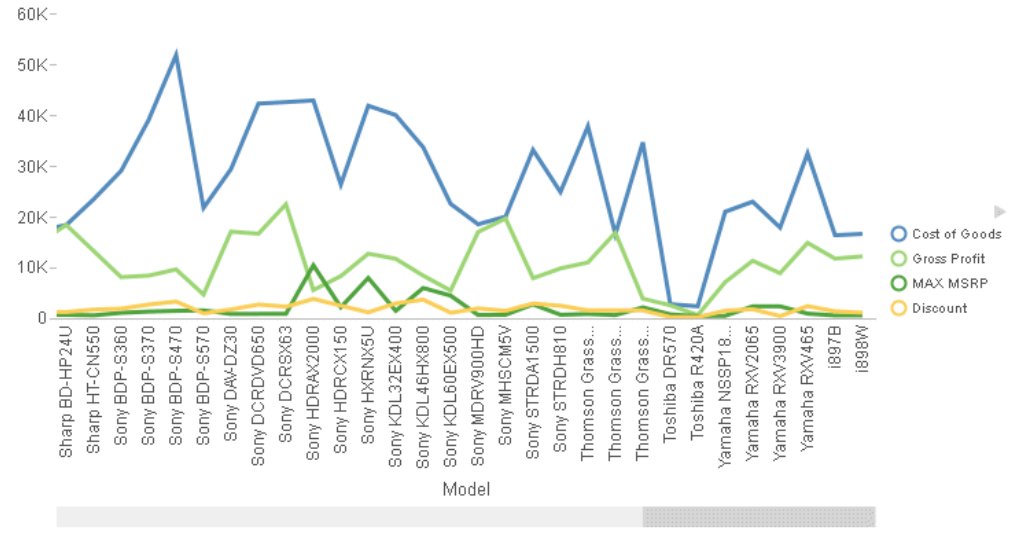
次のリクエストでは、グループエントリの数が大きすぎるため、描画領域に一度にすべてのグループを表示することができません。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US MAX.MSRP_US DISCOUNT_US
BY MODEL
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=model, bucket=x-axis,$
type=data, column=COGS_US , bucket=y-axis,$
type=data, column=GROSS_PROFIT_US, bucket=y-axis,$
type=data, column=MAX.MSRP_US, bucket=y-axis,$
type=data, column=DISCOUNT_US, bucket=y-axis,$
ENDSTYLE
END
```

出力結果は次のとおりです。下図ではグループエントリがすべて表示されていませんが、スクロールバーを使用することですべてのグループの値を表示することができます。



下図は、スクロールバーを最後のグループエントリまでドラッグした状態を示しています。



TIBCO WebFOCUS GRAPH FILE コマンドの概要

GRAPH FILE リクエスト構文は、TABLE FILE リクエスト構文に類似しています。表形式ではなく、グラフ形式でレポートを作成するには、リクエスト内の TABLE コマンドを GRAPH に変更します。このように、TABLE リクエストを GRAPH リクエストに変換するだけで、グラフを作成できる場合があります。

ただし、すべての TABLE には対応する GRAPH が存在するわけではなく、グラフで効果的に表すことのできる情報量にも実質上の制限があります。この方法で TABLE リクエストを変換すると、リクエストの本文を構成する句に基づいて、グラフのフォーマットおよびレイアウトが決定されます。GRAPH FILE リクエストで生成されるグラフのタイプは、使用される表示コマンド (SUM または PRINT)、ソート句 (BY または ACROSS)、および LOOKGRAPH パラメータ設定により決定されます。

TIBCO WebFOCUS HTML5 GRAPH リクエストの構造

ここでは、GRAPH FILE リクエストの構造について説明します。

```
GRAPH FILE filename

[HEADING ...]
```

```
{PRINT|SUM} field ...
{BY|ACROSS} sortfield ...

[{WHERE|IF} expression ...[WHEN condition]] ...

[FOOTING ...]

ON GRAPH PCHOLD FORMAT JSCHART

[ON GRAPH SET LOOKGRAPH charttype]
[ON GRAPH SET optional_parameter ...]

ON GRAPH SET STYLE *
[*GRAPH_JS
 JSON properties
*END]
[*GRAPH_SCRIPT
 Legacy Graph API methods and properties
 *END]
[WebFOCUS StyleSheet declarations ]
[*GRAPH_JS_FINAL
 JSON properties to override server properties
*END]
ENDSTYLE
END
```

下表は、各コマンドの説明です。

リクエスト構文	説明
`GRAPH FILE filename`	グラフリクエストを開始します (必須)。
`[HEADING ...]`	グラフタイトルです (オプション)。 デフォルト設定では、HTML ページ上のグラフの上側に見出しが配置されます。グラフに見出しを埋め込むには、リクエストに ON GRAPH SET EMBEDHEADING ON コマンドを含めます。この設定では、埋め込まれた見出しが JSON グラフタイトルオブジェクトの代わりに使用されます。リクエストに JSON タイトルオブジェクトも含めた場合は、WebFOCUS 見出しが JSON タイトルで上書きされます。

リクエスト構文	説明
`{PRINT|SUM} field ...`	表示コマンドです (必須)。フィールドは数値フォーマットにする必要があります。実フィールド、一時項目 (DEFINE)、一時項目 (COMPUTE) のいずれかを使用できます。PRINT を指定すると、各フィールド値が表示され、SUM を指定すると、ソート値ごとにフィールド値が集計されます。
`{BY|ACROSS} sortfield ...`	ソート句です (必須)。
`[{WHERE|IF} expression ...[WHEN condition]] ...`	フィルタ条件です (オプション)。
`[FOOTING ...]`	グラフ脚注です (オプション)。 デフォルト設定では、HTML ページ上のグラフの下側に脚注が配置されます。グラフに脚注を埋め込むには、リクエストに ON GRAPH SET EMBEDHEADING コマンドを含めます。この設定では、埋め込まれた脚注が JSON グラフ脚注オブジェクトの代わりに使用されます。リクエストに JSON 脚注オブジェクトも含めた場合は、WebFOCUS 脚注が JSON 脚注で上書きされます。
`ON GRAPH PCHOLD FORMAT JSCHART`	HTML5 出力を作成するための SET コマンドです (必須)。

`[ON GRAPH SET LOOKGRAPH charttype`
 	
`ON GRAPH SET AUTOFIT ON`
`ON GRAPH SET GRAPHDEFAULT OFF`
`ON GRAPH SET VZERO OFF]`
	SET コマンドです (オプション)。オプションの SET コマンドについての詳細は、1049 ページの 「 TIBCO WebFOCUS グラフのパラメータ 」 および『ibi™ WebFOCUS® アプリケーション作成ガイド』を参照してください。

リクエスト構文	説明
`ON GRAPH SET STYLE *`	リクエストのスタイルセクションを開始します。 スタイルセクションには、JSON ブロック、レガシーグラフ API ブロック、WebFOCUS スタイルシート宣言を含めることができます。これらは任意の順序で配置できます。また、タイプごとに複数のブロックを使用することができます。 プロパティが競合する場合は、リクエストのスタイルセクションで最後に指定されたプロパティが優先されます。

`[*GRAPH_JS`
`JSON properties`
`*END]`
	スタイルセクションに追加する JSON ブロック (オプション) の開始および終了を指定します。

`[*GRAPH_SCRIPT`
`Legacy graph methods and properties`
`*END]`
	スタイルセクションに追加するレガシーグラフ API ブロック (オプション) の開始および終了を指定します。
	これらのブロックは、下位互換性を保持するためにサポートされます。そのため、これらの API コールが含まれている既存のリクエストは、変更を加えることなく HTML5 グラフに変換することができます。ただし、ON GRAPH PCHOLD FORMAT JSCHART コマンドを含める必要があります。

`[WebFOCUS StyleSheet declarations]`	グラフ属性構文宣言についての詳細は、123 ページの「 TIBCO WebFOCUS グラフ構文 」 およびその他オプションの WebFOCUS スタイルシート宣言を参照してください。

リクエスト構文	説明
[*GRAPH_JS_FINAL JSON properties to override server properties *END]	サーバにより設定されたプロパティを上書きできるスタイルセクションの JSON ブロック (オプション) を開始および終了します。 グラフ属性構文を使用した場合、サーバがタイトルやツールヒントなど特定のプロパティを設定します。これらのプロパティの一部は、WebFOCUS リクエストでカスタマイズすることができます。たとえば、列タイトルは、リクエスト内の AS 名を使用してカスタマイズすることができます。プロパティがリクエスト構文で上書きできない場合は、スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加することができます。
ENDSTYLE	リクエストのスタイルセクションを終了します。
END	リクエストを終了します (必須)。

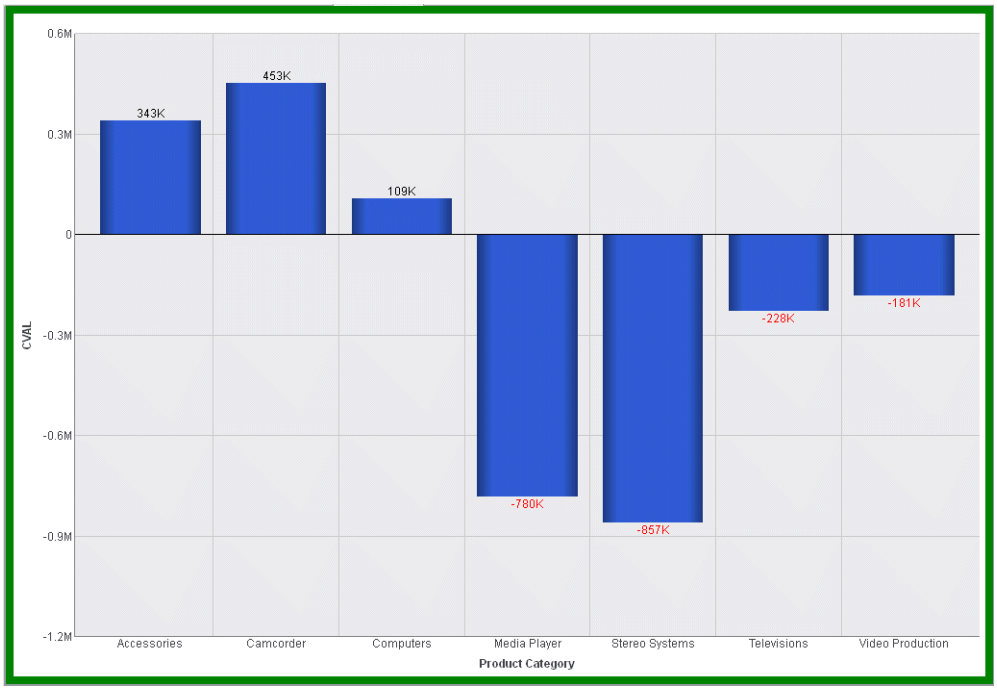
例　HTML5 GRAPH FILE リクエストの例

次のリクエストは、縦棒グラフを作成します (ON GRAPH SET LOOKGRAPH BAR)。Y 軸には CVAL という名前の一時項目 (DEFINE)、X 軸には PRODUCT_CATEGORY というソートフィールドを表示します。

```
-* Create virtual field
DEFINE FILE WF_RETAIL_LITE
CVAL/D12.2= IF PRODUCT_CATEGORY GT 'M' THEN -COGS_US ELSE COGS_US;
END
-*Define default values for amper variables referenced in the request
-DEFAULTH &WF_STYLE_UNITS='PIXELS';
-DEFAULTH &WF_STYLE_HEIGHT='405.0';
-DEFAULTH &WF_STYLE_WIDTH='770.0';
-* GRAPH request starts
GRAPH FILE WF_RETAIL_LITE
SUM CVAL
BY PRODUCT_CATEGORY
-*Required SET parameter for HTML5 output
ON GRAPH PCHOLD FORMAT JSCHART
-*Optional SET parameters
ON GRAPH SET GRAPHDEFAULT OFF
ON GRAPH SET UNITS &WF_STYLE_UNITS
ON GRAPH SET HAXIS &WF_STYLE_WIDTH
ON GRAPH SET VAXIS &WF_STYLE_HEIGHT
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT ON
-*Style section starts
ON GRAPH SET STYLE *
-*WebFOCUS StyleSheet declarations start
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, PAGECOLOR = Green,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=CVAL, BUCKET=Y-AXIS,$
-*JSON block starts
*GRAPH_JS
"mouseOverIndicator": {
"enabled": true,
"color": " "
},
"introAnimation": {
"enabled": true,
"duration": 1000
}
*END


-*Legacy graph API block starts
*GRAPH_SCRIPT
setUseNegativeDataTextColor(true);
setDataTextDisplay(true);
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

- スタイルセクションで指定された JSON ブロックにより、グラフの描画がアニメーション化されます。また、ライザの上にマウスポインタを置いた際に、ライザの色が変わり、ツールヒントが表示されます。
- WebFOCUS スタイルシート宣言は、リクエスト内のフィールドをグラフ属性カテゴリに割り当てます。これにより、埋め込みスタイルシートファイルからグラフスタイルが取得され、グラフが描画されるページの色が緑色に指定されます。
- レガシーグラフ API ブロックにより、負の値のライザのデータテキストを赤色で表示します (JSON プロパティを使用しても同じことが可能です)。



GRAPH と TABLE の類似点

通常、GRAPH リクエスト要素は、対応する TABLE リクエスト要素と同一の規則に準拠します。

- 「FILE」およびファイル名は、事前に SET FILE コマンドで指定されている場合を除いて、GRAPH コマンドの直後に指定する必要があります。

```
SET FILE=filename
```

JOIN 構造やクロスリファレンス構造を含め、WebFOCUS で利用可能な任意のデータソースを指定することができます。

- MORE コマンドを使用することにより、GRAPH リクエスト内の異なるタイプのデータソースファイルを連結することができます。
- HEADING および FOOTING 句を使用することで、グラフに見出しおよび脚注を作成することができます。
- 多くの WebFOCUS スタイルシートオプション (例、条件付きスタイル) を使用して、グラフの外観をカスタマイズすることができます。
- リクエスト内の句の順序は、グラフのフォーマットには影響しません。たとえば、選択句は、表示コマンドおよびソート句の前後いずれにも配置することができます。ソート句の順序は、TABLE リクエスト内のソート句の順序がレポートの表示に影響するように、グラフのフォーマットに影響します。
- リクエストを終了するには、「END」が 1 行に単独で記述されている必要があります。
- すべての日付は、文字フィールドに変更されない限り、MDY 形式で表示されます。

GRAPH と TABLE の相違点

GRAPH および TABLE には、いくつかの構文上の相違点があります。特に、次のような制限が適用されます。

- GRAPH リクエストでは、表示コマンドで少なくとも 1 つの表示フィールドを指定する必要があります。ほとんどの場合、意味のあるグラフを生成するには、少なくとも 1 つのソート句 (BY または ACROSS) を指定する必要があります。
- GRAPH リクエストでは、表示コマンドの目的語は、常に数値フィールド (メジャー) にする必要があります。
- 1 つのリクエストには BY 句を複数使用することができます。この場合、複数のグラフが作成されます。1 つの GRAPH リクエストで使用できる ACROSS 句は 1 つのみです。特定のグラフ形式のリクエストには、ACROSS 句と BY 句の両方を含めることができます。
- ACROSS 値の数は、64 以下でなければなりません。
- RUN オプションを END の代用として利用することはできません。

サンプルファイルの生成

このマニュアルの例の中には、WF_RETAIL サンプルの一部である WF_RETAIL_LITE データソースを使用するものがあります。また、例によっては、レガシーデータベースや、機能に特化したデータを使用するものもあります。WebFOCUS Retail サンプルファイルを生成するには、次の手順を実行します。

1. Reporting Server ブラウザインターフェースに移動します。
2. [アプリケーション] ページで、[新規] ボタンをクリックして [チュートリアル] を選択するか、アプリケーションフォルダを右クリックして [新規] を選択し、コンテキストメニューから [チュートリアル] を選択します。

   [チュートリアルフレームワークの作成] ページが開きます。
3. [チュートリアル] ドロップダウンリストから [WebFOCUS - Retail デモ] を選択します。

   WebFOCUS Retail サンプルファイルは、リレーショナル DBMS が構成されていない場合は Flat File として生成し、リレーショナル DBMS が構成されている場合はリレーショナルテーブルとして生成することができます。リレーショナルテーブルを生成する場合は、サンプルファイルを生成する前にサンプルファイルを受容するためのデータベースをリレーショナル DBMS に作成する必要があります。
4. リレーショナルアダプタが構成されていない場合は、DBMS として DATREC を選択します。リレーショナルアダプタが構成されている場合は、リレーショナル DBMS および接続を選択します。
5. DBMS で作成された各テーブル名に適用する接頭語を入力するか、入力済みのデフォルト名を受容します。
6. 大量のデータ生成を回避するためには、[チュートリアルデータのデータ量制限] ドロップダウンリストから [中] を選択することができます。

   注意：このマニュアルに使用したサンプル出力結果は、データ量 [中] で生成されたものです。
7. アプリケーション名を入力するか、入力済みのデフォルト名を受容します。また、[...] (参照) ボタンをクリックして、アプリケーションツリーからアプリケーションを選択することもできます。
8. [作成] をクリックします。

サンプルファイルが作成されます。

レガシーサンプルファイルを生成するには、[チュートリアルフレームワーク] ページで、[レガシーサンプルテーブルとファイルの作成] を選択します。次に、ファイルを生成するアプリケーション (デフォルトは ibisamp) を選択し、[作成] を選択します。

注意：このマニュアルの一部のサンプルリクエストでは、ページ幅に制限があるため、文字列が分割されて次の行に折り返されている場合があります。このリクエストをコピーして実行するには、文字列内の改行を削除する必要があります。 つまり、文字列を囲む先頭の引用符と末尾の引用符の両方がリクエストの同一行に記述されている必要があります。

2

グラフタイプの理解

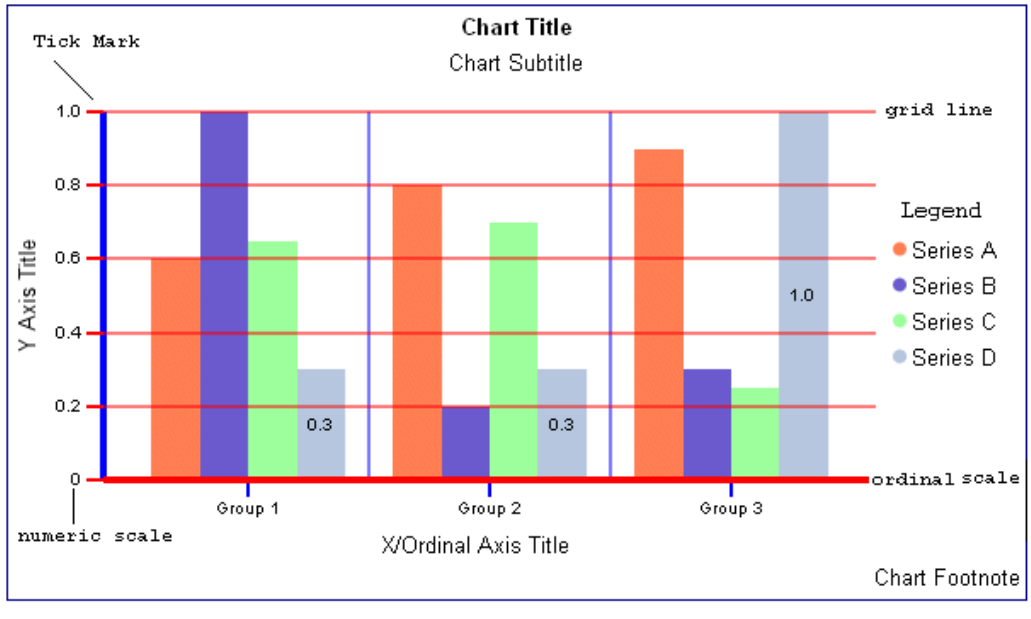
ここでは、HTML5 グラフの作成に使用可能なグラフタイプ、および HTML5 グラフの生成方法について説明します。

トピックス

- グラフのコンポーネント
- グラフタイプの制御
- グラフタイプの注意点およびサンプルグラフ

グラフのコンポーネント

下図は、グラフのコンポーネントを示しています。



一部の例外を除いて、リクエストで指定された各フィールドの一連の値はシリーズとして表され、各ソートフィールド値に関連するシリーズ値すべてがグループとして表されます。

グラフタイプの制御

グラフタイプを設定するには、リクエストに LOOKGRAPH パラメータを含める方法をお勧めします。

WebFOCUS には、このパラメータ以外にも多くのグラフプロパティを制御するパラメータが用意されています。これらのパラメータの大部分には、同等の JSON プロパティがあります。最良の方法は、すべての状況で WebFOCUS 構文を可能な限り使用し、特定のプロパティで WebFOCUS パラメータが存在しない場合は JSON プロパティを使用することです。

LOOKGRAPH によるグラフタイプの制御

一部の例外を除いて、GRAPH リクエストにはソート句 (ディメンション) と、少なくとも 1 つの表示フィールド (メジャー) を含める必要があります。

グラフに使用するフィールドは、実フィールドにすることも、一時項目 (DEFINE) にすることもできます。また、フィールドに演算接頭語 (例、AVE.、MIN.、MAX.) を使用することも可能です。また、一時項目 (COMPUTE) をフィールドとして使用することもできます。

注意：計算のみに使用される表示フィールドはグラフに表示する必要はありません。このようなフィールドを非表示にするには、NOPRINT 句を使用します。

また、LOOKGRAPH パラメータにより、個々の制御パラメータの設定や GRAPH リクエストの再構築を行うことなく、グラフフォーマットを変更することができます。ただし、LOOKGRAPH パラメータを使用する場合でも、個々の制御パラメータを設定 (例、SET GRID=ON) することができます。

注意：HTML5 グラフの生成には、従来の構文およびグラフ属性構文の 2 つのタイプの構文を使用することができます。2 つの構文で異なる LOOKGRAPH 値を使用する場合もあります。この章では、従来の構文に使用する LOOKGRAPH 値について説明します。グラフ属性構文で使用可能なグラフタイプおよび LOOKGRAPH 値についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

構文　LOOKGRAPH によるグラフタイプの指定

リクエストの前では、次の構文を使用します。

```
SET LOOKGRAPH= type
```

リクエスト内では、次の構文を使用します。

```
ON GRAPH SET LOOKGRAPH type
```

説明

`type`

生成するグラフタイプです。

次のセクションでは、HTML5 グラフの作成時によく使用する LOOKGRAPH 値について説明します。

参照 折れ線グラフのスタイル

グラフ属性構文を使用した折れ線グラフの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、よく使用する折れ線グラフタイプの従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
`VLINE` `LINE`	縦絶対折れ線グラフです。
`VLINE2`	縦 2 軸絶対折れ線グラフです。
`HLINE`	横絶対折れ線グラフです。
`HLINE2`	横 2 軸絶対折れ線グラフです。

参照 棒グラフのスタイル

グラフ属性構文を使用した棒グラフの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、よく使用する棒グラフタイプの従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
VBAR BAR	縦絶対棒グラフです (縦棒を左右に並べたグラフ)。
HBAR	横絶対棒グラフです (横棒を上下に並べたグラフ)。
VBRSTK1 STACK	縦積み上げ棒グラフです。
VBRSTKPC	縦パーセント棒グラフです。
HBRSTK1	横積み上げ棒グラフです。
HBRSTKPC	横パーセント棒グラフです。

参照　円グラフのスタイル

グラフ属性構文を使用した円グラフの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、よく使用する円グラフタイプの従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
PIE	一般的な円グラフです。

参照　散布図のスタイル

グラフ属性構文を使用した散布図の生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、よく使用する散布図タイプの従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
SCATTER	一般的な散布図です。数値ソートフィールドが X 軸になります。
SCATTERN	任意のデータタイプのソートフィールドを使用して散布図を生成します。2 つのメジャーが必要です。各メジャーフィールドがそれぞれ軸になり、ソートフィールド値がポイントとして描画されます。
SCATTERS	任意のデータタイプのソートフィールドを使用して散布図を生成します。任意の数のメジャーを使用できます。各マーカーは、いずれかのメジャーの値を表し、ソートフィールドが X 軸になります。

参照 **3D グラフのスタイル**

下表は、よく使用する 3D グラフタイプの LOOKGRAPH 値です。

SET LOOKGRAPH	説明
3DAREAG	3D 連続グループ面グラフです。
3D_BAR	一般的な 3D 棒グラフです。
3DSURFCE	3D 表面グラフです。すべてのデータ点がうねる波のような 3D 表面で表示されます。

参照 **面グラフのスタイル**

グラフ属性構文を使用した面グラフの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、面グラフの従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
VAREA	縦絶対値面グラフです。
VAREASTK	縦積み上げ面グラフです。
VAREAR2	縦 2 極絶対値面グラフです。
VARESTK2	縦 2 極積み上げ面グラフです。
VARESTKP	縦パーセント面グラフです。
HAREA	横絶対値面グラフです。
HAREAR2	横 2 極絶対値面グラフです。
HAREASTK	横積み上げ面グラフです。
HARESTK2	横 2 極積み上げ面グラフです。
HARESTKP	横パーセント面グラフです。

参照

極グラフのスタイル

極グラフのスタイルを変更するには、次の LOOKGRAPH 値のいずれかを選択します。

SET LOOKGRAPH	説明
POLAR	XY 極グラフ (分散極グラフ) です。
POLAR2	XY 極 2 軸グラフ (2 Y 極グラフ) です。2 つ目のデータセットは、第 2 の値 (Y) 軸上に表示されます。

参照 レーダーグラフのスタイル

レーダーグラフのスタイルを変更するには、次の LOOKGRAPH 値のいずれかを選択します。

SET LOOKGRAPH	説明
RADARA	レーダー面グラフ (円形面グラフ) です。
RADARL	レーダー折れ線グラフ (円形折れ線グラフ) です。
RADARL2	レーダー 2 軸グラフです。2 つ目のデータセットは、第 2 の値 (Y) 軸上に表示されます。

参照 バブルグラフのスタイル

グラフ属性構文を使用したバブルグラフの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、バブルグラフの従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
BUBBLE	バブルグラフです。
BUBBLED	2 軸バブルグラフです。
BUBBLEDL	2 軸ラベル付きバブルグラフです。
BUBBLEL	ラベル付きバブルグラフです。

参照 ブロック地図のスタイル

グラフ属性構文を使用したヒートマップの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、ブロック地図 (ヒートマップ) を生成する従来の LOOKGRAPH 値です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
SPECTRAL	ブロック地図グラフです。行または列マーカーがデータ値によって色分けされて表示されます。

参照 その他のグラフタイプ

グラフ属性構文を使用したメータグラフの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

高度なグラフタイプには、次の従来の LOOKGRAPH 値を使用します。LOOKGRAPH GAUGE1 を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
GAUGE1	メータグラフです。メータグラフでは、一般に円内に表示される軸で単一値を識別します。
VWATERFL	縦滝型グラフです。滝型グラフにはソートフィールドは必要ありません。
HWATERFL	横滝型グラフです。滝型グラフにはソートフィールドは必要ありません。
PARETO	パレートの 80:20 規則に従うデータを表示します。パレートグラフは、1 つの表示フィールドのみが必要です。
MULTI3Y MULTI4Y MULTI5Y	グラフの判読、分析、管理を容易にするために複数のグラフを積み重ねます。

参照

HTML5 固有のグラフタイプ

グラフ属性構文を使用したマップグラフ、ストリームグラフ、タグクラウド、ツリーマップの生成についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

次の従来の LOOKGRAPH 値は、HTML5 グラフを生成する場合にのみ有効です。次の LOOKGRAPH 値を使用した従来のグラフリクエストからグラフ属性構文への変換についての詳細は、1061 ページの 「 リクエストのグラフ属性構文への変換 」 を参照してください。

SET LOOKGRAPH	説明
BUBBLEMAP	バブルマップは、値に比例したサイズのバブルが地図上の関連する領域に表示されるグラフです。
CHOROPLETH	コロプレスマップは、表示するメジャーの値に応じて地図上の領域が濃淡やパターンで区分されたグラフです。
MEKKO	マリメッコチャートは、積み上げ棒グラフの一種で、棒の幅がデータセットの値に比例して調整されます。
PARABOX	パラボックス (並行座標プロット) は、標準の折れ線グラフに類似していますが、折れ線グラフの各グループにそれぞれ固有のインタラクティブな数値軸が使用される点が異なります。各折れ線は、データのシリーズを表します。各縦棒は、数値軸を表します。それぞれの軸上をクリックし、ドラッグして値の範囲を指定することで、軸のその部分を通過する線を選択 (フィルタ) することができます。
STREAM	ストリームグラフは、積み上げ面グラフを簡素化したグラフです。ストリームグラフでは、軸、罫線、フレームが省略されます。ベースラインが固定されていないため、データ全体で特定のレイヤの厚みを簡単に把握することができます。
TAGCLOUD	タグクラウドは、発生頻度をグラフ化したものです。このグラフには、グループのラベルのみが表示されます。データ値に比例して、各ラベルのサイズが変化します。

SET LOOKGRAPH	説明
TREEMAP	ツリーマップには、階層データが、ネストされた長方形の集合体として表示されます。

グラフタイプの注意点およびサンプルグラフ

ここでは、従来の WebFOCUS グラフ構文を使用した簡単な WebFOCUS リクエストで、さまざまなグラフタイプを生成します。各グラフタイプで使用可能な JSON メソッドおよびプロパティについては、後続の章で説明します。グラフタイプおよびグラフ属性構文が使用可能なサンプルグラフについての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

ヒストグラム

ヒストグラムは、データ分布を視覚的に表現したグラフです。一般に棒グラフのソートフィールドにはデータを分類するフィールドが使用されますが、ヒストグラムのソートフィールドは数値です。

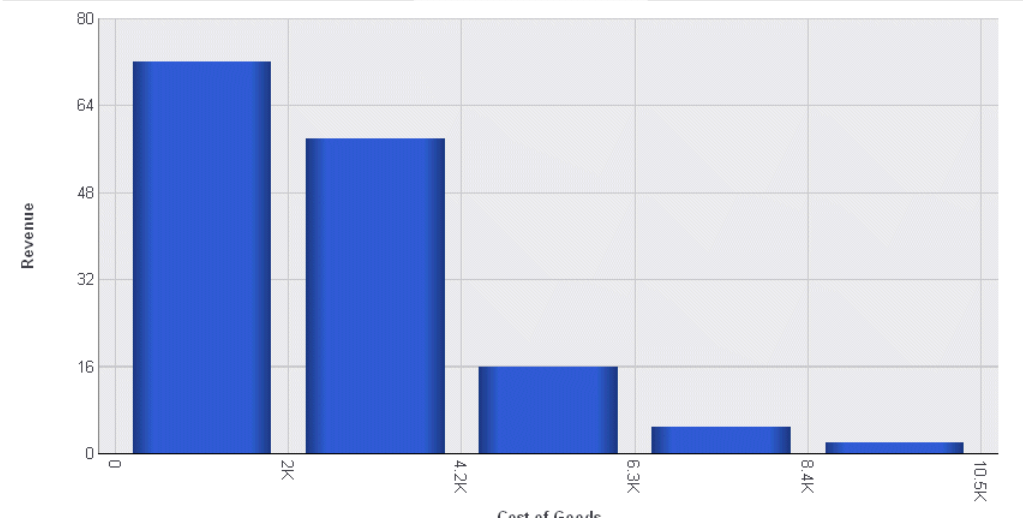
縦ヒストグラムでは、グラフの左側に Y 軸、グラフの下側に X 軸が描画されます。 横ヒストグラムでは、グラフの左側に X 軸、グラフの下側に Y 軸が描画されます。

例　ヒストグラムの例

次のリクエストは、縦ヒストグラムを生成します (ON GRAPH SET LOOKGRAPH VHISTOGR)。リクエストのスタイルセクションの JSON ブロックで、使用する X 軸グループ数を設定し、凡例を非表示にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY COGS_US
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VHISTOGR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"histogramProperties": {"binCount": 5},
"legend": {"visible": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



パラボックス

パラボックス (並行座標プロット) は、動的なクエリを使用して高次元のデータを視覚化する際によく使用されます。パラボックスは、標準の折れ線グラフに類似していますが、折れ線グラフの各グループにそれぞれ固有のインタラクティブな数値軸が使用される点が異なります。

色分けされた各折れ線は、データのシリーズを表します。各縦棒は、数値軸を表します。それぞれの軸上をクリックし、ドラッグして値の範囲を指定することで、軸のその部分を通過する線を選択 (フィルタ) することができます。

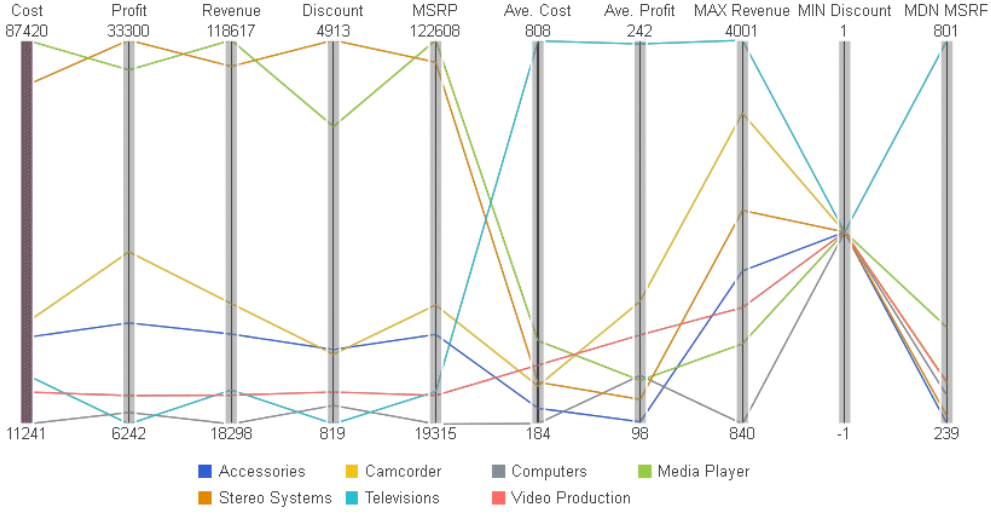
FORMAT JSCHART では、カテゴリ (文字) 縦軸を使用することもできます。これらは縦軸上にバブルマーカーで描画されます。バブルのサイズは、バブルが通過する線の数に一致します。数値軸になるかカテゴリ軸になるかは、データの 1 番目のシリーズによって定義されます。1 番目の配列の値が数値であれば、軸は数値軸になります。1 番目の配列の値が文字列であれば、軸はカテゴリ軸になります。カテゴリ軸は自動的にソートされるため、バブルは上からサイズの大きい順に表示されます。

例　　パラボックスの例

次のリクエストは、パラボックスを作成します (ON GRAPH PCHOLD FORMAT PARABOX)。スタイルセクションの JSON ブロックで、凡例を表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US AS 'Cost'
GROSS_PROFIT_US AS 'Profit'
REVENUE_US DISCOUNT_US MSRP_US
AVE.COGS_US AS 'Ave. Cost'
AVE.GROSS_PROFIT_US AS 'Ave. Profit'
MAX.REVENUE_US
MIN.DISCOUNT_US
MDN.MSRP_US AS 'MDN MSRP'
BY PRODUCT_CATEGORY
ON TABLE PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PARABOX
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": true}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



パレートグラフ

パレートグラフは複合グラフに類似しています。1 つ目のシリーズ (シリーズ 0) は、Y 軸上の絶対棒グラフとして描画されます。2 つ目のシリーズ (シリーズ 1、シリーズ 0 と同一のフィールドを使用して取得) は、Y2 軸上に積み上げパーセント折れ線グラフとして描画されます。これにより、各グループの絶対量を確認できるだけでなく、合計に対する割合が最大になるグループを特定することができます。

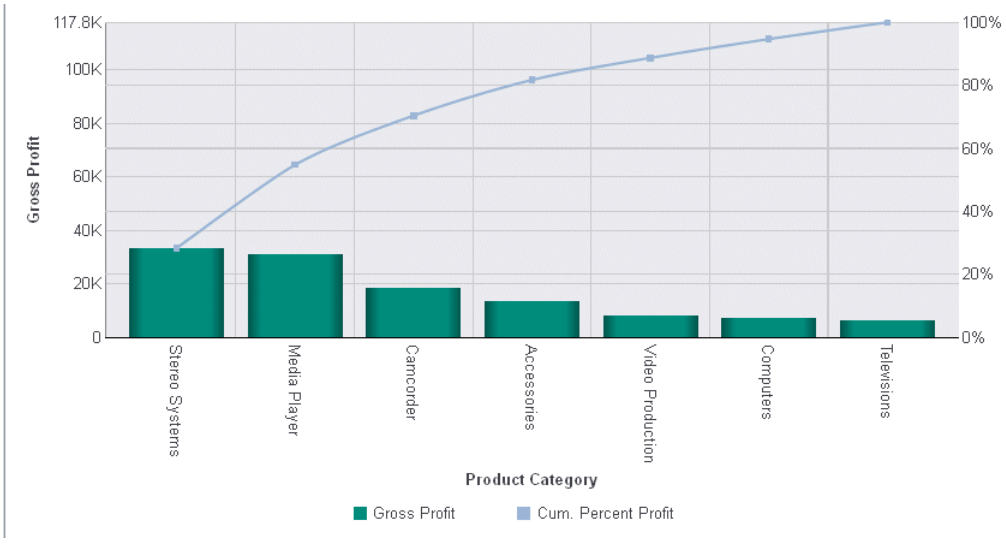
注意：パレートグラフでは、Y1 軸の最大値が全ライザの合計値に設定されます。Y 軸の最大値はカスタマイズしないことをお勧めします。設定は、Automatic (デフォルト設定) のままにしておきます。

例　パレートグラフの例

次のリクエストは、パレートグラフを作成します (ON GRAPH SET LOOKGRAPH PARETO)。棒グラフと折れ線グラフの両方は、単一の表示フィールド (GROSS_PROFIT_US) から生成されます。リクエストのスタイルセクションの JSON ブロックで、シリーズの色、マーカー形状、ラベルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON TABLE PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PARETO
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"mouseOverIndicator": {
    "enabled": true, "color": " "},
"legend": {"visible": true},
"blaProperties": {"orientation": "vertical", "lineConnection": "curved"},
"series": [
    {"series": 0, "color": "rgb(0,142,126)", "marker": {"visible": true}},
    {"series": 1, "color": "rgb(152,181,211)", "marker": {"visible": true},
            "label": "Cum. Percent Profit"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



極グラフ

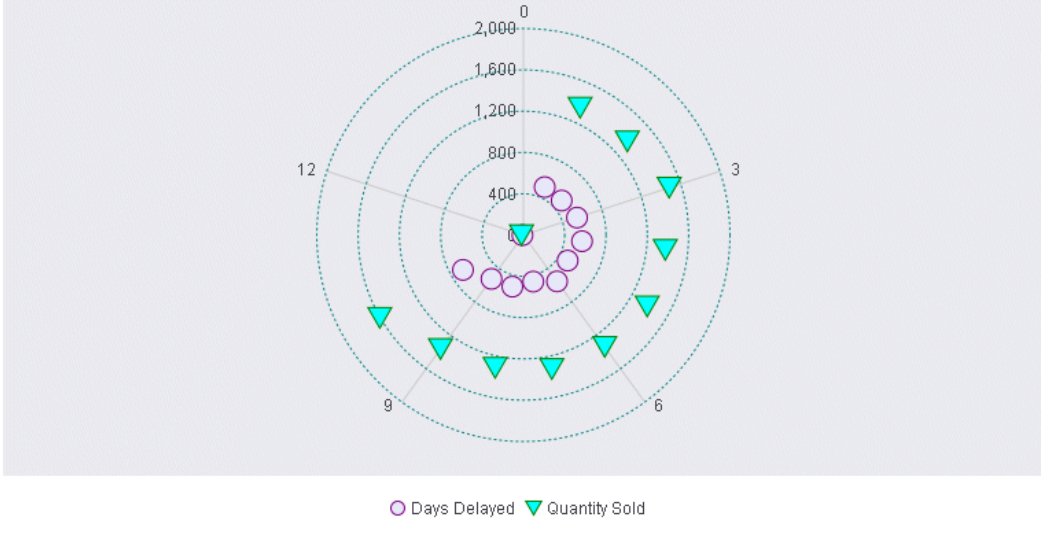
極グラフは、円形の散布図です。散布図と同様に、極グラフには各マーカーを描画するための値が 2 つ必要です。

例　極グラフの例

次のリクエストは、極グラフを作成します (ON GRAPH SET LOOKGRAPH POLAR)。スタイルセクションの JSON ブロックで、シリーズの色および Y 軸主罫線の線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
ACROSS TIME_MTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH POLAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"yaxis": {"majorGrid": {"lineStyle": {"width": 1, "color": "teal", "dash": "2 2"}}},
"series": [
{"series": 0, "color": "lavender", "marker": {"size": 15, "shape": "circle",
    "border": {"width": 1,"color": "purple"}}},
{"series": 1, "color": "cyan", "marker": {"size": 15, "shape": "triangle",
    "border": {"width": 1, "color": "green"}}}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



レーダーグラフ

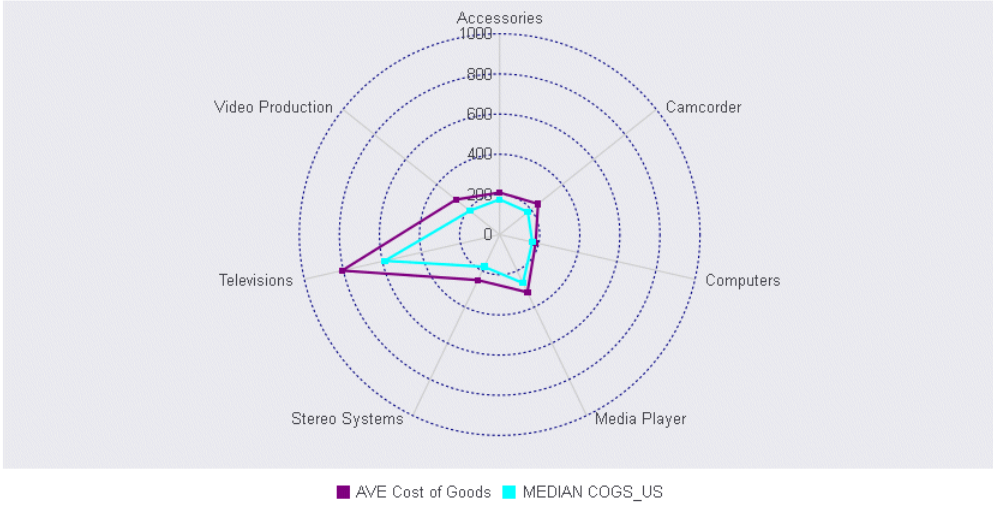
レーダーグラフは、円形の折れ線グラフです。レーダーグラフには、線セグメント (マーカーを含む) のそれぞれに値が 1 つ必要です。

例　レーダーグラフの例

次のリクエストは、レーダーグラフを作成します (ON GRAPH SET LOOKGRAPH RADARL)。スタイルセクションの JSON ブロックで、Y 軸主罫線の色とスタイル、およびシリーズの色と境界線を設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH RADARL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": true},
"yaxis": {"majorGrid": {"lineStyle": {
    "width": 1, "color": "navy", "dash": "2 2"}}},
"series": [
    {"series": 0, "color": "purple", "border": {"width": 2}},
    {"series": 1, "color": "cyan", "border": {"width": 2}}
    ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



3D 面グラフ

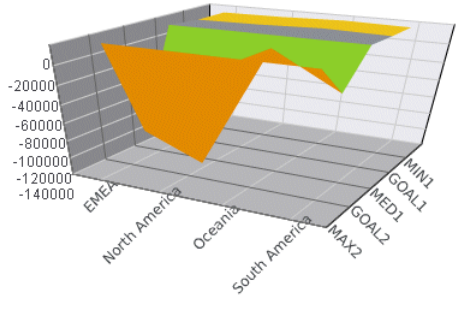
3D 面グラフには、一連のデータポイントが接続された 3 次元の面が表示されます。グラフ上で色分けされた色は、異なるシリーズを表すのではなく、類似した値を示します。

例　3D 面グラフの例

次のリクエストは、3D 面グラフを作成します (ON GRAPH SET LOOKGRAPH 3DSURFCE)。スタイルセクションの JSON ブロックで、凡例を非表示にします。その他すべてのプロパティでは、デフォルト値を使用します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF = GROSS_PROFIT_US - REVENUE_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF NOPRINT AND COMPUTE
MIN1 = MIN.DIFF;
GOAL1 = MIN1 +200;
MED1 =MDN.DIFF;
GOAL2 =DIFF;
MAX2 = MAX.DIFF;
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH 3DSURFCE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



滝型グラフ

滝型グラフは、初期値に正の値または負の値を連続的に加算することで、累積結果を視覚的に表現したグラフです。初期値および最終値 (合計値) は、X 軸のベースラインから伸びたライザの長さで表されます。初期値から最終値までの正の値および負の値は、浮動ライザとして表されます。

滝型グラフにはソート句は必要ありません。

例　滝型グラフの例

次のリクエストは、滝型グラフを作成します (ON GRAPH SET LOOKGRAPH VWATERFL)。スタイルセクションの JSON ブロックで、ライザ色を設定し、合計のライザを生成します。また、接続線のプロパティを設定します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
DECR1 = (COGS_US - 1000);
DECR2 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 AS 'Qtr1'
INCR2 AS Qtr2'
DECR1 AS 'Qtr3'
DECR2 AS 'Qtr4'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"waterfallProperties": {
    "appendTotalRiser": true,
    "positiveRiserColor": "#77b39a",
    "negativeRiserColor": "#e2675b",
    "zeroRiserColor": "#7593bd",
    "connectorLine": {
        "width": 1,
        "color": "red",
        "dash": ""}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

ブレットグラフ

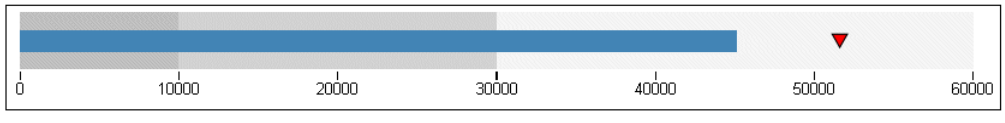
ブレットグラフは、棒グラフの一種のマイクログラフで、メータグラフの代わりとして使用されることがあります。ブレットグラフは、定性的な範囲で定量的な測定値を比較することを目的にしたグラフです。たとえば、利益や売上などの定量的な測定値を定性的な指標 (良好、十分、不十分) と比較することで、目標値に対する進捗度を表すことができます。

例　ブレットグラフの例

次のリクエストは、ブレットグラフを生成します。このグラフには、土曜日と日曜日の売上原価を表示します。ブレットグラフの LOOKGRAPH 値は存在しないため、LOOKGRAPH 値を CUSTOM に設定し、スタイルセクションの JSON ブロックで "chartType": "bullet" プロパティを指定します。また、Y 軸の色バンドおよび各曜日 (グループ) の色を設定します。グラフの高さは、VAXIS パラメータで設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'SAT' OR 'SUN'
ON GRAPH SET VAXIS 80
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CUSTOM
ON GRAPH SET STYLE *
*GRAPH_JS
"chartType": "bullet",
"yaxis": {"colorBands": [
    {"start": 0, "stop": 10000, "color": "silver"},
    {"start": 10000, "stop": 30000, "color": "lightgrey"},
    {"start": 30000, "stop": 60000, "color": "whitesmoke"}]},
"series": [
    {"series": 0, "group": 0, "color": "steelblue"},
    {"series": 0, "group": 1, "color": "red"}
    ]
*END
ENDSTYLE
END
```

出力結果では、土曜日の値が青色の棒、日曜日の値が赤色の四角形で表示されています。



スパークライングラフ

スパークライングラフは、タイトル、ラベル、凡例が省略されたマイクログラフです。スパークライングラフは、軸や座標を使用せずに、値の変化を示す輪郭を小さい領域に描画することを目的にした単一の折れ線グラフです。スパークライングラフは、テキストやテーブルに埋め込むことができます。

例　スパークライングラフの例

次のリクエストは、スパークライングラフを生成します。スパークライングラフの LOOKGRAPH 値は存在しないため、LOOKGRAPH 値を CUSTOM に設定し、スタイルセクションの JSON ブロックで "chartType": "sparkline" プロパティを指定します。ここで境界線の色と幅、シリーズ (線) の色も設定できます。グラフの高さおよび幅は、GRAPH パラメータの HAXIS および VAXIS で設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
ACROSS PRODUCT_CATEGORY
ON GRAPH SET HAXIS 50
ON GRAPH SET VAXIS 20
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CUSTOM
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"chartType": "sparkline",
"series": [{"series": 0, "color": "red"}],
"border": {"width": 2, "color": "green"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

3

HTML5 グラフの JSON プロパティの概要

ここでは、基本的な JSON 構文、色とグラデーションの定義、数値フォーマットについて説明します。

トピックス

- TIBCO WebFOCUS HTML5 JSON 構文の基礎
- 色およびグラデーション
- フォントの定義
- 数値のフォーマット設定
- 文字列内での HTML コードの使用
- データ定義
- グラフのプロパティ値の設定
- ラベル値を返すコールバック関数の定義
- グラフテンプレートエンジンを使用したツールヒント、タイトル、データラベルのカスタマイズ

TIBCO WebFOCUS HTML5 JSON 構文の基礎

JSON は、属性と値の組み合わせをテキスト形式で記述することで、メソッドおよびプロパティを定義するオープン標準フォーマットです。JSON は、特定の言語に依存しません。

参照 JSON オブジェクトの定義

オブジェクトを定義するには、オブジェクトのプロパティを定義する名前と値の組み合わせを順不同で記述します。

各オブジェクトの名前の後にコロン (:) を追加し、次にプロパティのリストを記述します。このリストは、左中括弧 ({) と右中括弧 (}) で囲みます。

各プロパティは、名前と値の組み合わせで定義します。たとえば、オブジェクトの色を定義するプロパティがあります。このプロパティの名前は「color」です。このプロパティの値で色を指定します。色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

値は、数値、文字列、ブール値 (true または false)、NULL 値、オブジェクト、配列のいずれかで指定することができます。文字列値は、一重引用符 (') または二重引用符 (") で囲む必要があります。

注意：厳密な JSON 構文では、プロパティ名、オブジェクト名、および文字値のすべてを二重引用符 (") で囲む必要があります。WebFOCUS マップグラフのリクエストにはすべて、厳密な JSON 構文が必要です。

リクエストでは、1 つの文字列値を複数の行に分割して記述することはできません。以下はその例です。

```
"color": "green"
```

注意：このマニュアルの一部のサンプルリクエストでは、行の長さに制限があるため、文字列が分割されて次の行に折り返されている場合があります。このリクエストをコピーして実行するには、文字列内の改行を削除する必要があります。

各プロパティを区切るには、カンマ (,) を使用します。各オブジェクト定義を区切る場合にも、カンマ (,) を使用します。

オブジェクトには、配列を含めることができます。たとえば、グラフには、リクエストで指定された複数のフィールドをシリーズとして表示することができます。リクエストでは、各シリーズを 0 (ゼロ) から始まる番号で識別します。そのため、シリーズオブジェクトに配列を使用し、各メンバーをそれぞれ 1 つのシリーズ番号で表すことができます。配列は、左大括弧 ([) と右大括弧 (]) で囲みます。

たとえば、次のようにシリーズオブジェクトを記述して、グラフに表示する最初の 2 つのシリーズ (シリーズ 0 およびシリーズ 1) の色プロパティを定義します。数字の 0 および 1 はシリーズプロパティの値、文字列の "cyan" および "bisque" は色プロパティの値です。

```
"series":
     [
        {"series": 0, "color": "cyan"},
        {"series": 1, "color": "bisque"}
     ]
```

1 つのオブジェクトに複数のプロパティおよびオブジェクトを含めることができます。

たとえば、次のように記述して、シリーズ 0 およびシリーズ 1 の色、マーカーオブジェクト、境界オブジェクトを定義します。マーカーオブジェクトで、マーカーの形状およびサイズのプロパティを定義します。境界オブジェクトで、境界の幅および色のプロパティを定義します。

```
"series":
     [
      {"series": 0, "color": "cyan",
           "marker": {"shape": "triangle", "size": 12,
           "border": {"width": 1, "color":"blue"}}},
      {"series": 1, "color": "bisque",
           "marker": {"shape": "square", "size": 13,
           "border": {"width": 1, "color":"brown"}}}
    ]
```

名前と値の間、および名前と値の組み合わせごとにブランクを挿入する必要はありませんが、可読性を向上させるためにブランクを挿入することをお勧めします。

大括弧 ([) は独自の行に配置する必要はありませんが、独自の行に配置し、インデントを使用することで、オブジェクト定義の識別が容易になります。同様に、配列内の各オブジェクトを新しい行から開始する必要はありませんが、各オブジェクトを新しい行から開始することで、オブジェクトの特定が容易になります。

色およびグラデーション

オブジェクトの色は、色のプロパティで定義します。すべてのオブジェクトに、色および透明度の設定を割り当てることができます。領域オブジェクトおよび線オブジェクトには、色定義またはグラデーション定義を割り当てることができます。

構文　色プロパティの指定

```
"color": "string"
```

説明

`"color": "string"`

次のいずれかの値です。

- 色名 (例、"red")

  サポートされる色名のリストについては、http://www.w3.org/TR/css3-color/#svg-color を参照してください。

- 3 つの RGB 値、または 3 つの RGB 値と 1 つの透明度設定

  ```
  "rgb (r,g,b)"
  ```

  または

```
"rgba(r,g,b,a)"
```

r、g、b 値は、赤、緑、青の強度 (0 から 255) を表します。

透明度を表す a 値では、オブジェクトが背景と混合される度合いを 0.0 (完全に透明) から 1.0 (完全に不透明) の値で定義します。

たとえば、黒を指定するには「"rgb(0,0,0)"」と記述します。

- 3 つの HSL 値、または 3 つの HSL 値と 1 つの透明度設定

  ```
  "hsl(h,s,l)"
  ```

  または

  ```
  "hsla(h,s,l,a)"
  ```

  - 色相 (H) は、色相環の角度として表されます。最上部の赤は 0 (ゼロ) または 360 度、緑は 120 度、青は 240 度として定義されます。
  - 彩度 (S) は、色相の鮮やかさの度合いを表し、パーセント記号 (%) を含むパーセント値で定義します。純色は彩度が 100%、灰色は無彩色です。
  - 明度 (L) は、色相の明るさの度合いを表し、パーセント記号 (%) を含むパーセント値で定義します。白は明度 100%、黒は明度 0% です。
  - 透明度を表す a 値では、オブジェクトが背景と混合される度合いを 0.0 (完全に透明) から 1.0 (完全に不透明) の値で定義します。

- 16 進数の色値

  ```
  "#hexvalue"
  ```

  16 進数の色値は、シャープ記号 (#) で始まり、次に赤、緑、青の色値 (RGB) の組み合わせをそれぞれ 2 桁の 16 進数値で指定します。各構成要素の最小値は 0 (ゼロ) です (16 進数の 00)。最大値は 255 です (16 進数の FF)。

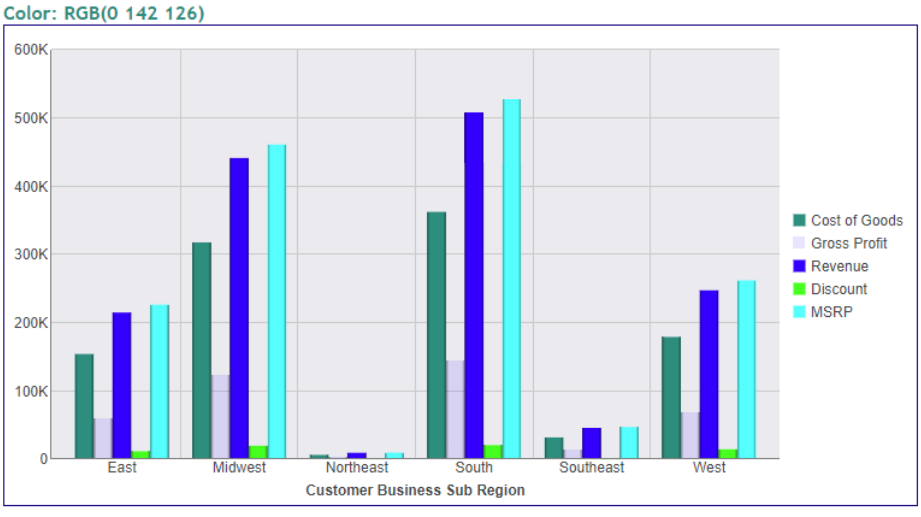
  たとえば、黒は #000000 で、これは rgb(0,0,0) に相当します。赤は #FF0000 で、これは rgb(255,0,0) に相当します。白は #FFFFFF で、これは rgb(255,255,255) に相当します。

例　色の定義

次のリクエストでは、5 つのシリーズを指定します。各シリーズの JSON 定義では、シリーズごとに異なるタイプの色定義を使用し、その色定義を凡例のシリーズラベルとして表示します。また、見出しおよび境界にも色定義が指定されています。

```
GRAPH FILE WF_RETAIL_LITE
HEADING
"Color: RGB(0 142 126)"
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US MSRP_US
BY BUSINESS_SUB_REGION
WHERE COUNTRY_NAME EQ 'United States'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": true},
"border": {"width": 1, "color": "navy"},
"series": [
   {"series": 0, "color": "rgb(0,142,126)", "label": "rgb(0,142,126)"},
   {"series": 1, "color": "hsla(240, 100%, 50%, 0.1)",
     "label": "hsla(240, 100%, 50%, 0.1)"},
   {"series": 2, "color": "hsl(240, 100%, 50%)", "label": "hsl(240, 100%, 50%)"},
   {"series": 3, "color": "#00FF00", "label": "#00FF00"},
   {"series": 4, "color": "cyan", "label": "cyan"}
   ]
*END
TYPE=HEADING, COLOR = rgb(0 142 126), JUSTIFY=LEFT,$
ENDSTYLE
END
```

出力結果は次のとおりです。



グラデーションの定義

グラデーションは、特定の色から別の色に移行する色の変化です。色の変化は、グラフ全体に線状に適用することも、中心点から外側へ放射状に適用することもできます。

グラデーションは、文字列または JSON オブジェクトとして定義することができます。

構文　　線状グラデーションのプロパティを文字列として定義

```
"color": "linear-gradient(startX, startY, endX, endY,
 stopsOffset  stopsColor,...stopsOffset  stopsColor)"
```

説明

`startX, starty`

グラデーションを開始する位置の X 軸値および Y 軸値です。

位置の X 軸値および Y 軸値は、数値 (例、0) で指定することも、文字列 (例、"5%") で指定することもできます。数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。たとえば、「start: x:0, y:0」は、オブジェクトの左上隅を表します。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。開始位置および終了位置の数値座標は、領域サイズの設定が可能なオブジェクト (例、グラフの背景領域) の場合にのみ役立ちます。たとえば、グラフの背景領域が 200 x 200 ピクセルに設定されている場合、「start x:0, y:0」および「stop x:100, y:0」を使用して線状グラデーションを定義すると、矩形の左端から中央までのグラデーションが適用されます。オブジェクトのサイズがライブラリによって計算される場合 (例、凡例領域、グラフフレーム、ライザ)、通常は文字列のパーセントを使用します。

`endX, endy`

線状グラデーションにのみ適用されます。グラデーションを終了する位置の X 軸値および Y 軸値です。

位置の X 軸値および Y 軸値は、数値 (例、20) で指定することも、文字列 (例、"20%") で指定することもできます。数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。たとえば、「start: x:0, y:0」は、オブジェクトの左上隅を表します。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。開始位置および終了位置の数値座標は、領域サイズの設定が可能なオブジェクト (例、グラフの背景領域) の場合にのみ役立ちます。たとえば、グラフの背景領域が 200 x 200 ピクセルに設定されている場合、「start x:0, y:0」および「stop x:100, y:0」を使用して線状グラデーションを定義すると、矩形の左端から中央までのグラデーションが適用されます。オブジェクトのサイズがライブラリによって計算される場合 (例、凡例領域、グラフフレーム、ライザ)、通常は文字列のパーセントを使用します。

`stopsOffset`

色の変化点を定義するオフセット値です。この値は、数値または文字列で指定します。文字列は、パーセント記号 (%) を含めて二重引用符 (") で囲みます。有効な数値は 0 (ゼロ) から 1 までの範囲で、この数値はパーセントとして処理されます。数値の 0.5 は、50% と同一です。0 (ゼロ) より小さい数値は 0 (ゼロ) として処理され、1 より大きい数値は 1 として処理されます。

`stopsColor`

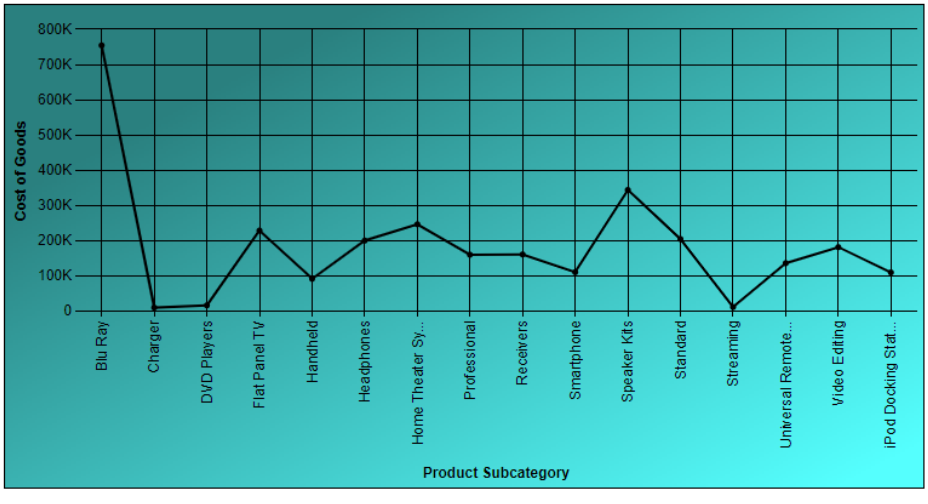
オフセットの色を指定します。

例　線状グラデーション文字列の例

次のリクエストでは、グラフの背景に線状グラデーションを指定します。開始位置および終了位置の座標でグラフ領域全体を定義し、青緑色からシアンに変化するグラデーションを適用します。線状グラデーションは文字列で指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_SUBCATEG
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
 *GRAPH_JS
"fill": {
   "color": "linear-gradient(0,0,100%,100%, 20% teal, 95% cyan)"
   },
"series": [{"series": 0, "color": "black"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　放射状グラデーションのプロパティを文字列として定義

```
"color": "radial-gradient(startX, startY, radius,
 stopsOffset stopsColor,...stopsOffset stopsColor)"
```

説明

`startX, starty`

グラデーションを開始する位置の X 軸値および Y 軸値です。

位置の X 軸値および Y 軸値は、数値 (例、0) で指定することも、文字列 (例、"5%") で指定することもできます。数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。開始位置および終了位置の数値座標は、領域サイズの設定が可能なオブジェクト (例、グラフの背景領域) の場合にのみ役立ちます。オブジェクトのサイズがライブラリによって計算される場合 (例、凡例領域、グラフフレーム、ライザ)、通常は文字列のパーセントを使用します。 放射状グラデーションの場合、「start: x/y」でグラデーションの中心を定義します。たとえば、「start: x: "50%", y: "50%"」と指定すると、オブジェクトの中心を開始位置としてグラデーションが描画されます。

`radius`

放射状グラデーションにのみ適用されます。グラデーションの中心から最外端までの距離を定義します。開始位置の X 座標と Y 座標でグラデーションの中心を定義します。たとえば、「"start": "x": "50%", "y": "50%"」と指定すると、オブジェクトの中心からグラデーションが描画されます。オブジェクトのサイズが 200 x 200 ピクセルで、グラデーションを「"x":"50%", "y":"50%"」および「"radius": "100%"」と指定すると、グラデーションの最外端は、オブジェクトの外端から 100 ピクセル外側の位置になります。この例では、200 ピクセルの 100% は 200 ピクセルになるため、実際のグラデーションは最左端から最右端まで 400 ピクセルになります。オブジェクトの中心から開始する放射状グラデーションの場合 ("start": "x":"50%"/"y":"50%")、オブジェクトの中心から外端までのグラデーションを描画するには、一般に「"radius": "50%"」を使用します。

`stopsOffset`

色の変化点を定義するオフセット値です。この値は、数値で指定することも、パーセント記号 (%) を含めた文字列で指定することもできます。有効な数値は 0 (ゼロ) から 1 までの範囲で、この数値はパーセントとして処理されます。数値の 0.5 は、50% と同一です。0 (ゼロ) より小さい数値は 0 (ゼロ) として処理され、1 より大きい数値は 1 として処理されます。

`stopsColor`

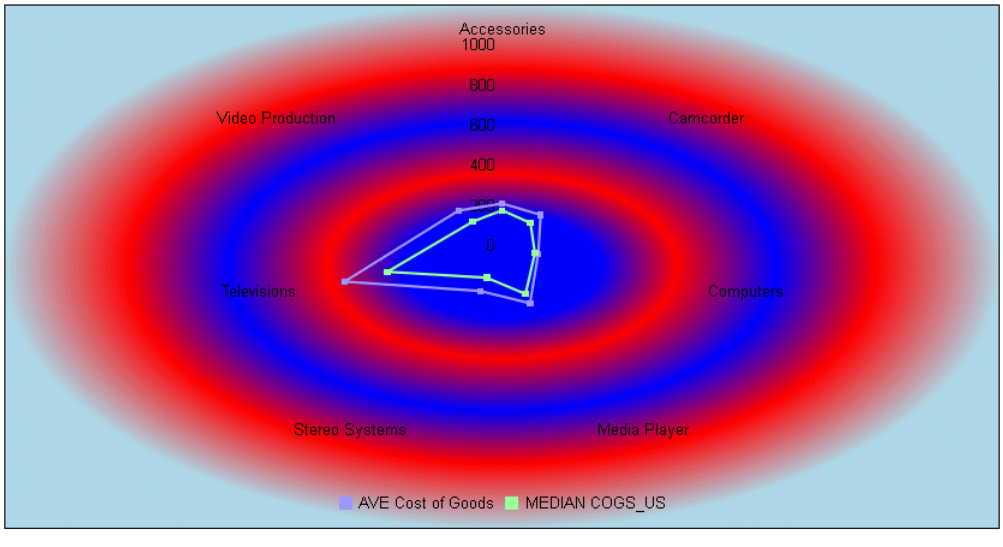
オフセットの色を指定します。

例　放射状グラデーション文字列の例

次のリクエストでは、グラフの背景に放射状グラデーションを指定します。開始位置および半径で、グラフ背景全体を定義します。色は、内側から外側へ青、赤、青、赤、水色へ変化します。グラデーションは文字列として指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH RADARL
ON GRAPH SET STYLE *
*GRAPH_JS
"fill": {
   "color":
"radial-gradient(50%,50%,50%, 20% blue, 35% red, 55% blue, 75% red, 1 lightblue)"
   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　グラデーションのプロパティを JSON オブジェクトとして定義

```
"color": {
   "type": "string",
   "start":{
      "x": sx1,
      "y": sy1   },
   "end":{
      "x": ex1,
      "y": ey1   },
   "radius": "rstring",
   "stops": [
         {
            "offset": off1,
            "color": "color1"
         },
         ...
      ]
}
```

説明

`"type": "string"`

グラデーションのタイプを定義する文字列です。有効な値は、"linear" または "radial" です。

`"start": "x", "start": "y"`

グラフに描画するグラデーションの座標を指定します。

`"x": x`

開始位置をグラフの X 座標で指定します。この値は、数値またはパーセント値で指定します。パーセント値は、パーセント記号 (%) を含めて二重引用符 (") で囲みます。

数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。たとえば、「"start": "x":0, "y":0」は、オブジェクトの左上隅を表します。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。

パーセント値の座標は、オブジェクトのサイズがライブラリによって計算される場合や放射状グラデーションの場合に役立ちます。放射状グラデーションの場合、50% がオブジェクトの中心になります。

`"y": y`

開始位置をグラフの Y 座標で指定します。この値は、数値またはパーセント値で指定します。パーセント値は、パーセント記号 (%) を含めて二重引用符 (") で囲みます。

数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。たとえば、「"start": "x":0, "y":0」は、オブジェクトの左上隅を表します。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。

パーセント値の座標は、オブジェクトのサイズがライブラリによって計算される場合や放射状グラデーションの場合に役立ちます。放射状グラデーションの場合、50% がオブジェクトの中心になります。

start: x/y および end: x/y パラメータは、数値 (例、0/20) で指定することも、文字列 (例、"5%"/"20%")で指定することもできます。数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。たとえば、「"start": "x":0/"y":0」は、オブジェクトの左上隅を表します。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。start/end 座標の数値指定は、領域サイズの設定が可能なオブジェクト (例、グラフの背景領域) の場合にのみ役立ちます。たとえば、グラフの背景領域 (描画領域) が 200 x 200 ピクセルに設定されている場合、「"start" "x":0/"y":0」および「"stop" "x": 100/"y":0」を使用して線状グラデーションを定義すると、矩形の左端から中央までのグラデーションが適用されます。オブジェクトのサイズがライブラリによって計算される場合 (例、凡例領域、グラフフレーム、ライザ)、通常は文字列のパーセントを使用します。放射状グラデーションの場合、「"start": "x"/"y"」でグラデーションの中心を定義します。たとえば、「"start": "x": "50%"/"y": "50%"」と指定すると、オブジェクトの中心からグラデーションが描画されます。

`"end": "x", "end": "y"`

グラフに描画するグラデーションの座標を指定します。

`"x": x`

線状グラデーションにのみ適用されます。終了位置をグラフの X 座標で指定します。この値は、数値またはパーセント値で指定します。パーセント値は、パーセント記号 (%) を含めて二重引用符 (") で囲みます。

数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。

パーセント値の座標は、オブジェクトのサイズがライブラリによって計算される場合や放射状グラデーションの場合に役立ちます。放射状グラデーションの場合、50% がオブジェクトの中心になります。

`"y": y`

線状グラデーションにのみ適用されます。終了位置をグラフの Y 座標で指定します。この値は、数値またはパーセント値で指定します。パーセント値は、パーセント記号 (%) を含めて二重引用符 (") で囲みます。

数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。

パーセント値の座標は、オブジェクトのサイズがライブラリによって計算される場合や放射状グラデーションの場合に役立ちます。放射状グラデーションの場合、50% がオブジェクトの中心になります。

start: x/y および end: x/y パラメータは、数値 (例、0/20) で指定することも、文字列 (例、"5%"/"20%")で指定することもできます。数値は、Raw SVG (Scalable Vector Graphics) ピクセル座標として解釈されます。たとえば、「"start": "x":0/"y":0」は、オブジェクトの左上隅を表します。負の数値は無効です。0 (ゼロ) より小さい数値は、すべて 0 (ゼロ) と解釈されます。start/end 座標の数値指定は、領域サイズの設定が可能なオブジェクト (例、グラフの背景領域) の場合にのみ役立ちます。たとえば、グラフの背景領域 (描画領域) が 200 x 200 ピクセルに設定されている場合、「start x:0/y:0」および「stop x:100/y:0」を使用して線状グラデーションを定義すると、矩形の左端から中央までのグラデーションが適用されます。オブジェクトのサイズがライブラリによって計算される場合 (例、凡例領域、グラフフレーム、ライザ)、通常は文字列のパーセントを使用します。放射状グラデーションの場合、「start: x/y」でグラデーションの中心を定義します。たとえば、「"start": "x": "50%"/"y": "50%"」と指定すると、オブジェクトの中心からグラデーションが描画されます。

`"radius": "string"`

放射状グラデーションにのみ適用されます。グラデーションの中心から最外端までの距離を定義します。

たとえば、オブジェクトのサイズが 200 x 200 ピクセルで、グラデーションを「"start": "x":"50%", "y":"50%"」および「"radius": "100%"」と指定すると、グラデーションの最外端は、オブジェクトの外端から 100 ピクセル外側の位置になります。この例では、200 ピクセルの 100% は 200 ピクセルになるため、実際のグラデーションは最左端から最右端まで 400 ピクセルになります。オブジェクトの中心から開始する放射状グラデーションの場合 ("start": "x":"50%", "y":"50%")、オブジェクトの中心から外端までのグラデーションを描画するには、一般に「"radius": "50%"」を使用します。

`"stops":`

オフセットと色の組み合わせを列記したカンマ区切りのリストです。この配列の長さに制限はなく、複数のオブジェクトまたは配列のリストにすることもできます。

`"offset": offset`

数値またはパーセントで指定するオフセットです。

数値を使用する場合、0 (ゼロ) から 1 までの数値を指定する必要があります。この数値は、色が変化する位置のグラデーション領域の割合を表します。値の 0.5 は、50% と同一です。0 (ゼロ) より小さい数値は 0 (ゼロ) として処理され、1 より大きい数値は 1 として処理されます。

パーセントを使用する場合、パーセント記号 (%) を含めてパーセント値を二重引用符 (") で囲む必要があります。このパーセント値は、色が変化する位置のグラデーション領域の割合を表します。

`"color": "string"`

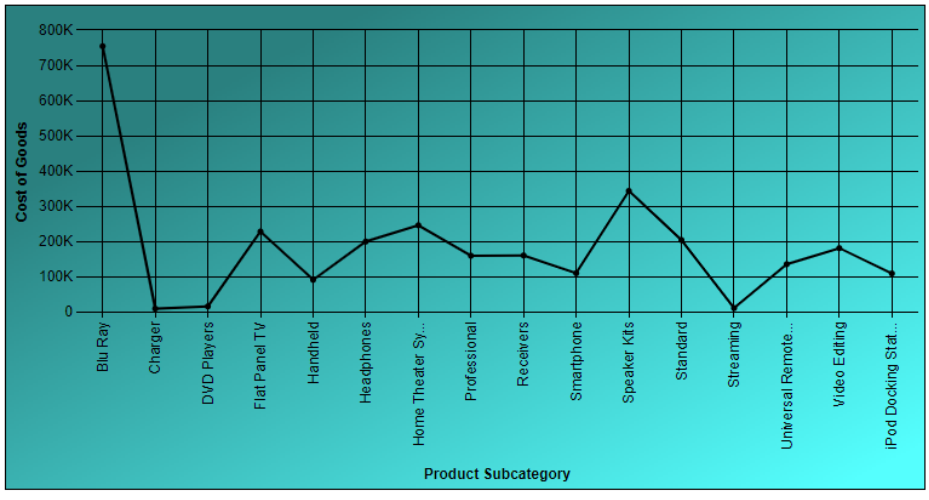
オフセットの色を指定します。

例　線状グラデーション JSON オブジェクトの例

次のリクエストでは、グラフの背景に線状グラデーションを指定します。開始位置および終了位置の座標でグラフ領域全体を定義し、青緑色からシアンに変化するグラデーションを適用します。線状グラデーションは、JSON オブジェクトとして指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_SUBCATEG
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"fill": {"color":
  {"type": "linear",
   "start": {"x": 0, "y": 0},
   "end": {"x": "100%", y: "100%"},
   "stops": [
         {"offset": "20%", "color": "teal"},
         {"offset": "95%", "color": "cyan"}
         ]
    }},
"series": [{"series": 0, "color": "black"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

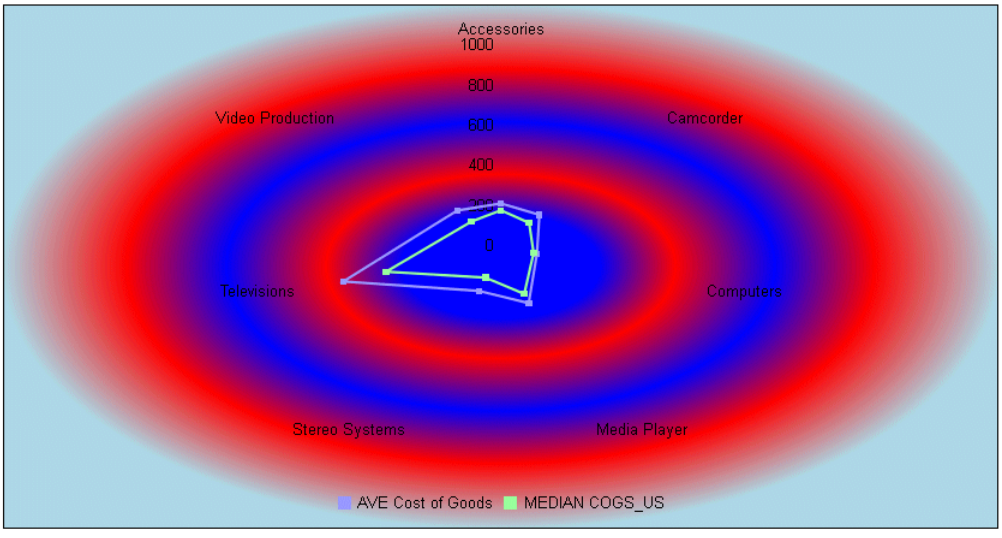


例　放射状グラデーション JSON オブジェクトの例

次のリクエストでは、グラフの背景に放射状グラデーションを指定します。開始位置および半径で、グラフ背景全体を定義します。色は、内側から外側へ青、赤、青、赤、水色へ変化します。グラデーションは、JSON オブジェクトとして指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH RADARL
ON GRAPH SET STYLE *
*GRAPH_JS
"fill": {
  "color":
   {"type": "radial",
   "start": {"x": "50%", "y": "50%"},
   "end": {"x": "50%", "y": "50%"},
   "stops": [
        {"offset": "20%", "color: "blue"},
        {"offset": "35%", "color": "red"},
        {"offset": "55%", "color": "blue"},
        {"offset": "75%", "color": "red"},
        {"offset": 1, "color": "lightblue"}
         ]
    }}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



SVG 色パターン

ライザやマーカーなどのグラフ要素を区別する場合やグラフフレームを塗りつぶす場合に、パターンを使用することができます。この方法は、色分けが困難なモノクロ環境で使用すると便利です。

構文　　SVG 色パターンの定義

```
"color":{
  "type":"pattern",
   "color": "string",
   "backgroundColor": "string",
   "shape": "string",
   "size": number,
   "pad": number}
```

説明

`"color": "string"`

色名または数値指定の文字列を使用して、パターン形状の色を定義します。

`"backgroundColor": "string"`

色名または数値指定の文字列を使用して、パターン領域の背景色を定義します。

`"shape": "string"`

パターン文字の形状を定義します。任意の SVG パス文字列およびマーカー形状のすべてがサポートされます。マーカー形状についての詳細は、467 ページの 「 シリーズマーカーのサイズ、境界線、色、形状、回転の定義 」 を参照してください。

`"size": number`

パターン形状のサイズをピクセル単位で指定します。

`"pad": number`

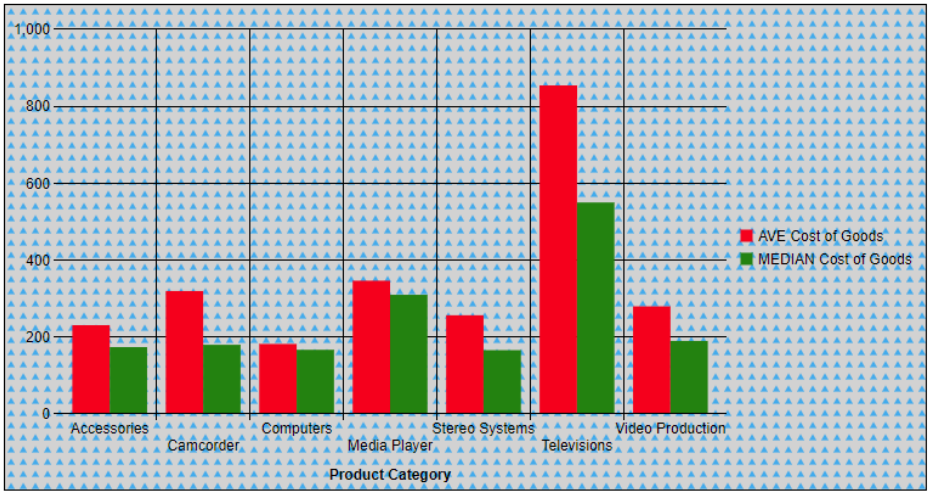
パターン形状間のパディングのサイズをピクセル単位で指定します。

例　グラフフレームを塗りつぶすパターンの使用

次のリクエストは、グラフフレームを三角形のパターン形状で塗りつぶし、パターン形状を青色、サイズを 10 ピクセルに設定します。三角形の間隔を 8 ピクセル、背景色を淡い灰色にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"fill": {
  "color":{
   "type":"pattern",
     "color": "#00aeef",
     "backgroundColor": "lightgray",
     "shape": "triangle",
     "size": 10,
     "pad": 8}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



例　パターンを定義する SVG パス文字列の使用

パス文字列は、起点からの紙面上の移動としてパターンを説明します。線は、次のコマンドを使用して記述することができます。

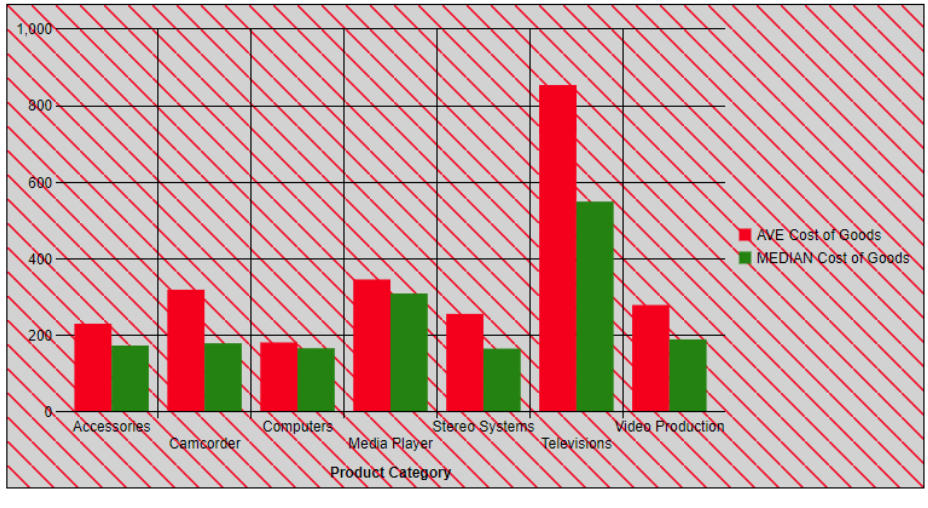
- moveto コマンド (M) で、M に続く数値の組み合わせで定義された新しい座標まで移動します。
- lineto コマンド (L) で、現在位置から新しい座標まで直線に沿って移動します。

複数の M コマンドと L コマンドが使用できます。

次のリクエストは、SVG パス文字列を使用して、角度 45 のクロスハッチパターンを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"fill": {
  "color":{
   "type": "pattern",
   "color": "red",
   "backgroundColor": "lightgray",
   "shape": "M-10,-10L20,20",
   "size": 20}
      }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

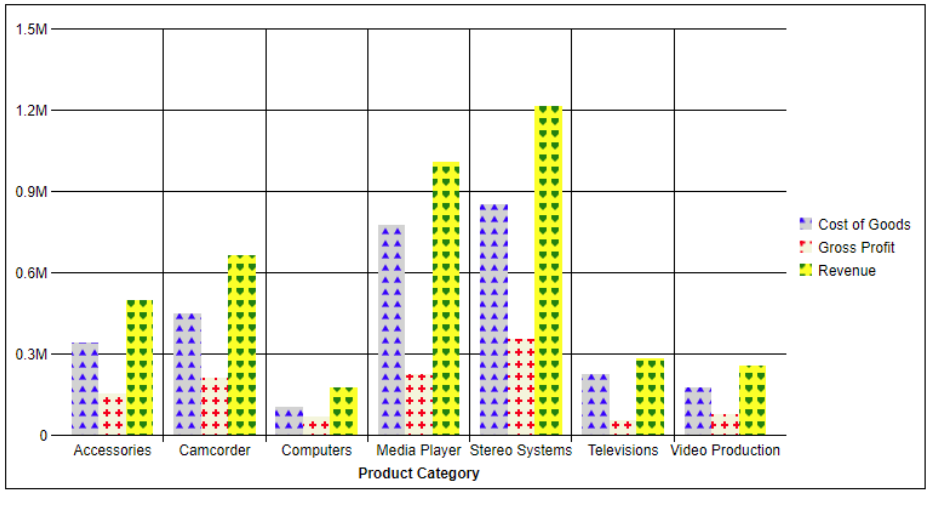


例　シリーズライザを塗りつぶすパターンの使用

次のリクエストは、縦棒グラフのライザにパターンを適用し、シリーズごとに異なる色のパターンを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
*GRAPH_JS
"series":[
 {"series":0, "color":{"type":"pattern","color": "blue","backgroundColor":
"lightgray",
     "shape": "triangle","size": 10,"pad": 8}} ,
 {"series":1, "color":{"type":"pattern","color": "red","backgroundColor":
"beige",
     "shape": "plus","size": 10,"pad": 8}} ,
 {"series":2, "color":{"type":"pattern","color": "green","backgroundColor":
"yellow",
     "shape": "house","size": 10,"pad": 8}}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

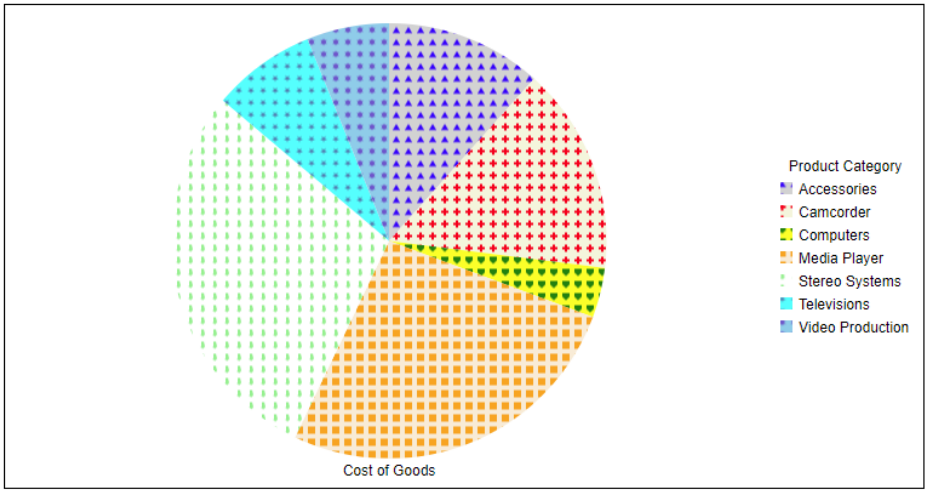


例　円項目を塗りつぶすパターンの使用

次のリクエストは、円グラフのシリーズ (円項目) にパターンを適用し、シリーズごとに異なる色のパターンを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=color,$
type=data, column=cogs_us, bucket=measure,$
*GRAPH_JS
"series": [
 {"series": 0, "color": {"type":"pattern","color":
"blue","backgroundColor": "lightgray",
     "shape": "triangle","size": 10,"pad": 8}} ,
 {"series": 1, "color": {"type":"pattern","color": "red","backgroundColor":
"beige",
     "shape": "plus","size": 10,"pad": 8}} ,
 {"series": 2, "color": {"type":"pattern","color":
"green","backgroundColor": "yellow",
     "shape": "house","size": 10,"pad": 8}},
 {"series": 3, "color": {"type":"pattern","color":
"orange","backgroundColor": "antiquewhite",
     "shape": "square","size": 10,"pad": 8}},
 {"series": 4, "color": {"type":"pattern","color":
"lightgreen","backgroundColor": "white",
     "shape": "arrow","size": 10,"pad": 8}},
 {"series": 5, "color": {"type":"pattern","color":
"steelblue","backgroundColor": "cyan",
     "shape": "fiveStar","size": 10,"pad": 8}},
 {"series": 6, "color": {"type":"pattern","color":
"slateblue","backgroundColor": "skyblue",
     "shape": "sixStar","size": 10,"pad": 8}}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

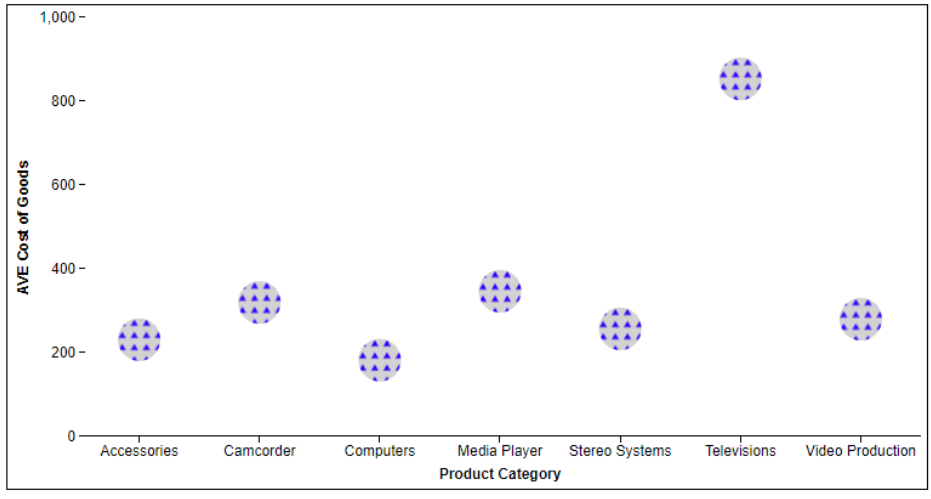


例　シリーズマーカーを塗りつぶすパターンの使用

次のリクエストは、散布図を生成します。マーカーをパターンで塗りつぶし、パターンが収まるようマーカーのサイズを変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"series": [
{"series":0, "marker":{"size":35},"color":{"type":"pattern","color":
"blue","backgroundColor": "lightgray",
    "shape": "triangle","size": 10,"pad": 8}},
],
"axisList":
{
"x1": {"majorGrid": {"visible": false}},
"y1": {"majorGrid": {"visible": false}}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



フォントの定義

フォントのプロパティはすべて、以下のサイトで定義されたフォーマットを使用して、フォント属性の文字列として指定します (例、"10pt Sans-Serif")。

http://www.w3.org/TR/CSS2/fonts.html#font-shorthand

すべてのフォントには、フォント名、サイズ、スタイル (例、normal、bold、italic)、および色のプロパティがあります。各フォントには、使用するマシンにインストールされているフォント、ブラウザで使用可能なフォント、現在の構成に応じてデフォルト値が設定されています。

注意：下線付きはサポートされません。

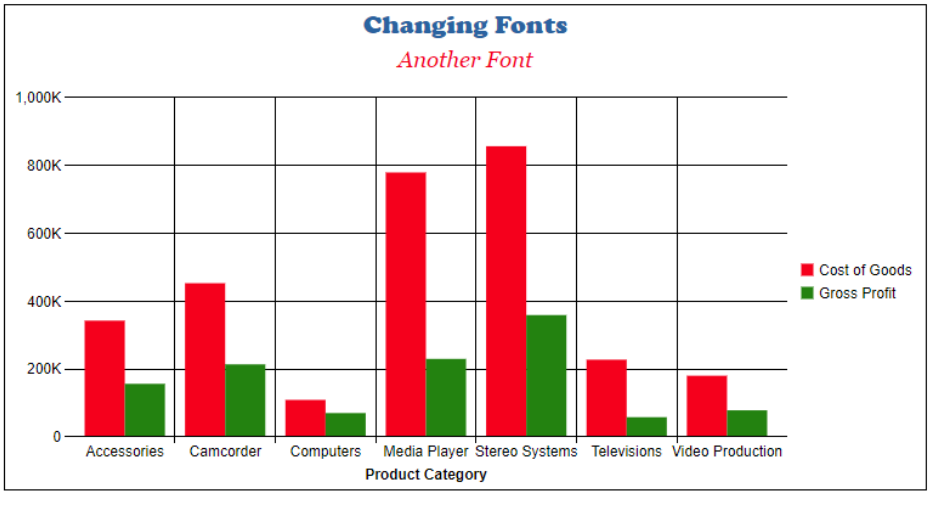
フォントを定義する際は、使用するマシンにインストールされ、ブラウザで使用可能なフォントを指定する必要があります。

例　フォントの指定

次のリクエストでは、グラフのタイトルおよびサブタイトルに使用するフォントおよびフォント特性を指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"title": {"visible": true, "text": "Changing Fonts",
   "font": "bold 16pt Cooper", "color": "rgb(0,101,163)"},
"subtitle": {"visible": true, "text": "Another Font",
   "font": "italic 14pt Georgia", "color": "red"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Google フォントの使用

SET CSSURL コマンドで Google Fonts API を指定することにより、リクエストで Google フォントを使用することができます。

利用可能な Google フォントのリストについては、http://www.google.com/fonts を参照してください。

構文　グラフリクエストでの Google フォントの使用

```
SET CSSURL='http://fonts.googleapis.com/css?family=font1|font2 ... '
```

説明

`font1|font2 ...`

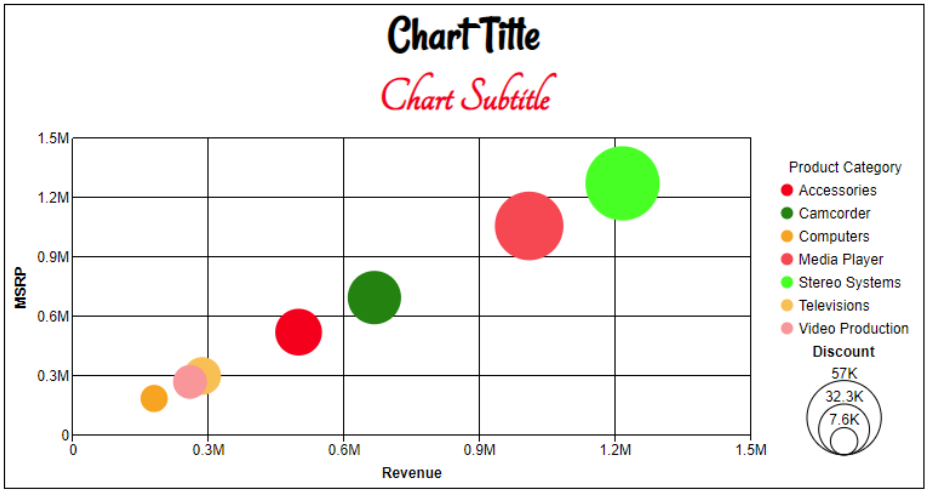
Google フォントの名前です。複数のフォント名は縦棒 (|) で区切ります。

例　グラフでの Rancho および Tangerine Google フォントの使用

次のリクエストでは、グラフのタイトルに Rancho フォント、グラフのサブタイトルに Tangerine フォントを使用します。

```
SET CSSURL='http://fonts.googleapis.com/css?family=Rancho|Tangerine'
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US MSRP_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"title": {"text": "Chart Title", "visible": true, "align": "center",
   "font": "bold 30pt Rancho", "color": "black"},
"subtitle": {"text": "Chart Subtitle", "visible": true, "align": "center",
   "font": "bold 30pt Tangerine", "color": "red"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



数値のフォーマット設定

numberFormat プロパティでは、次のいずれかを使用して数値ラベルのフォーマットを指定します。

- "auto" (自動)
- JSON オブジェクト
- "string" (フォーマット文字列)
- function()

数値フォーマットの自動設定

グラフエンジンがグラフデータに基づいて数値フォーマットを選択するよう設定するには、numberFormat: "auto" を使用します。選択されたフォーマットに接頭語および接尾語を追加するには、auto を 2 つの中括弧 ({{}}) で囲みます。たとえば、数値フォーマットの自動設定で数値の左側にチルダ記号 (~) とドル記号 ($)、右側にプラス記号 (+) を表示するには、次のように記述します。

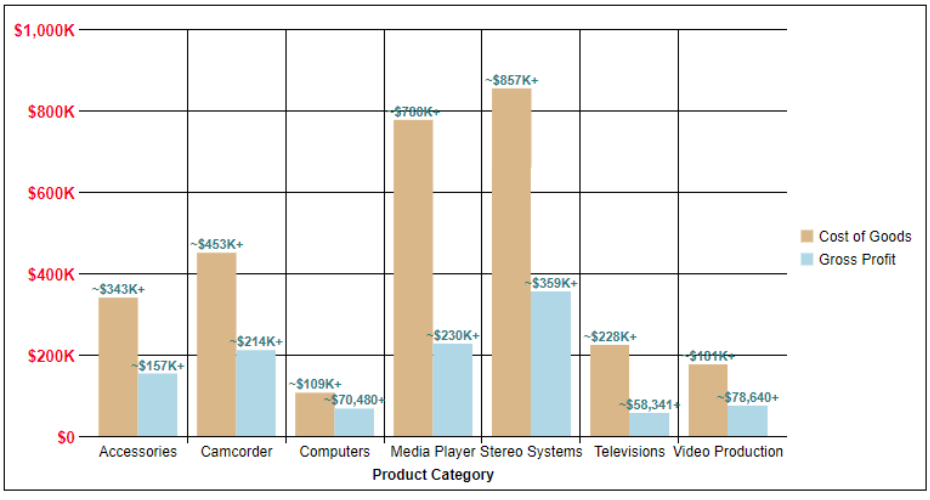
```
"numberFormat": "~${{auto}}+"
```

例　数値フォーマット自動設定の使用

次のリクエストでは、Y 軸およびデータテキストに数値フォーマットの自動設定を使用します。Y 軸の数値には接頭語としてドル記号 ($) を表示し、データテキストラベルには接頭語としてチルダ記号 (~) とドル記号 ($)、接尾語としてプラス記号 (+) を表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"axisLIst": {
"y1": {
   "labels": {"font": "bold 10pt Sans-Serif","color": "red"},
   "numberFormat": "${{auto}}"}},
"dataLabels": {"visible": true, "font": "bold 8pt Sans-Serif", "color":
"teal",
   "numberFormat": "~${{auto}}+"},
"series": [
   {"series": "all", "showDataValues": true},
   {"series": 0, "color": "burlywood"},
   {"series": 1, "color": "lightblue"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



JSON オブジェクトによる数値フォーマット設定

JSON オブジェクトには、次のプロパティが含まれています。プロパティのデフォルト値を使用する場合は、そのプロパティを省略することができます。

```
"numberFormat": {
   "mode": "string",
   "thousandSep": "string",
   "decimalSep": "string",
   "decimalPlaces": number,
   "grouping": "string",
   "prefix": "string",
   "suffix": "string",

}
```

説明

`"mode": "string"`

数値タイプを指定します。有効な値には、次のものがあります。

- "numeric" - これがデフォルト値です。
- "percent" - 値を 100 で乗算し、末尾にパーセント記号 (%) を追加した数値を表示します。
- "currency" - 通貨記号を追加した数値を表示します。表示される通貨記号は、数値フォーマットの設定に応じて異なります。

- "scientific" - 指数表記で数値を表示します。

`"thousandSep": "string"`

3 桁ごとの区切り文字を指定します。たとえば、カンマ (,) を指定すると、1,000,000 のように表示されます。デフォルト値は、数値フォーマットの設定に応じて異なります。

`"decimalSep": "string"`

小数部の区切り文字を指定します。たとえば、ピリオド (.) を指定すると、1.00 のように表示されます。デフォルト値は、数値フォーマットの設定に応じて異なります。

`"decimalPlaces": number`

小数点以下の桁数 (decimalSep で指定した区切り文字の右側に表示する桁数) を指定します。通貨および指数の場合、デフォルト値は 2 です。数値およびパーセント値の場合、デフォルト値は 0 (ゼロ) です。

`"grouping": "string"`

桁数の大きい数値の桁を省略し、略称を使用してグループ化する方法を指定します。有効な値には、次のものがあります。

- "K" - 1000 単位でグループ化します。たとえば、1,000 = 1K です。
- "M" - 100 万単位でグループ化します。たとえば、1,000,000 = 1M です。
- "B" - 10 億単位でグループ化します。たとえば、1,000,000,000 = 1B です。
- "T" - 1 兆単位でグループ化します。たとえば、1,000,000,000,000 = 1T です。

`"prefix": "string"`

ラベルの先頭に追加する 1 つまたは複数の文字を指定します。

`"suffix": "string"`

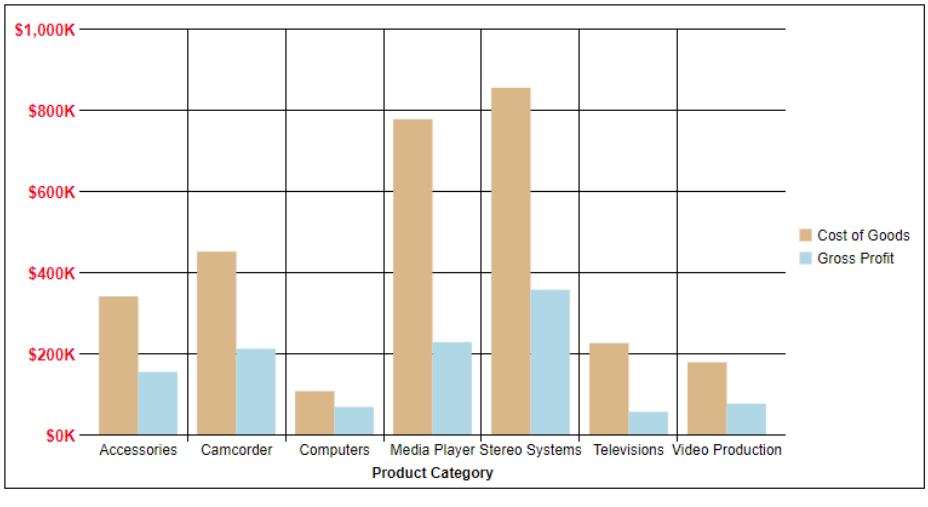
ラベルの末尾に追加する 1 つまたは複数の文字を指定します。

例　JSON オブジェクトによる数値フォーマット設定

次のリクエストは、Y 軸ラベルを通貨表記、1000 単位のグループ化、小数点以下の桁数 0 (ゼロ) のフォーマットに設定します。これにより、ドル記号 ($) および文字 K が追加された数値が表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"yaxis": {
   "labels": {"font": "bold 10pt Sans-Serif", "color": "red"},
   "numberFormat": {"mode": "currency", "decimalPlaces": 0,
        "grouping": "K"},
"series": [
   {"series": 0, "color": "burlywood"},
   {"series": 1, "color": "lightblue"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



フォーマット文字列による数値フォーマット設定

フォーマット文字列を使用して、数値ラベルのフォーマットを指定することができます。

```
numberFormat: "string"
```

文字列は、以下のサイトの Custom Numeric Format Strings で規定されているフォーマットで指定する必要があります。

http://msdn.microsoft.com/en-us/library/0c899ak8(vs.71).aspx

参照　フォーマット文字列の概要

数値のフォーマット文字列は、最大で 3 つのセクションで構成されます。セクションが 1 つの場合、指定したフォーマット文字列がすべての数値のフォーマット設定に使用されます。セクションが 2 つの場合、1 つ目のセクションが正の数値と 0 (ゼロ) のフォーマット設定に使用され、2 つ目のセクションが負の数値のフォーマット設定に使用されます。セクションが 3 つの場合、1 つ目のセクションが正の数値のフォーマット設定、2 つ目のセクションが負の数値のフォーマット設定、3 つ目のセクションが 0 (ゼロ) のフォーマット設定に使用されます。

これらのセクションはセミコロン (;) で区切ります。2 つ目のセクションがブランクの場合 (1 つ目と 3 つ目のセクション間に 2 つの連続したセミコロン (;) を記述した場合)、負の数値を除くすべての数値のフォーマット設定に 1 つ目のセクションが使用されます。

負の数値が端数処理により 0 (ゼロ) になる場合、その数値は 0 (ゼロ) 値としてフォーマット設定されます。

フォーマット文字列で有効な指定子は次のとおりです。

ゼロプレースホルダ

1 つのゼロ (0) または 2 つのゼロ (00)

フォーマット設定する数値に、フォーマット文字列で 0 (ゼロ) が指定された桁が存在する場合、その桁の数字が出力文字列にコピーされます。桁が存在しない場合、その桁に 0 (ゼロ) が挿入されます。整数部 (小数点の左側) の左端の 0 (ゼロ) と、小数部 (小数点の右側) の右端の 0 (ゼロ) により、出力文字列に常に表示される桁数が決定されます。

2 つのゼロ (00) プレースホルダを指定した場合、値が端数処理され、一の位に丸められます。端数処理では、端数が常に四捨五入されます。たとえば、34.5 を "00" でフォーマット設定すると、結果は 35 になります。

桁プレースホルダ

1 つのポンド記号 (#) または 2 つのポンド記号 (##)

フォーマット設定する数値に、フォーマット文字列で # が指定された桁が存在する場合、その桁の数字が出力文字列にコピーされます。桁が存在しない場合、出力文字列のその桁には何も挿入されません。この指定子では、文字列内の数字が 0 (ゼロ) のみの場合を含め、有効桁以外に存在する 0 (ゼロ) は表示されません。

フォーマット文字列で ## を指定した場合、値が端数処理され、一の位に丸められます。端数処理では、端数が常に四捨五入されます。たとえば、34.5 を "##" でフォーマット設定すると、結果は 35 になります。

小数部の区切り文字

数値フォーマットの設定に応じて異なります。英語 (米国) の場合、この区切り文字はピリオド (.) です。

フォーマット設定する値の小数点の位置は、フォーマット文字列の最初の小数部区切り文字で決定されます。その他の小数部区切り文字は無視されます。

3 桁の区切り文字および数値の位取り

数値フォーマットの設定に応じて異なります。英語 (米国) の場合、この区切り文字はカンマ (,) です。

3 桁の区切り文字には、2 つの機能があります。1 つ目は、整数部に 3 桁の区切り文字を挿入する機能です。フォーマット文字列で整数部 (小数部区切り文字の左側) の 2 つの桁プレースホルダ間 (0 または #) に 3 桁の区切り文字を指定すると、出力文字列の整数部に 3 桁ごとの区切り文字が挿入されます。

2 つ目は、数値の位取りを設定する機能です。フォーマット文字列で小数部区切り文字のすぐ左側に 1 つ以上の 3 桁区切り文字を指定すると、フォーマット設定する数値が、3 桁区切り文字の個数を 1000 倍した値で除算されます。たとえば、次のようにフォーマット文字列を指定すると、数値の 100,000,000 は 100 にフォーマット設定されます。

```
0,,
```

3 桁の区切り文字を使用して数値の位取りを設定した場合、フォーマット設定された数値に 3 桁の区切り文字は挿入されません。そのため、数値の位取りを 1,000,000 に設定し、3 桁ごとの区切り文字を挿入するには、次のフォーマット文字列を使用します。

```
#,##0,,
```

指数表記

E0、E+0、E-0、e、e+0、e-0

フォーマット文字列でこれらの指定子のいずれかを使用し、その直後に 0 (ゼロ) を 1 つ以上続けた場合、数値は指数表記でフォーマット設定され、数値と指数の間に "E" または "e" が挿入されます。指数表記の指定子の直後に続ける 0 (ゼロ) の個数により、出力文字列の指数部の最小桁数が決定されます。E+ または e+ のフォーマットでは、正の符号 (+) または負の符号 (-) が指数部の先頭に常に挿入されます。E、E-、e、e- のフォーマットでは、指数部が負の値の場合にのみ符号が挿入されます。

パーセントプレースホルダ

数値フォーマットの設定に応じて異なります。英語 (米国) の場合はパーセント記号 (%) です。

フォーマット文字列にパーセントプレースホルダ文字が存在する場合、数値がフォーマット設定される前に、その数値が 100 で乗算されます。フォーマット文字列でパーセントプレースホルダ文字が指定された位置の数値自体に特定の記号が挿入されます。

リテラルまたはその他の文字

一重引用符 (') または二重引用符 (") で囲まれた文字です。たとえば、"ABC" と指定します。

一重引用符 (') または二重引用符 (") で囲まれた文字は、結果の文字列にそのままコピーされます。

このセクションで定義されたフォーマット指定子以外に、次の文字を使用することもできます。

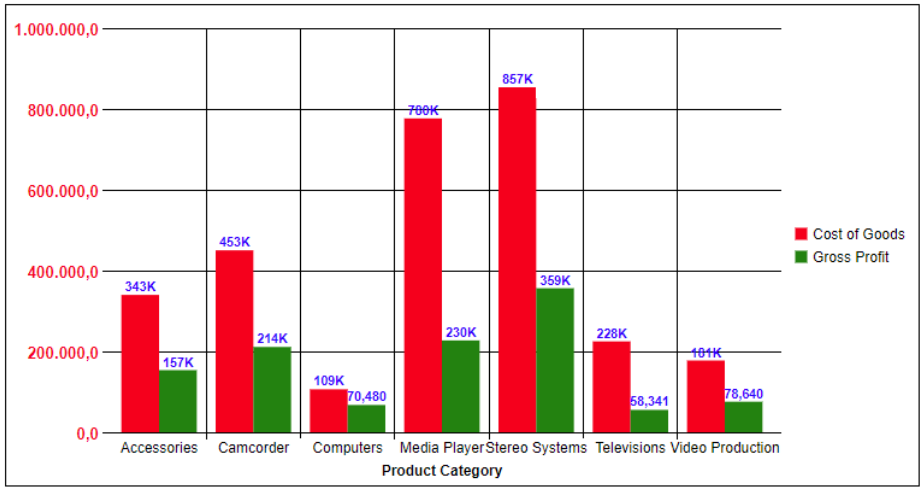
- `"{tc}"` - 3 桁ごとの区切り文字を指定します。ここで、c は区切り文字を表します。
- `'{dc}'` - 小数部の区切り文字を指定します。ここで、c は区切り文字を表します。
- アクサングラーブ (`) - フォーマットにパーセント記号 (%) を使用しますが、フォーマット設定前に数値を 100 で乗算しません。

例　数値フォーマット文字列によるラベルのフォーマット設定

次のリクエストでは、Y 軸ラベルおよびデータラベルに数値フォーマット文字列を使用します。Y 軸オブジェクトのフォーマット文字列で、{t} および {d} 構文を使用して 3 桁ごとの区切り文字をピリオド (.) に変更し、小数部の区切り文字をカンマ (,) に変更します。シリーズデータラベルオブジェクトでは、ラベルの表示を青色で有効にし、デフォルトの数値フォーマットオプションを使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"series": [{"series": "all",
  "dataLabels": {"visible": true, "position": "top", "font": "bold 8pt Sans-
Serif", "color": "blue"}}],
"axisList": {
"y1": {
   "labels": {"font": "bold 10pt Sans-Serif", "color": "red"},
   "numberFormat": "{t.}{d,}#,#.#"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　条件付き数値フォーマットの指定

条件付き数値フォーマットを指定するには、次の関係演算子を使用し、値の範囲に対して数値フォーマットを定義します。

- > (より大きい)
- >= (以上)
- = (等しい)
- < (より小さい)
- <= (以下)

定義する範囲は、左大括弧 ([) と右大括弧 (]) で囲む必要があります。

フォーマット文字列を使用して数値フォーマットを適用するデータ範囲を指定する場合は、フォーマット文字列にデータ範囲全体が含まれるようにします。範囲とフォーマットの各組み合わせは、セミコロン (;) で区切ります。たとえば、次のようにフォーマットを定義します。

- #,# - 9 以下の値 ([<=9]#,#)
- -#.# - 0 (ゼロ) より小さい値 ([<0]-#.#)
- # - 0 (ゼロ) に等しい値 ([=0]#)
- #.# - 9 より大きい値 ([>9]#.#)

```
numberFormat:'[<=9]#,#;[<0]-#.#;[=0]#;[<0]-#.#;[>9]#.#'
```

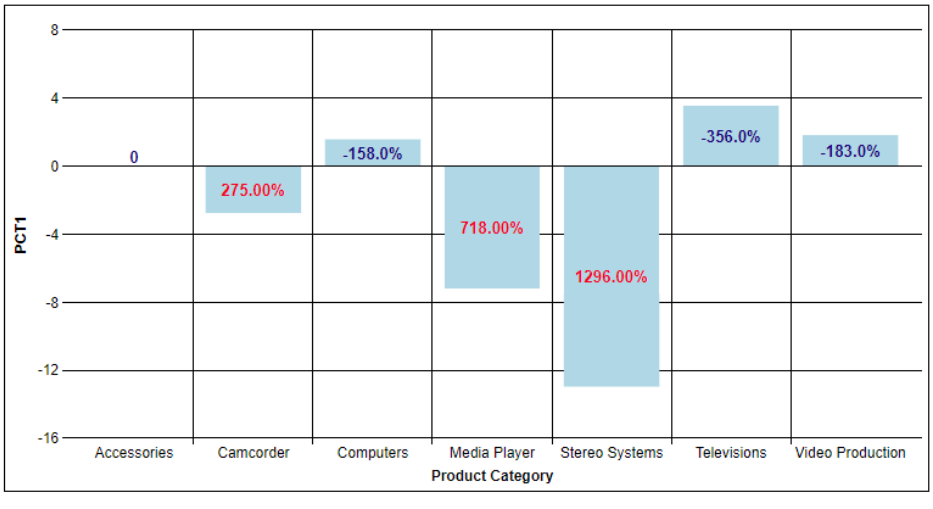
例　条件付き数値フォーマットの指定

次のリクエストは、割合を計算します。データラベルオブジェクトで、データラベルの表示を有効にし、正の値に紺色、負の値に赤色を使用します。また、数値フォーマットを次のように設定します。

- 数値が 0 (ゼロ) より大きい場合、数値を小数点以下 2 桁のパーセント値として表示します ([>=0]#.##%)。
- 数値が 0 (ゼロ) より小さい場合、数値を小数点以下 1 桁のパーセント値として表示し、先頭にマイナス記号 (-) を追加します ([<0]-#.#%)。
- 数値が 0 (ゼロ) の場合、数値を 0 と表示します ([=0]0)。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED NOPRINT
AND COMPUTE
PCT1 = IF DAYSDELAYED NE 763 THEN (500-DAYSDELAYED)/100 ELSE 0;
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"series": [{"series": "all",
 "dataLabels": {"visible": true, "useNegativeColor": true, "negativeColor":
"red",
   "font": "bold 10pt Sans-Serif", "color": "navy", "position": "inside",
   "numberFormat": "[<=0]#.##%;[>0]-#.#%;[=0]0"}},
   {"series": 0, "color": "lightblue"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



関数による数値フォーマット設定

関数は 1 つの引数 (フォーマット設定する数値) を取り、文字列を返します。以下はその例です。

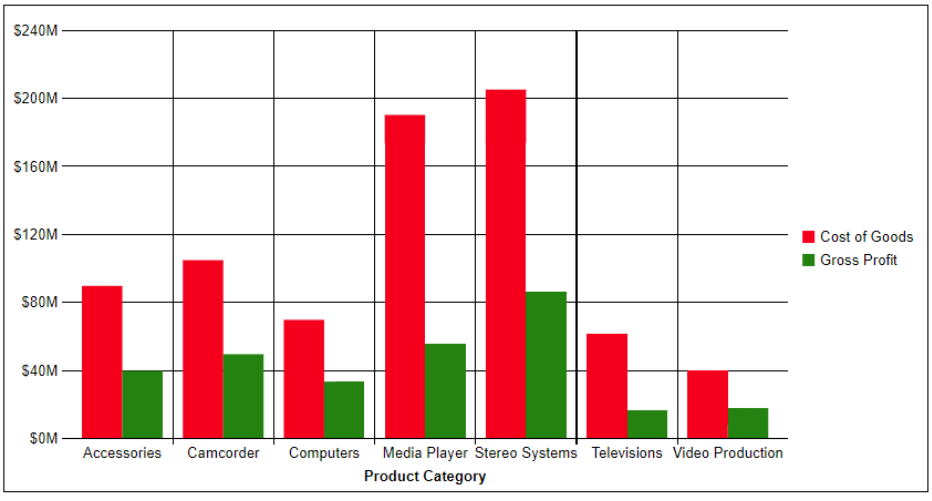
```
// Return number with a "$" in front
numberFormat = function(n){ return "$" + n; };
```

例　関数による数値フォーマット設定

次のリクエストでは、関数を使用して Y 軸ラベルの値を 100 万で除算し、先頭にドル記号 ($)、末尾に「M」を追加して結果を返します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"axisList": {
"y1": {
"numberFormat": function(n){ return "$" + n/1000000 + "M";}}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



参照　　数値フォーマットの使用上の注意

- numberFormat プロパティを "auto" に設定する場合を除いて、接頭語、接尾語、3 桁ごとの区切り文字、小数部の区切り文字としてセミコロン (;) を使用することはできません。セミコロン (;) は、複数の条件付きフォーマットを区切る場合にのみ有効です (例、"[>=0]#,#;[<0]-#.#")。
- numberFormat プロパティを "auto" に設定する場合を除いて、接頭語または接尾語としてポンド記号 (#)、ピリオド (.)、疑問符 (?) を使用することはできません。これらの記号は、文字列として解釈されない特殊文字です。
- JSON オブジェクト定義で接頭語または接尾語としてパーセント記号 (%) を使用する場合、モード設定に関係なく、数値フォーマットではパーセントモードが使用されます。
- すべての数値をヨーロッパ言語の数値フォーマットでグラフに表示するには、SET CDN コマンドを発行することができます。

ツールヒントおよびラベルの数値フォーマット制御

autoNumberFormats プロパティで、データラベル、積み上げグラフの合計ラベル、円グラフの合計ラベル、およびツールヒント ("tooltip": "auto" プロパティ) の数値フォーマットを制御します。

構文　ツールヒントおよびラベルの数値フォーマット制御

```
"autoNumberFormats": {
      "object": "numberfmt", ...
}
```

説明

`object`

次のいずれかのオブジェクトです。

- dataLabels
- tooltip
- stackTotalLabel
- totalLabel

カンマ (,) で区切ることで複数のオブジェクトを含めることができます。

`numberfmt`

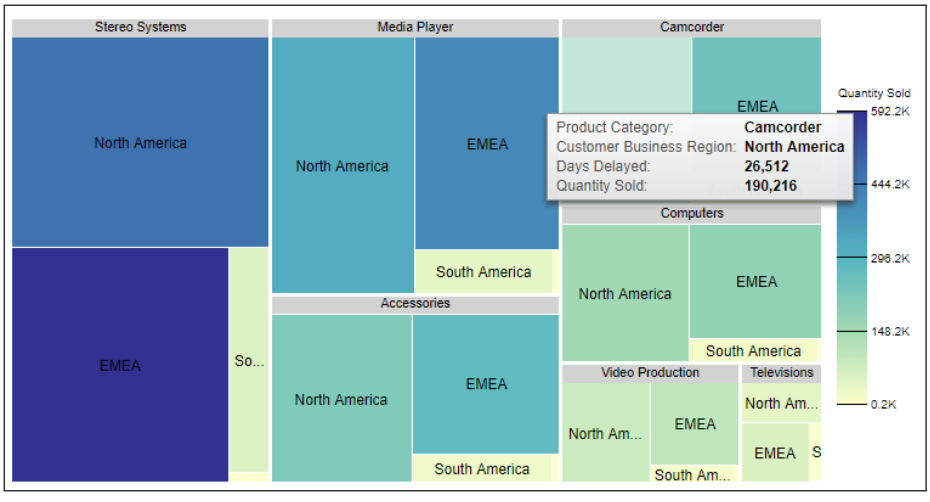
有効な数値フォーマットです。数値フォーマットについての詳細は、86 ページの 「 数値のフォーマット設定 」 を参照してください。

例　ツールヒントの数値フォーマット指定

次のリクエストは、ツリーマップを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
type=data, column=n1, bucket=detail,$
type=data, column=n2, bucket=detail,$
type=data, column=daysdelayed, bucket=size,$
type=data, column=quantity_sold, bucket=color,$
ENDSTYLE
END
```

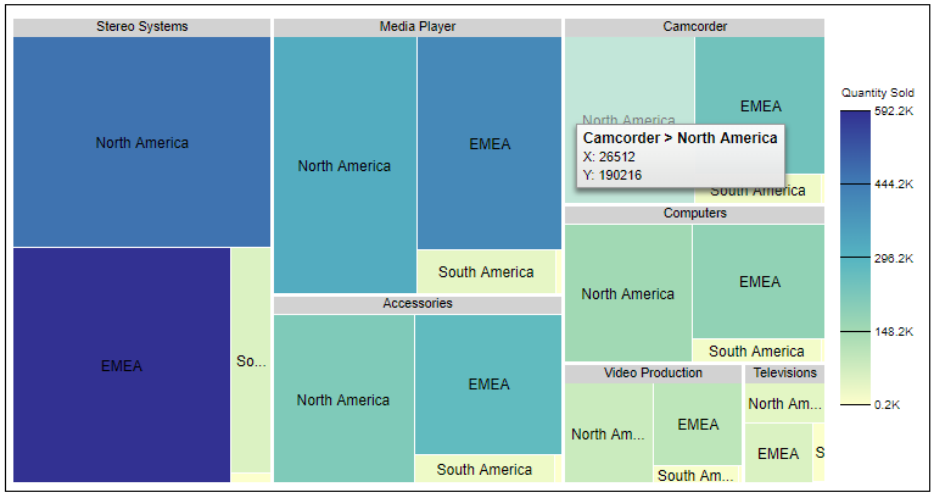
下図は、デフォルト設定のツールヒントの数値フォーマットを示しています。



次のリクエストでは、カンマ (,) を使用しないツールヒントの数値フォーマットを指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
type=data, column=n1, bucket=detail,$
type=data, column=n2, bucket=detail,$
type=data, column=daysdelayed, bucket=size,$
type=data, column=quantity_sold, bucket=color,$
*GRAPH_JS_FINAL
"autoNumberFormats": {"tooltip": "###"},
"series": [{"series": 0, "tooltip": "auto"}]
*END
ENDSTYLE
END
```

下図は、新しいツールヒントの数値フォーマットを示しています。



文字列内での HTML コードの使用

ラベルまたはタイトルの文字列に HTML コードを埋め込むことができます。タイトルおよびラベル内の HTML コードは、ライブラリによる処理は実行されません。これらのコードはブラウザに送信され、ブラウザ側で処理されて表示されます。以下はその例です。

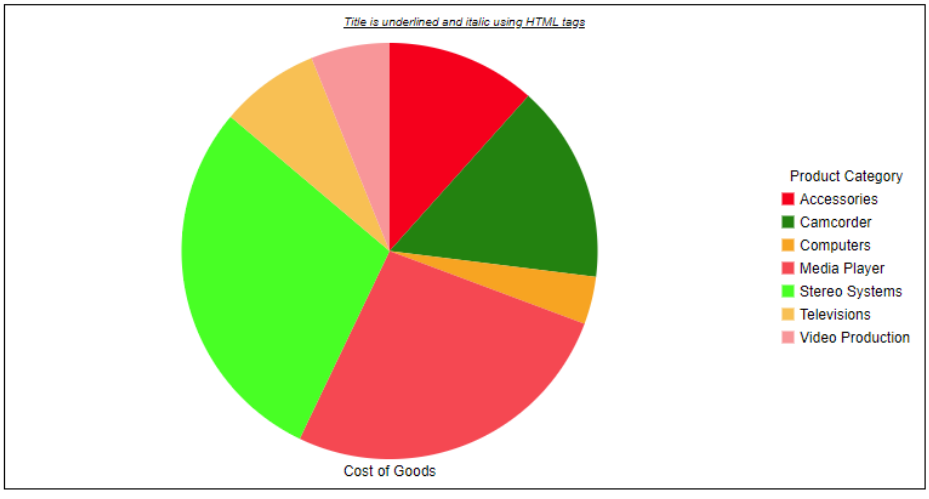
```
"title": {"text": "<u><em>Title is underlined and italic
using HTML tags<¥/em><¥/u>"},
```

例　グラフタイトルテキスト内での HTML タグの使用

次のリクエストでは、タイトルテキストの前後に HTML 下線タグ (<u> および </u>) と斜体タグ (および) を追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"title": {"visible": true,
   "text": "<u><em>Title is underlined and italic using HTML tags<¥/em><¥/u>"}
*END
ENDSTYLE
END
```

下図のように、出力結果では、下線付きおよび斜体のグラフタイトルが表示されます。



データ定義

データソースにミッシング値が含まれている場合、VZERO パラメータを使用することで、グラフ上のミッシング値の処理方法を指定することができます。

VZERO を ON に設定した場合、Y 軸のミッシングデータは 0 (ゼロ) として処理されます。OFF に設定した場合、Y 軸のミッシングデータは無視され、値はプロットマトリックスに格納されません。デフォルト値は OFF です。

グラフのプロパティ値の設定

すべてのプロパティには、undefined または null (引用符で囲まれていない文字列) を値として割り当てることができます。プロパティ値として undefined または null を使用した場合、そのプロパティは無視されます。そのプロパティに割り当てられているデフォルト値が使用されます。

ラベル値を返すコールバック関数の定義

ラベルおよびツールヒントに表示するデータ値、シリーズ番号、グループ番号、データオブジェクトを返すコールバック関数を定義することができます。コールバック関数は、JavaScript でよく使用されます。コールバック関数の定義および使用についての詳細は、JavaScript のマニュアルを参照してください。

この関数は、最大で 4 つの引数を取ることができます。これらの引数は位置が固定され、1 つ目の引数はデータ値、2 つ目の引数はシリーズ番号、3 つ目の引数はグループ番号を表します。

構文　ラベル値を返すコールバック関数の定義

```
formatCallback: function(d,s,g, dataset){return expression;}
```

説明

`d`

データ値を返す引数です。このオブジェクトのプロパティで、関数から返されるデータのタイプを指定します。返されるデータのタイプ (例、X 値、Y 値) は「データモード」と呼ばれ、生成するグラフタイプに応じて異なります。

関数からデータ値を返す式では、返されるデータモードを指定する必要があります。たとえば、X 値を返すには、式に「d.x」を指定します。

下表は、各グラフタイプで使用可能なデータモードを示しています。ここに記載されていないグラフタイプには、デフォルトモードがあります。これらのデフォルトモードは、value、size、color などです。

グラフタイプ	データモード
ボックスプロット	min, lower, median, upper, max
極グラフ	phi, r, size, color
バブルグラフ	x, y, size
散布図	x, y
ツリーマップ	size, color
コントロール	value, sampleSize
マップ	name, value (引数が 2 つの場合に使用) または lng, lat, color (引数が 3 つの場合に使用)

`s`

シリーズ番号を返す引数です。

`g`

グループ番号を返す引数です。

`dataset`

シリーズのこのポイントに使用するデータセットです。グラフタイプによっては、データセット全体を指定することも、undefined を指定することもできます。通常、積み上げ棒グラフおよび円グラフでデータシリーズの合計を計算する場合に役立ちます。

`expression`

JavaScript 関数でサポートされる式です。

関数に渡される引数以外に、一重引用符 (') で囲まれたテキストや HTML タグを式に挿入することもできます。また、JavaScript でサポートされる他のオブジェクト、プロパティ、関数を使用することもできます。

戻り値の各コンポーネントは、それぞれをプラス記号 (+) で区切ります。式は、セミコロン (;) で終了する必要があります。

また、コールバック関数で新しい変数を宣言したり、CSS スタイルを使用したりすることも可能です。さらに、パラメータとして渡されたオブジェクトのプロパティを取得することもできます。たとえば、シリーズの引数が s の場合、シリーズ 0 の色プロパティを取得するには、次のように記述します。

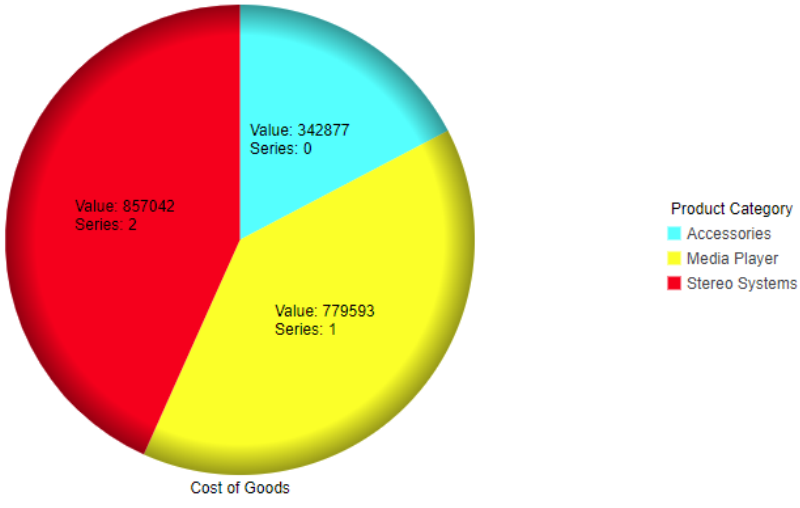
```
"series": 0, "color": "green",
              "tooltip": function(d,s,g){
                      var c = this.getSeries(s).color;
```

例　円グラフのデータラベルを返すコールバック関数の定義

次のリクエストは、円グラフを生成し、データラベルを表示します。ラベルは、a (データ) と b (シリーズ) の 2 つの引数を取るコールバック関数から返されます。この関数は、データ値 (a.value) を返す以外に、表示内容を示すテキストを返します。また、ラベルを 2 行で表示するための HTML 改行タグ (<br>) も含まれています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Accessories' OR 'Stereo Systems'
OR 'Media Player'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"series": [{series: "all"},
   {"series": 0, "color": "cyan"},
   {"series": 1, "color": "yellow"},
   {"series": 2, "color": "red"}
],
"dataLabels": {
   "visible": true,
   "position": "middle",
   "formatCallback": function(a,b){return "Value: " + a.value +
      "<br>Series: " + b ;}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



例　コールバック関数によるカスタムツールヒントの定義

次のリクエストは、積み上げ棒グラフを生成し、2 つのシリーズを表示します。各シリーズのツールヒントは、コールバック関数から返されます。

シリーズ 0 のコールバック関数は、シリーズ 0 の色およびラベルを取得し、これらのプロパティを返します。ツールヒントのテキスト色として、シリーズ色を使用します。<span オブジェクトは、コールバック関数から返された値にフォーマットを設定する CSS スタイルです。二重引用符 (") の前の円記号 (¥) は、エスケープ文字です。これにより、コールバック関数が、二重引用符 (") を特殊文字として処理する代わりに、文字として二重引用符 (") を返します。ツールヒントには、複数行を生成するための改行タグ (<br>) が含まれています。

```
 "series": [
{"series": 0, "color": "green",
 "tooltip": function(d,s,g){
 var c = this.getSeries(s).color;
 var lbl = this.getSeries(s).label;
 return '<span style=¥"color: ' + c + '¥">value: ' + d + '<br />series:
' + s + '<br />group: ' + g + '<br />series name: ' + lbl +'</span>'; }
     }
```

シリーズ 1 のコールバック関数は、シリーズ 1 から色およびラベルを取得します。また、両方のシリーズからデータ値を取得し、これらを加算して合計値を求めます。

```
{"series": 1, "color": "blue",
        tooltip: function(d,s,g){
        var c = this.getSeries(s).color;
        var lbl = this.getSeries(s).label;
        var total = this.data[0][g][0] + this.data[1][g][0];
        return '<span style=¥"color: ' + c + '¥">value: ' + d + ' out of '
+
    total + '<br />series name: ' + lbl +'</span>';}
                    }
```

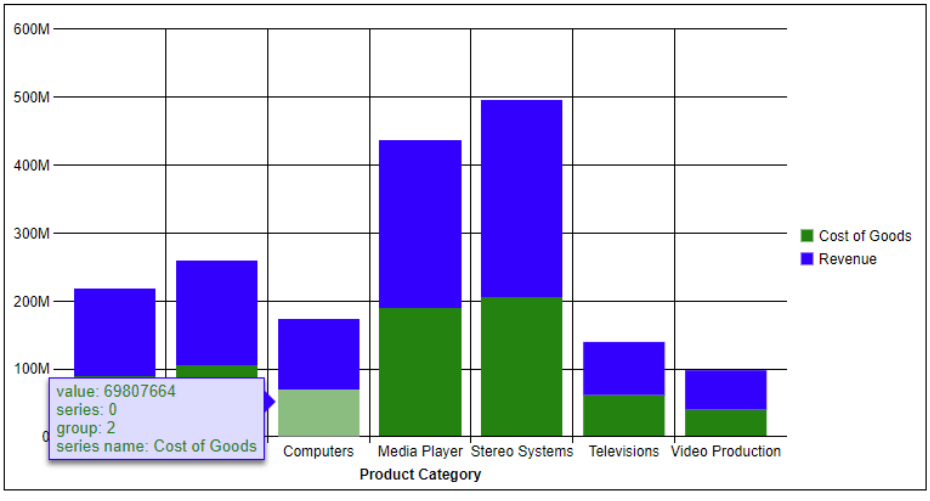
ツールヒントは、境界色が青色、塗りつぶし色が水色 (#DDDDFF) の HTML ツールヒントとして表示されます。

注意：戻り値は、一重引用符 (') で囲みます。このマニュアルの一部のサンプルリクエストでは、文字列が分割されて次の行に折り返されている場合がありますが、リクエストでは文字列を 1 行に記述する必要があります。このサンプルリクエストをコピーして実行するには、戻り値内の改行を削除する必要があります。

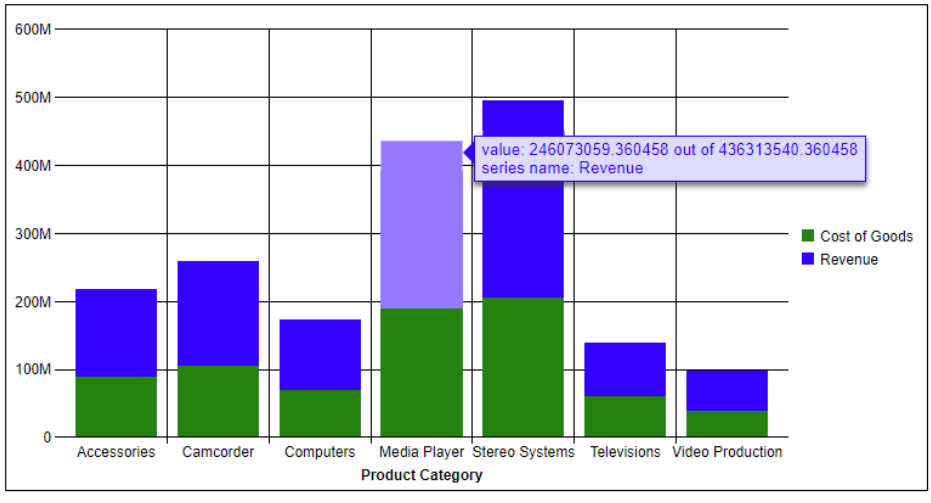
リクエストは次のようになります。

```
GRAPH FILE retail8205/WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=report, chart-series-layout=stacked,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"series": [
    {"series": 0, "color": "green",
        tooltip: function(d,s,g){var c = this.getSeries(s).color;
            var lbl = this.getSeries(s).label;
            return '<span style=¥"color: ' + c + '¥">value: ' + d + '<br />series: ' +
            s + '<br />group: ' + g + '<br />series name: ' + lbl +'</span>'; }
    },
    {"series": 1, "color": "blue",
        tooltip: function(d,s,g){var c = this.getSeries(s).color;
            var lbl = this.getSeries(s).label;
            var total = this.data[0][g][0] + this.data[1][g][0];
            return '<span style=¥"color: ' + c + '¥">value: ' + d + ' out of ' +
            total + '<br />series name: ' + lbl +'</span>';}}],
"htmlToolTip": {"snap": true, "fill": "#DDDDFF", "border": {"color": "blue"}}
ENDSTYLE
END
```

下図は、シリーズ 0 のツールヒントを示しています。



下図は、シリーズ 1 のツールヒントを示しています。



グラフテンプレートエンジンを使用したツールヒント、タイトル、データラベルのカスタマイズ

特殊な文字列で記述されるテンプレートを使用すると、グラフエンジンがグラフを描画する際に、そのテンプレートがグラフコンポーネント情報で動的に置換されます。テンプレートは、複数のマクロで構成されます。マクロは、二重の中括弧 ({{ }}) で囲まれた文字列です。各マクロがそれぞれ独立して解析、置換された結果、最終的な文字列が生成されます。グラフエンジンには、多数の定義済みマクロが組み込まれています。これらのマクロは、データテキストラベル、ツールヒント、グラフタイトル、軸タイトル、凡例タイトルに使用されます。また、WebFOCUS リクエストの見出しに挿入することもできます。この場合、WebFOCUS ではマクロを処理することはできませんが、見出しをグラフエンジンに渡して処理することができます。

マクロはパラメータ化することができます。パラメータは、カンマ区切りの文字列リストで、括弧 ( ) で区切られています。組み込みマクロの row_label および col_label はパラメータ化されており、どの行または列ラベルを含めるか選択することができます。1 行目が行番号ゼロ (0) になります。以下はその例です。

```
{{row_label(0)}}
```

グラフ属性構文に適用されるマクロもあります。グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

カスタムデータテキストラベルを定義する際に、テンプレートマクロを使用してグラフコンポーネント情報を表示することができます。

データテキストラベルにはデフォルト値が自動的に挿入されるため、スタイルシートの GRAPH_JS_FINAL ブロックで、次のプロパティを追加してデフォルトフォーマット設定を無効にした上で、カスタムデータラベルを配置する必要があります。

```
"dataLabels": {"visible":true,"formatCallback": null, "displayMode": null}
```

参照　テンプレートマクロ

下表は、グラフテンプレートエンジンの定義済みマクロのリストです。

マクロ	定義
auto_percent	円グラフ、積み上げ棒グラフ、ツリーマップの内部グループ、およびマリメッコチャートのパーセント値の指定
auto_tooltip_content	自動的に生成されるツールヒントのテキスト

マクロ	定義
auto_value	データラベル文字列に、次の項目を自動的に挿入 ❑ グラフタイプに応じて、マーカーまたはライザに関連する最も意味のあるメジャー ❑ マスターファイルのフォーマットに準拠した各メジャーのフォーマット ❑ 散布図で X 値または Y 値のみを追加した場合などの部分的なデータ **注意：**このマクロは、グラフ属性構文を使用するグラフリクエストのみでサポートされます。
auto_value_compressed	auto_value と同義ですが、数値の短縮形を使用します。たとえば、値 187,000 は、187K と表示されます。
col_id	マトリックスグラフの列番号
col_label	マトリックスグラフの列ラベル
coloraxis_title	色スケール使用時の色軸上の表示テキスト
data_page_slider_current_index	スライダのインデックスの位置
data_page_slider_current_label	スライダの現在位置ラベル
group_count	グループ数
group_id	グループの番号
group_label	グループのラベル
group_percent	グループ内の各シリーズのパーセント
group_sum	現在のグループのシリーズ合計
metadata_value	メタデータ数値フォーマットのデータラベルおよびツールヒントへの適用 詳細は、118 ページの 「 メタデータ数値フォーマットのデータラベルおよびツールヒントへの適用 」 を参照してください。

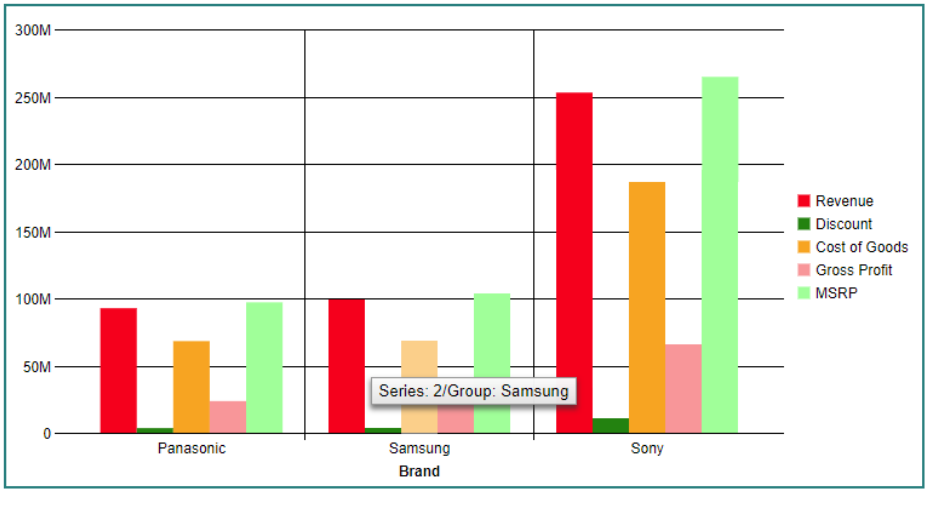
マクロ	定義
misc_id	特定のライザのその他 ID。実際の値は、評価される現在のライザによって異なります。たとえば、棒の場合は misc_id は 'bar' になり、マーカーの場合は 'marker'、折れ線グラフの線の場合は 'line' になります。
nested_label	グループのラベル
object_id	評価される現在のライザまたはグラフオブジェクトのオブジェクト ID。たとえば、棒グラフの棒の場合、object_id は 'riser' になります。
row_id	マトリックスグラフの行番号
row_label	マトリックスグラフの行ラベル
series_count	リクエスト内のシリーズの数
series_id	シリーズの番号
series_label	シリーズのラベル
series_percent	現在のシリーズの全グループに占めるパーセント
series_sum	現在のシリーズの全グループの合計
tooltip*n*	tooltip 属性カテゴリの *n* 番目のフィールド
value	グラフ属性構文を使用した場合に、シリーズが表す値
xaxis_title	X 軸のタイトル
yaxis_title	Y 軸のタイトル
zaxis_title	Z 軸のタイトル

例　ツールヒントでのテンプレートエンジンの使用

次のリクエストは、シリーズ ID とグループラベルを含むツールヒントを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US DISCOUNT_US COGS_US GROSS_PROFIT_US MSRP_US
BY BRAND
WHERE BRAND EQ 'Sony' OR 'Samsung' OR 'Panasonic'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
*GRAPH_JS_FINAL
"border": {"width": 2, "color": "teal"},
"series": [{"series": "reset",
    "tooltip": "Series: {{series_id}}/Group: {{group_label}}"}]
*END
ENDSTYLE
END
```

下図では、グループ 1、シリーズ 2 のツールヒントが表示されています。



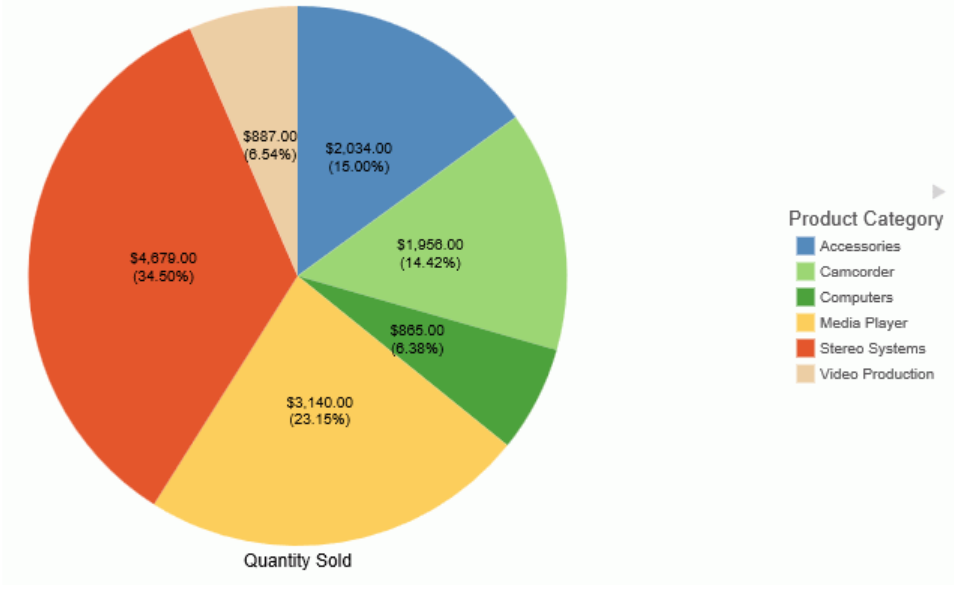
注意：リクエストにグラフ属性構文が使用されている場合は、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにツールヒントカスタマイズのプロパティを追加する必要があります。グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」を参照してください。

例　データテキストラベルでのテンプレートマクロの使用

次のリクエストは、円グラフを生成します。シリーズのデータテキストラベルには、販売数量の合計が、ドル記号 ($) と小数点以下 2 桁でフォーマット設定された数値として表示されます。このリクエストでは、HTML 文字列の「¥n」を使用して新しい行に移動し、auto_percent テンプレートマクロを使用してパーセント値を表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY NE 'Televisions'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=color, $
TYPE=DATA, COLUMN=N2, BUCKET=measure, $
*GRAPH_JS_FINAL
"dataLabels": {"visible":true,"formatCallback": null, "displayMode": null},
"series":[{"series": "reset",
 "dataLabels":{"content":"{{value | formatNumber('$#,###.00')}}
¥n({{auto_percent}})"}}],
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

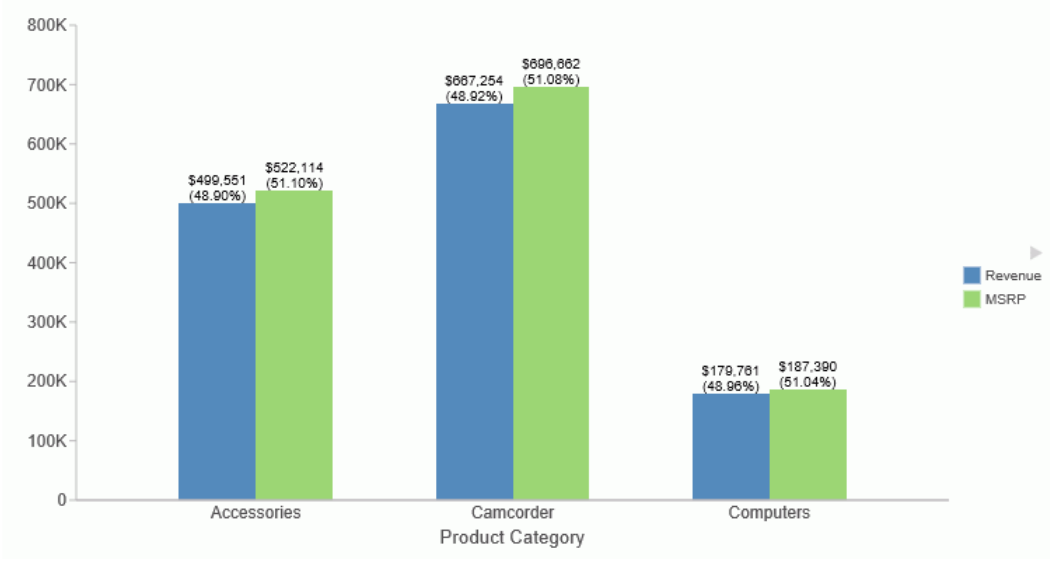


例　棒グラフのデータテキストラベルでのテンプレートマクロの使用

次のリクエストは、棒グラフを生成します。シリーズのデータテキストラベルには、収益および希望小売価格が、円記号 (¥) で小数点以下の桁数なしでフォーマット設定された数値として表示されます。このリクエストでは、HTML 文字列の「¥n」を使用して新しい行に移動し、group_percent テンプレートマクロを使用してグループ内の各シリーズのパーセント値を表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Accessories' OR 'Computers' OR 'Camcorder'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"dataLabels": {"visible":true,"formatCallback": null, "displayMode": null},
"series":[{"series": "reset",
 "dataLabels":{"content":"{{value | formatNumber('$#,###')}}¥n({{group_percent}})"}}],
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

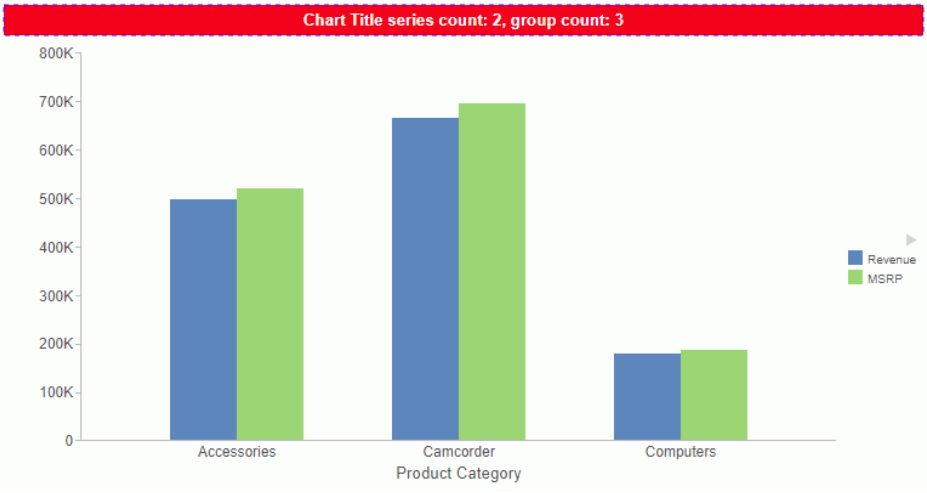


例　グラフタイトルでのテンプレートマクロの使用

次のリクエストは、グラフタイトルに series_count および group_count マクロを追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
MSRP_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Accessories' OR 'Computers' OR 'Camcorder'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"title": {
"visible": true,
"text": "Chart Title series count: {{series_count}}, group count:
{{group_count}} ",
"color": "white",
"backgroundColor": "red",
"border":
        {
        "width": 1,
        "color": "blue",
        "dash": "4 4"
        }
},

*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフタイトル、サブタイトル、脚注での series_label マクロの使用

series_label マクロの値は、グラフタイトル、サブタイトル、脚注に挿入することができます。

構文　　グラフタイトル、サブタイトル、脚注での series_label マクロの使用

```
"titletype": {
    "visible": true,
    "text": "text1 {{series_label(null, null,{series: number})}} text2"
},
```

説明

`"titletype"`
次のグラフタイトルオブジェクトのいずれかです。

- "title" - グラフタイトルにシリーズラベルを配置します。
- "subtitle" - グラフサブタイトルにシリーズラベルを配置します。
- "footnote" - グラフ脚注にシリーズラベルを配置します。

`text1, text2`
タイトル、サブタイトル、脚注テキストでサポートされる任意のテキストまたはオブジェクトです (オプション)。

series: number
タイトル、サブタイトル、脚注にラベルを挿入するシリーズの番号です。

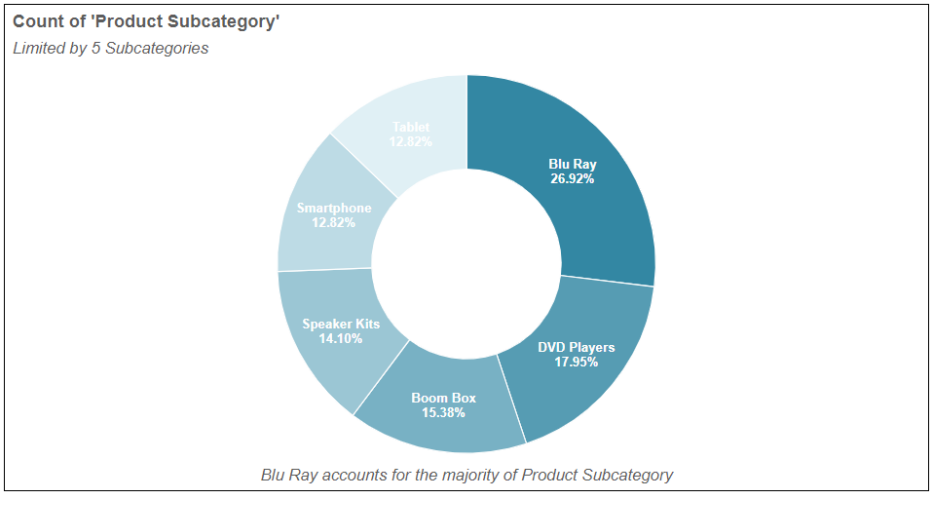
例 series_label マクロのグラフ脚注への挿入

次のリクエストは、製品詳細区分合計の上位 5 件を降順にソートした円グラフを生成します。最上位の製品詳細区分ラベルは、ソート順のシリーズ 0 に対応しており、脚注に挿入されています。

```
GRAPH FILE wf_retail_lite
SUM CNT.PRODUCT_SUBCATEG
BY TOTAL HIGHEST 5 CNT.PRODUCT_SUBCATEG NOPRINT
BY PRODUCT_SUBCATEG
ON TABLE SET LOOKGRAPH PIE
ON TABLE SET EMBEDHEADING ON
ON TABLE PCHOLD FORMAT JSCHART
ON TABLE SET STYLE *
*GRAPH_JS
"colorMode": {"mode": "byInterpolation", "colorList":
["#3387A3","#E0F0F5"]} ,
"series": [
        {"series": "all", "border": {"width": 1, "color": "white"},
            "dataLabels":{
                "color": "white",
                "font": "bold 8pt Sans-Serif",
                "content": "{{series_label}}¥n{{auto_percent}}",
                "visible": true,
             }}
        ],
"title": {
    "color": "#5C5C5C",
    "font": "bold 11pt Sans-Serif",
    "text": "Count of 'Product Subcategory'",
    "align": "left",
    "visible": true
},
"subtitle": {
    "color": "#5C5C5C",
    "font": "italic 10pt Sans-Serif",
    "text": "Limited by 5 Subcategories",
    "align": "left",
    "visible": true
},
"footnote": {
    "color": "#5C5C5C",
    "font": "italic 10pt Sans-Serif",
    "text":
    "{{series_label(null, null,{series: 0})}} accounts for the majority of
Product Subcategory",
    "align": "center",
    "visible": true
},
```

```
"pieProperties": {
    "holeSize": "50%",
    "skew": 22,
    "label": {"visible": false},
    "totalLabel": {"visible": false},
},
"legend": {
    "visible": false
},

*END
  TYPE=HEADING, STYLE=BOLD, $
  TYPE=REPORT, ORIENTATION=LANDSCAPE, $
  TYPE=DATA, COLUMN=N1, BUCKET=color, $
  TYPE=DATA, COLUMN=N2, BUCKET=color, $
  TYPE=DATA, COLUMN=N3, BUCKET=measure, $
ENDSTYLE
END
```

出力結果は次のとおりです。2 つの合計件数が同一だったため、上位 5 件の合計に 6 件目の詳細区分が選択されています。脚注では、最も多くのデータを含む製品詳細区分として、「Blu Ray」が特定されています。



メタデータ数値フォーマットのデータラベルおよびツールヒントへの適用

metadata_value グラフテンプレートマクロは、WebFOCUS メタデータの数値フォーマットをグラフオブジェクトに適用します。GRAPH FILE プロシジャでフォーマットが指定されている場合、このフォーマットが使用されます。それ以外の場合は、マスターファイルのフォーマットが使用されます。

構文　メタデータのフォーマットのデータラベルまたはツールヒントへの適用

```
{{metadata_value(data_array_entry)}}
```

説明

`data_array_entry`
データ配列マップのエントリです。データ配列マップのエントリは、グラフタイプによって異なり、それぞれデフォルトエントリが設定されています。エントリ名が分からない場合でも、データ配列エントリに "default" を指定することで、デフォルトエントリを使用することができます。グラフタイプ別エントリのリストは次のとおりです。

- **ボックスプロット**
  - "min"
  - "lower"
  - "median" (デフォルト)
  - "upper"
  - "max"
- **極グラフ**
  - "phi"
  - "r" (デフォルト)
  - "size"
  - "color"
- **バブルグラフ**
  - "x"
  - "y"
  - "size" (デフォルト)
- **散布図**
  - "x" (デフォルト)
  - "y"
  - "pie"

- **コロプレス**
  - "name"
  - "value" (デフォルト)
- **バブルマップ**
  - "lng"
  - "lat"
  - "value" (デフォルト)
  - "color"
- **マトリックスマーカーグラフ**
  - "size" (デフォルト)
  - "color"
- **ツリーマップ**
  - "size" (デフォルト)
  - "color"
- **ヒートマップ**
  - "color" (デフォルト)
- **タグクラウド**
  - "value" (デフォルト)
  - "color"
- **ブレットグラフ**
  - "value" (デフォルト)
  - "markers"
- **デフォルト** (上記以外で、value、size、color プロパティを使用するグラフタイプ。例、棒グラフ、折れ線グラフ、円グラフ)
  - "value" (デフォルト)
  - "size"

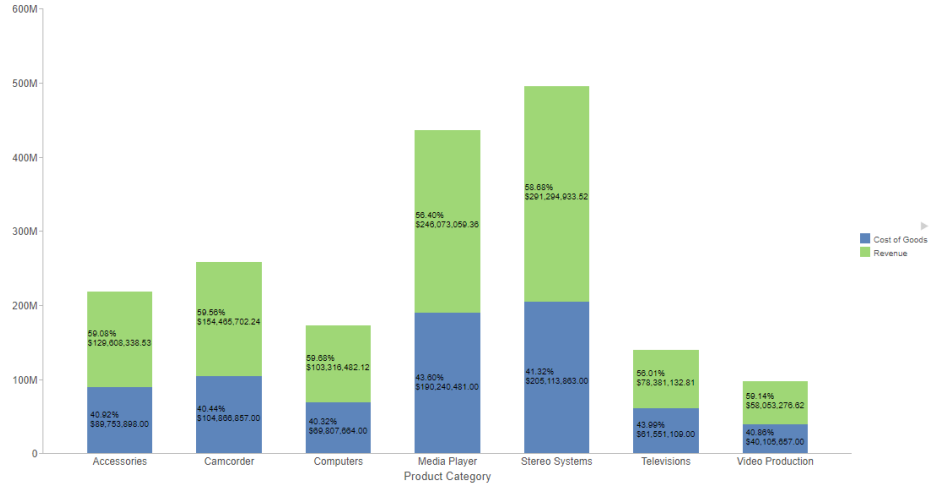
- "color"

例　マスターファイルのフォーマットを使用したデータラベルの表示

次のリクエストは、積み上げ棒グラフを作成します。データラベルのコンテンツは、2 行に記述されます。1 行目には、auto_percent マクロを使用して各積み上げ棒のパーセントを表示し、改行を追加します。2 行目には、マスターファイルの数値表示フォーマット (D20.2M) を使用して、各積み上げ棒のシリーズの値を表示します。このフォーマットでは、有効数字の最上位桁の左側に円記号 (¥) を追加し、小数点以下 2 桁で数値を表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART

ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, ORIENTATION=LANDSCAPE, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_SCRIPT

*GRAPH_JS_FINAL
"series":     [
   {"series": "all",
        "border": {
                "color": "rgba(0,0,0,0)"
            },
    "dataLabels": {
    "visible": true,
    "position": "center",
    "content": '{{auto_percent}} <br> {{metadata_value("value")}}'

},
        }
    ],
"blaProperties": {
    "seriesLayout": "stacked"
},


*END
ENDSTYLE
END
```

出力結果は次のとおりです。



4

TIBCO WebFOCUS グラフ構文

グラフ属性構文は、WebFOCUS スタイルシート属性を使用して、HTML5 リクエストのディメンションとメジャーに役割を割り当てます。グラフ属性構文を使用すると、この構文でのみ作成可能なグラフタイプを生成できるほか、複数のグラフをマトリックス配列で表示することもできます。また、この構文では簡素化されたグラフタイプリストが使用され、特定のグラフタイプではスライダやページの追加、および追加ソートの機能を提供します。

グラフ属性構文では、グラフコンポーネントが欠落している場合に部分出力が表示されます。たとえば、棒グラフで Y 軸に割り当てられたフィールドが欠落している場合、棒の高さがすべて同じ高さになります。

JSON プロパティおよび API コールをグラフ属性構文と組み合わせて使用することで、グラフ出力の追加プロパティを制御することができます。

トピックス

- グラフ属性構文の概念
- グラフ属性構文のグラフタイプ
- グラフ方向の指定
- 棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定
- グラフデータの属性カテゴリへの割り当て
- ソートフィールドに対する属性カテゴリの割り当て順序
- グラフタイプ別のサポート可能な属性カテゴリの概要

グラフ属性構文の概念

グラフ属性構文では、WebFOCUS スタイルシートの TYPE=DATA 宣言を使用して、リクエスト内のフィールドを属性カテゴリに割り当てます。属性カテゴリの一例として、x-axis、y-axis、size、color があります。

使用できるカテゴリは、生成するグラフタイプ、リクエスト内のソートフィールドおよびメジャーフィールドの数によって異なります。ソートフィールドは、特定の順序でカテゴリに割り当てる必要があります。この順序は、リクエスト内のソートフィールドの順序および使用するカテゴリのタイプによって異なります。ソート順序および属性カテゴリについての詳細は、207 ページの 「 ソートフィールドに対する属性カテゴリの割り当て順序 」 を参照してください。

注意：この構文では、BY ソートフィールドのみがサポートされます。ACROSS ソートフィールドはサポートされません。

属性カテゴリへのフィールドの割り当て方法についての理解

グラフ属性構文を使用する場合、WebFOCUS スタイルシートではリクエスト内の各フィールドに宣言 (TYPE=DATA 宣言) が含まれます。これらの宣言により、フィールドが適切な属性カテゴリに割り当てられます。スタイルシートには、グラフの方向とレイアウト (棒グラフ、折れ線グラフ、面グラフの場合) を指定するグローバル変数の宣言 (TYPE=REPORT) も含まれます。

構文についての詳細は、197 ページの 「 グラフデータの属性カテゴリへの割り当て 」 を参照してください。グラフの方向およびレイアウトについての詳細は、191 ページの 「 グラフ方向の指定 」 および 193 ページの 「 棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定 」 を参照してください。

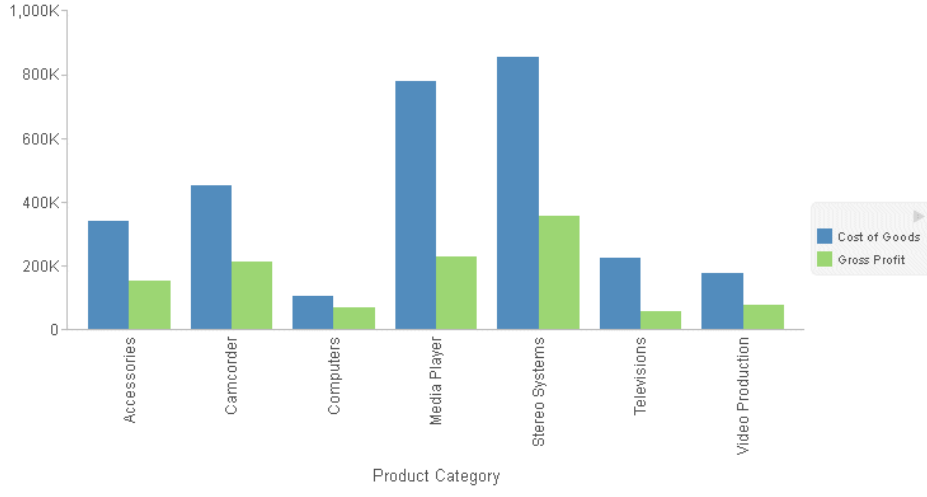
以下は、グラフ属性構文を使用した単純な棒グラフの例です。

例　グラフ属性構文を使用した棒グラフの例

次のリクエストは、棒グラフを生成します。ソートフィールド (PRODUCT_CATEGORY) は X 軸に割り当てられ、2 つのメジャー (COGS_US および GROSS_PROFIT_US) は Y 軸に割り当てられます。グラフタイプは BAR (棒グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



参照　列順序番号についての理解

WebFOCUS リクエストが生成する出力データレコードでは、すべての列に順序番号が付けられています。リクエスト内のフィールドは、出力結果に複数列を生成する場合も、出力結果に表示されない場合もあります。これらの余分な列についての説明はここでは省略します。

これらの列順序番号を使用して、WebFOCUS スタイルシートのフィールドを参照することができます。最上位ソートのソートフィールドを先頭に、ソートフィールド、メジャーフィールドの順で続きます。スタイルシートではこれらのフィールドは、N1、N2、N3...と呼ばれます。

注意：WebFOCUS ツールを使用してグラフを作成する場合は、フィールドをカテゴリに割り当てる際にフィールド名ではなく列順序番号が使用されます。

列順序の決定方法を理解するため、次の簡単なリクエストについて考察します。リクエストには 3 つのソートフィールドと 3 つのメジャーフィールドが含まれます。

```
TABLE FILE GGSALES
HEADING
"N1           N2           N3                N4             N5              N6"
SUM UNITS AS 'First Measure, Units'
DOLLARS AS 'Second Measure, Dollars'
BUDDOLLARS AS 'Third Measure, Budget Dollars'
BY CATEGORY AS 'First Sort, Category'
BY REGION AS 'Second Sort, Region'
BY PRODUCT AS 'Third Sort, Product'
ON TABLE SET PAGE NOPAGE
ON TABLE PCHOLD FORMAT WP
END
```

以下の部分出力は、出力結果上のフィールドの配置、および各フィールドに割り当てられた順序番号を示しています。1 番目の (上位) ソートフィールドは列番号 1、2 番目のソートフィールドは列番号 2 と順に続きます。ソートフィールドがすべて表示された後に、メジャーフィールドがリクエスト内の順序に従って表示されます。

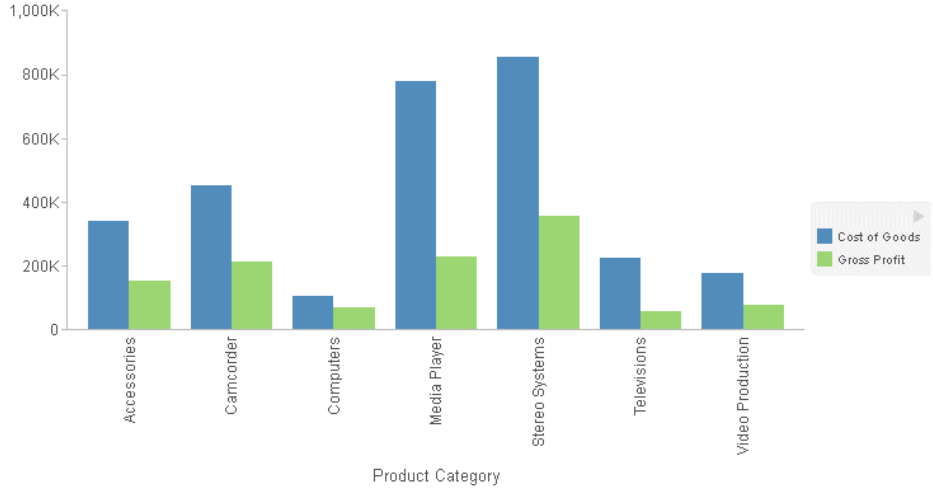
N1 First Sort Category	N2 Second Sort Region	N3 Third Sort Product	N4 First Measure Units	N5 Second Measure Dollars	N6 Third Measure Budget Dollars
Coffee	Midwest	Espresso	101154	1294947	1258232
		Latte	231623	2883566	2827800
	Northeast	Capuccino	44785	542095	561491
		Espresso	68127	850107	872902
		Latte	222866	2771815	2818069
	Southeast	Capuccino	73264	944000	956661
		Espresso	68030	853572	849465
		Latte	209654	2617836	2625303
	West	Capuccino	71168	895495	877304
		Espresso	71675	907617	923941
		Latte	213920	2670405	2722718
Food	Midwest	Biscotti	86105	1091727	1067629
		Croissant	139182	1751124	1708733
		Scone	116127	1495420	1444359
	Northeast	Biscotti	145242	1802005	1848682
		Croissant	137394	1670818	1739522
		Scone	70732	907171	865703
	Southeast	Biscotti	119594	1505717	1512019
		Croissant	156456	1902359	1969906
		Scone	73779	900655	927363
	West	Biscotti	70436	863868	861804
		Croissant	197022	2425601	2406554
		Scone	72776	912868	914886
Gifts	Midwest	Coffee Grinder	50393	619154	613453
		Coffee Pot	47156	599878	614007
		Mug	86718	1086943	1096150
		Thermos	46587	577906	564010
	Northeast	Coffee Grinder	40977	509200	511642
		Coffee Pot	46185	590780	573349
		Mug	91497	1144211	1170314
		Thermos	48870	604098	615247
	Southeast	Coffee Grinder	47083	605777	569585
		Coffee Pot	49922	645303	654579
		Mug	88474	1102703	1124345
		Thermos	48976	632457	618745

例　列順序番号を使用した棒グラフの例

次のリクエストは、先の例と同じ棒グラフを、列順序番号を使用して生成します。ソートフィールド (PRODUCT_CATEGORY、field N1) は X 軸 に割り当てられ、各メジャーフィールドは Y 軸 に割り当てられます。グラフタイプは BAR (棒グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



サーバ生成プロパティの上書き

リクエストにグラフ属性構文を使用する場合は、出力結果に表示されるテキストの一部が WebFOCUS Reporting Server によって生成されます。たとえば、サーバは行ラベル、列ラベル、X 軸ラベル、ツールヒントを生成します。サーバは、グラフエンジンによるグラフ描画終了後にこれらのラベルを生成します。

リクエスト内のデフォルト設定の行ラベル、列ラベル、X 軸ラベルは、AS 句 (例、SUM COGS_US AS 'Cost') を使用して変更することもできます。これにより、サーバはデフォルト設定のラベルではなくカスタムラベルを使用します。

また、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックに該当するプロパティを挿入しても、サーバ生成のプロパティを上書きすることができます。ツールヒントなど一部のグラフコンポーネントについては、WebFOCUS スタイルシートに *GRAPH_JS_FINAL ブロックを追加しないとデフォルト設定のサーバ生成プロパティを上書きすることができません。これにより、サーバ処理の完了後に *GRAPH_JS_FINAL ブロックのプロパティが返されるようグラフエンジンに指示します。

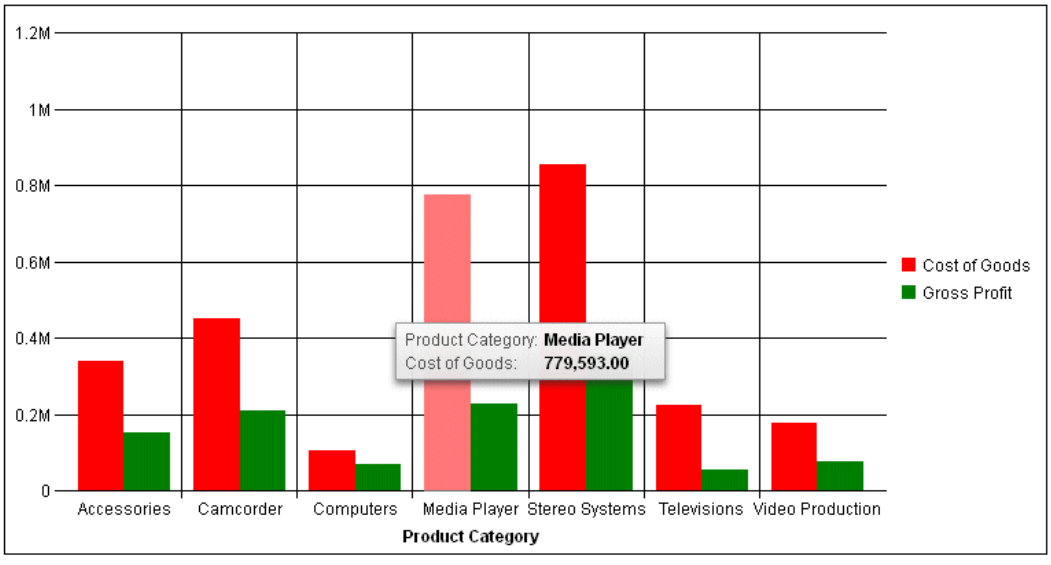
WebFOCUS スタイルシートでは、複数の *GRAPH_JS および *GRAPH_JS_FINAL ブロックを含めることができます。*GRAPH_JS ブロックは、サーバの処理前にスタイルシート内の順序に従って処理されます。*GRAPH_JS_FINAL ブロックは、サーバ処理の完了後にスタイルシート内の順序に従って処理されます。

例　グラフ属性構文を使用したカスタムツールヒントの生成

次のリクエストは、カスタムツールヒントを含む棒グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US , BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"series": [
    {"series": "reset", "tooltip": "This is my custom tooltip"}
]
*END
ENDSTYLE
END
```

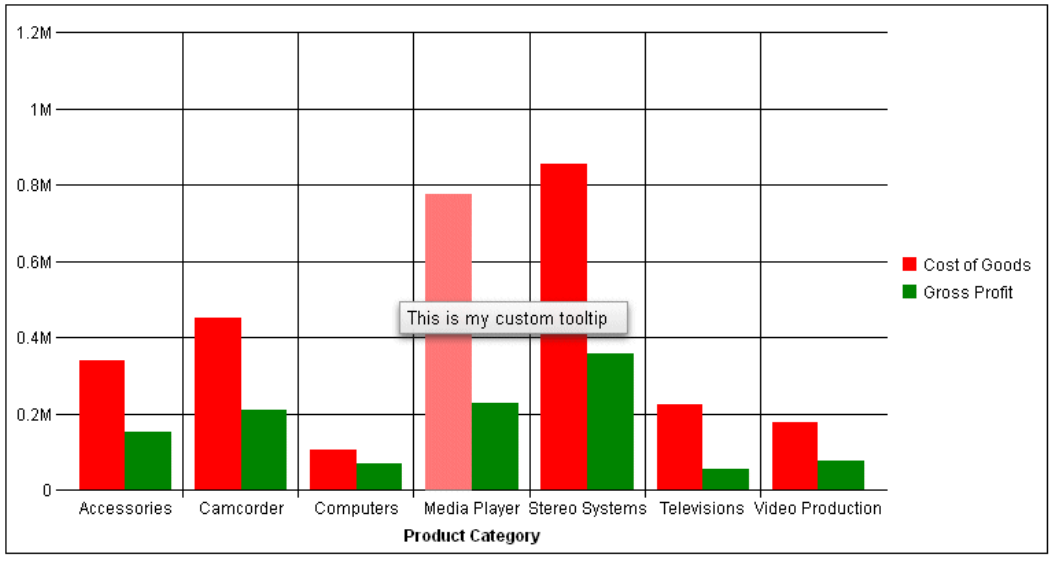
下図のように、カスタムツールヒントのテキストはサーバ生成のプロパティによって上書きされます。



以下は同じリクエストですが、WebFOCUS スタイルシートで *GRAPH_JS ブロックの代わりに *GRAPH_JS_FINAL ブロックを使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US , BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS_FINAL
"series": [
    {"series": "reset", "tooltip": "This is my custom tooltip"}
]
*END
ENDSTYLE
END
```

下図のようにカスタムツールヒントが表示されます。



また、マクロを使用してグラフテンプレートエンジンでツールヒントをカスタマイズすることもできます。詳細は、61 ページの「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

マトリックス配列での複数グラフの表示

リクエスト内のフィールドを特定のグラフコンポーネントに割り当てる以外に、row カテゴリと column カテゴリにフィールドを割り当てることで、複数グラフのマトリックスを生成することができます。

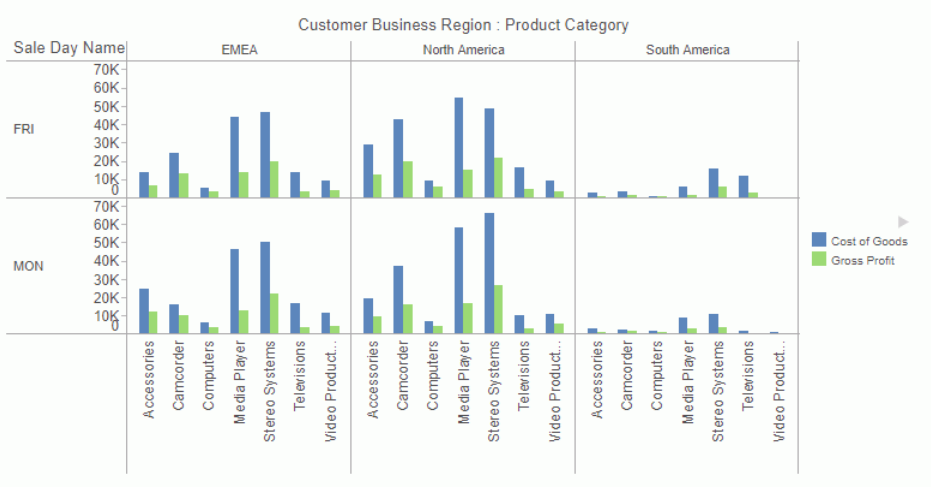
複数のフィールドを行と列の属性カテゴリに割り当てても、行または列のみを指定しても、細かく調整したグラフグループを生成することができます。

例　棒グラフマトリックスの生成

次のリクエストは、ビジネス地方区分 (BUSINESS_REGION) と 曜日 (TIME_DAYNAME) の組み合わせごとに新しい棒グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックスマーカーグラフ

マトリックスグラフはすべてグラフ属性構文を使用することが求められますが、マトリックスマーカーグラフと呼ばれるタイプのグラフは、特にグラフ属性構文のマトリックス機能を使用するために開発されたものです。

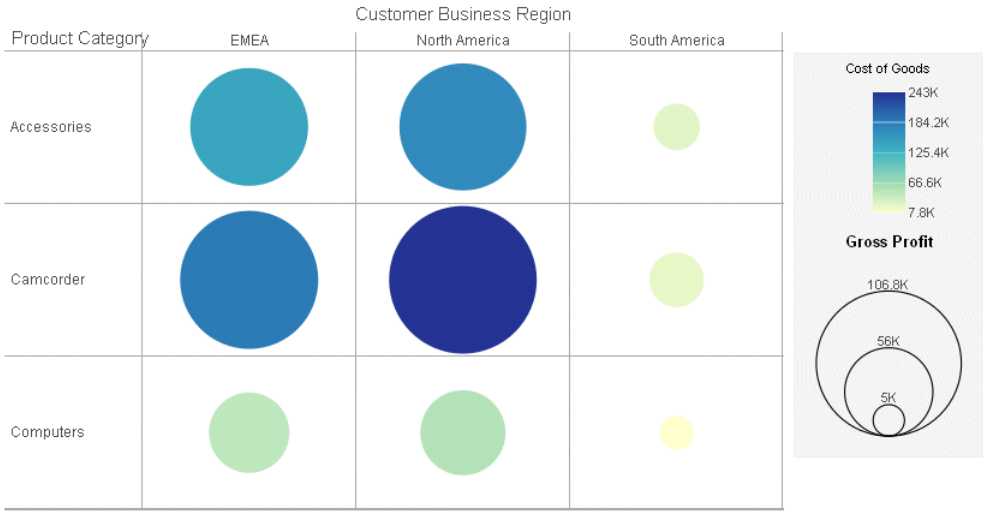
マトリックスマーカーグラフは、表形式のデータをグラフで表現したものです。このグラフでは、マトリックスの各セルにマーカーが表示されます。1 つ目のメジャーに基づいてマーカーのサイズが決定され、2 つ目のメジャーに基づいてマーカーの色が決定されます。
LOOKGRAPH 値は、MARKER です。

例　グラフ属性構文を使用したマトリックスマーカーグラフの生成

次のリクエストは、マトリックスマーカーグラフを生成します。行は、製品区分 (PRODUCT_CATEGORY) を示します。列は、ビジネス地方区分 (BUSINESS_REGION) を示します。各セルのマーカーは、粗利益 (GROSS_PROFIT_US) のメジャーに基づいてサイズが決定され、売上原価 (COGS_US) のメジャーに基づいて色が決定されます。color カテゴリに割り当てられたフィールドがメジャーである場合、凡例はグラデーションとして表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
WHERE PRODUCT_CATEGORY LE 'D'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH MARKER
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=size, $
ENDSTYLE
END
```

出力結果は次のとおりです。



データグリッドの生成

データグリッドは、表形式のレポートのようにデータを表形式で表示します。グリッドは、複数の行、列、メジャーで構成することができます。データグリッドは、WebFOCUS TABLE FILE コマンドで生成される表形式レポートと異なり、PCHOLD FORMAT JSCHART を使用した GRAPH FILE リクエストで生成されます。LOOKGRAPH 値は、DATAGRID です。

データグリッドのプロパティを使用して、グリッドのフォーマットを設定したり、グリッド内のメジャーの列合計を表示したりすることができます。データグリッドの行と列には、ゼロ (0) から始まる番号が付けられます。データグリッドのフォーマットで使用可能なプロパティについての詳細は、513 ページの 「 グラフ固有のプロパティ 」 を参照してください。

例　データグリッドの生成

次のリクエストは、データグリッドを生成し、列合計を表示します。行は、製品区分 (PRODUCT_CATEGORY) と製品区分 (詳細) (PRODUCT_SUBCATEG) を示します。列は、当該年度の第 1 四半期と第 2 四半期、および北米と南米のビジネス地方区分を示します。メジャーは、売上原価 (COGS_US) と粗利益 (GROSS_PROFIT_US) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY TIME_QTR
BY BUSINESS_REGION
WHERE (BUSINESS_REGION NE 'Oceania' OR 'EMEA') AND (TIME_QTR EQ 1 OR 2)
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=row, $
type=data, column=product_subcateg, bucket=row, $
type=data, column=time_qtr, bucket=column, $
type=data, column=business_region, bucket=column, $
type=data, column=cogs_us, bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {
    "columnTotals": {"visible": true}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Sale Quarter		1				2			
Customer Business Region		North America		South America		North America		South America	
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Prof
Accessories	Charger	$1,445.00	$1,441.10	$170.00	$157.98	$945.00	$901.25	$232.00	$242.9
	Headphones	$23,243.00	$11,691.55	$2,659.00	$1,551.87	$23,680.00	$12,076.97	$5,266.00	$1,987.5
	Universal Remote Controls	$20,671.00	$7,428.05	$5,034.00	$1,577.96	$19,995.00	$8,044.73	$2,559.00	$695.8
Camcorder	Handheld	$12,172.00	$12,772.12	$3,170.00	$3,172.94	$12,692.00	$13,031.36	$1,644.00	$1,411.6
	Professional	$23,570.00	$7,811.15			$31,030.00	$1,850.75		
	Standard	$31,199.00	$13,095.73	$7,277.00	$3,065.45	$36,694.00	$13,544.14	$3,113.00	$1,187.1
Computers	Smartphone	$15,052.00	$8,932.73	$3,720.00	$2,514.65	$15,568.00	$10,256.59	$1,321.00	$679.8
Media Player	Blu Ray	$113,791.00	$34,313.74	$25,734.00	$7,551.09	$107,000.00	$29,846.63	$16,449.00	$4,340.9
	DVD Players	$7,743.00	$4,663.02	$2,184.00	$1,012.87				
	Streaming	$1,748.00	$1,187.62	$184.00	$135.96	$1,380.00	$983.70	$46.00	$33.9
Stereo Systems	Home Theater Systems	$36,291.00	$17,731.46	$7,527.00	$3,509.50	$37,369.00	$18,277.90	$4,609.00	$2,183.6
	Receivers	$20,082.00	$8,774.40	$3,992.00	$1,425.23	$23,382.00	$8,922.36	$6,542.00	$2,709.4
	Speaker Kits	$54,032.00	$17,453.00	$12,007.00	$4,001.42	$45,503.00	$14,251.98	$11,014.00	$3,570.9
	iPod Docking Station	$17,786.00	$10,859.56	$2,888.00	$1,373.32	$17,902.00	$9,982.36	$2,907.00	$1,770.5
Televisions	Flat Panel TV	$29,889.00	$8,931.25	$3,829.00	$883.92	$33,167.00	$9,311.83	$15,847.00	$4,310.9
Video Production	Video Editing	$27,278.00	$12,561.31	$3,026.00	$1,550.34	$29,025.00	$11,446.98	$4,592.00	$1,760.2
TOTAL		$435,992.00	$179,647.79	$83,401.00	$33,484.50	$435,332.00	$162,729.53	$76,141.00	$26,885.8

バブルグラフまたは散布図への詳細レベルの追加

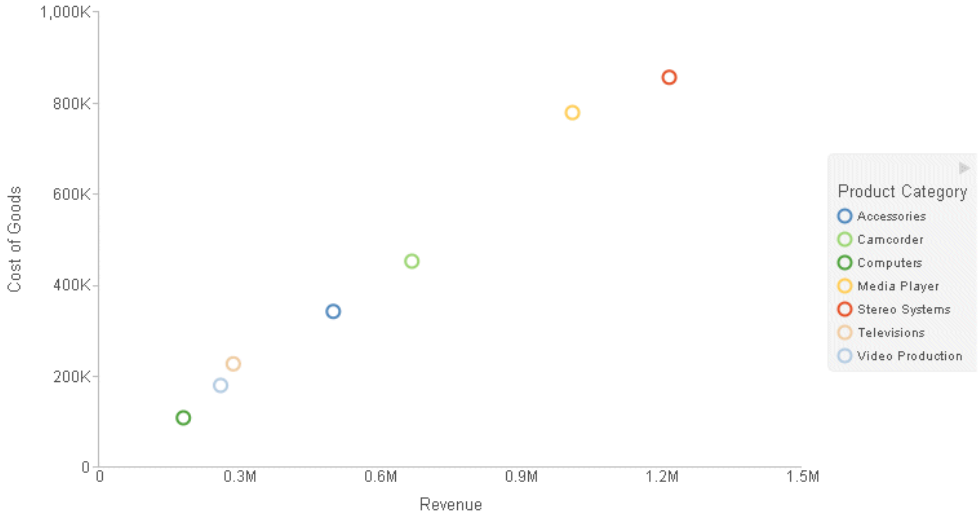
バブルグラフおよび散布図では、下位ソートフィールドをリクエストに追加し、そのフィールドを detail カテゴリに割り当てることができます。この詳細ソートフィールドを追加することで、さらに多くのデータポイントがグラフに表示されます。

例　　散布図への詳細カテゴリの追加

次のリクエストは、散布図を生成し、収益 (REVENUE_US) に対する売上原価 (COGS_US) を示します。マーカーは製品区分 (PRODUCT_CATEGORY) ごとに色分けされます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=x-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
ENDSTYLE
END
```

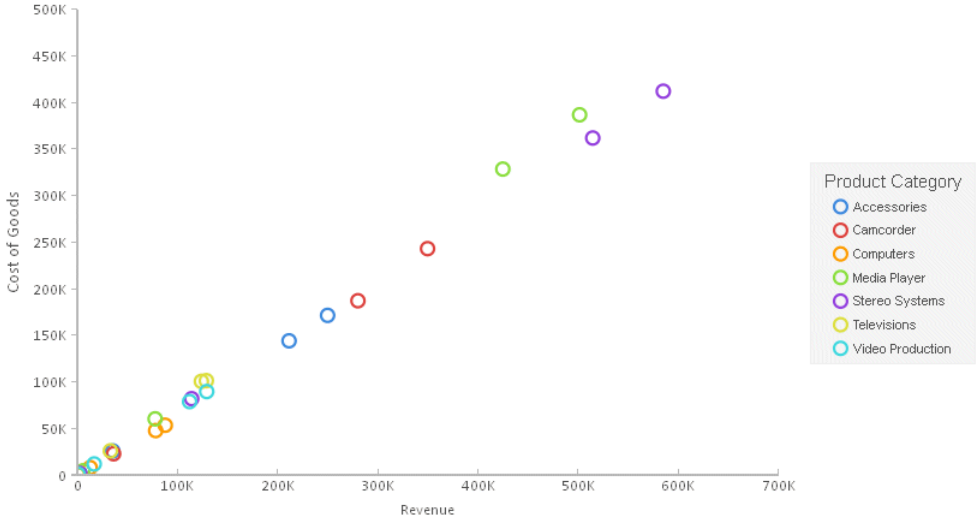
出力結果は次のとおりです。



下位ソートフィールドとしてビジネス地方区分 (BUSINESS_REGION) を追加し、detail カテゴリに割り当てることで、このグラフに詳細レベルを追加することができます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
REVENUE_US
BY PRODUCT_CATEGORY
  BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=x-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=detail, $
ENDSTYLE
END
```

出力結果は次のとおりです。



色フィールドおよびサイズフィールドの追加

バブルグラフなど一部のグラフタイプでは、自動的に size カテゴリを使用します。バブルの大きさは、size カテゴリに割り当てられたメジャー値によって異なります。

グラフタイプによっては、自動的に size カテゴリを使用しないものもあります。たとえば、棒グラフの棒の高さはメジャーのサイズで表します。棒の高さを表すメジャーは、y-axis カテゴリに割り当てられています。

グラフ属性構文では、メジャーを追加してこれを size カテゴリに割り当てることができます。この場合、サイズは棒の幅に反映されます。

通常、棒にはシリーズに基づいて色も割り当てられています。グラフ属性構文では、フィールドを追加してこれを color カテゴリに割り当て、追加したフィールドの値に基づいて棒を色分けすることができます。色フィールドがメジャーの場合、棒と凡例の色はグラデーションとして表示されます。色フィールドがディメンションの場合、棒と凡例の色は不連続の色で表示されます。

折れ線グラフでも、グラフ属性構文を使用して size カテゴリと color カテゴリを追加することができます。マーカーが表示されている場合、size メジャー値はマーカーのサイズで表されます。マーカーが表示されていない場合、size メジャー値は線の幅で表されます。色フィールドは、メジャーまたはディメンションフィールドにすることができます。色は、マーカー、線、凡例に反映されます。色フィールドがメジャーの場合、色はグラデーションとして表示されます。色フィールドがディメンションの場合、色は不連続の色で表示されます。

マトリックス配列で表示される円グラフの場合も、リクエストにメジャーを追加し、size カテゴリに割り当てることでサイズを決定することができます。マトリックスの各セル内の円グラフのサイズは、このように追加されたメジャーの値を表します。

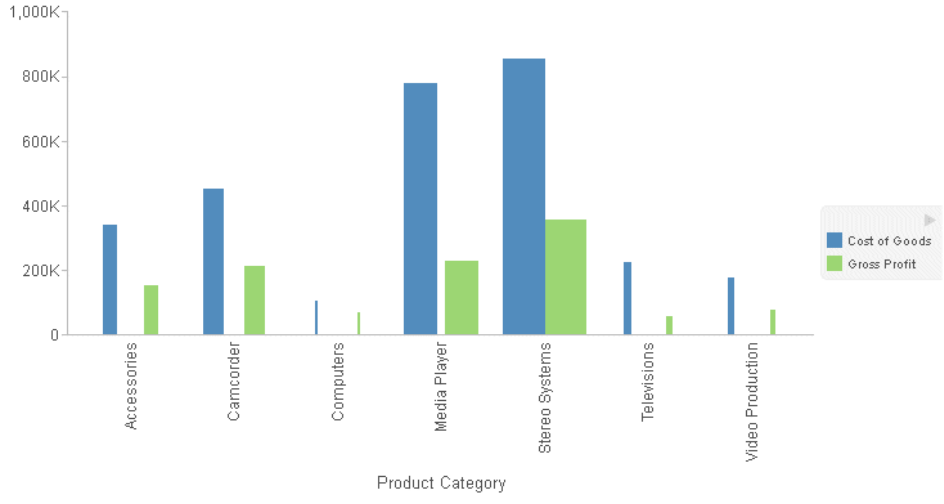
この機能がサポートされるグラフタイプについての詳細は、153 ページの 「 グラフ属性構文のグラフタイプ 」 を参照してください。各グラフタイプでサポートされる属性カテゴリのリストは、208 ページの 「 グラフタイプ別のサポート可能な属性カテゴリの概要 」 を参照してください。

例　棒グラフでのサイズメジャーの使用

次のリクエストは、棒グラフを生成します。棒の高さは COGS_US および GROSS_PROFIT_US メジャーを表し、棒の幅は MSRP_US メジャーを表します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US  MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=MSRP_US, BUCKET=size, $
ENDSTYLE
END
```

出力結果は次のとおりです。

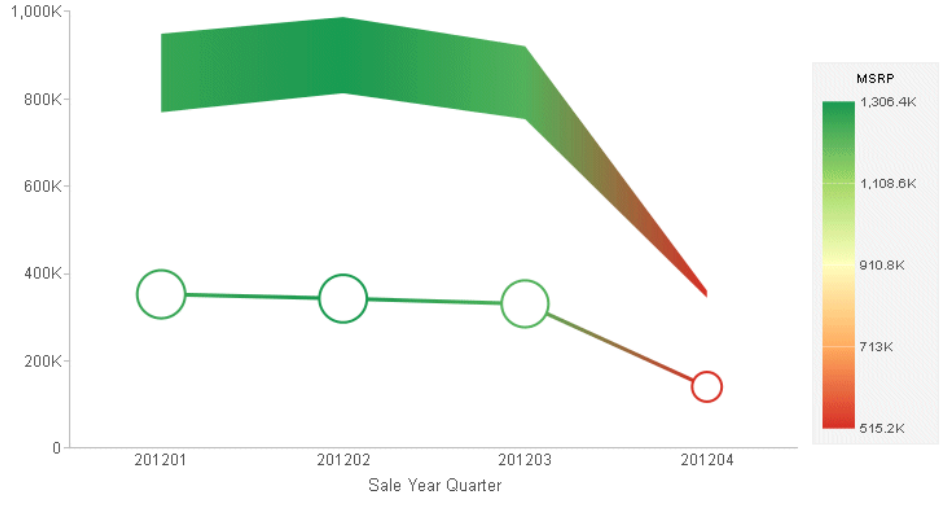


例　折れ線グラフでの色メジャーおよびサイズメジャーの使用

次のリクエストは、折れ線グラフを生成します。線およびマーカーは MSRP_US メジャーに基づいて色分けされます。シリーズ 1 のマーカーは、SIZE1 一時項目 (メジャー) に基づいてサイズが決定されます。シリーズ 0 については、マーカーが表示されていないため、線の幅が SIZE1 一時項目 (メジャー) を表します。

```
DEFINE FILE WF_RETAIL_LITE
SIZE1 WITH WF_RETAIL_SALES.COGS_US =
    IF WF_RETAIL_TIME_SALES.TIME_YEARQTR LT 200902 THEN 5
    ELSE IF WF_RETAIL_TIME_SALES.TIME_YEARQTR LT 200903 THEN 50
    ELSE 200;
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US MSRP_US SIZE1
BY WF_RETAIL_TIME_SALES.TIME_YEARQTR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=WF_RETAIL_TIME_SALES.TIME_YEARQTR, BUCKET=x-axis, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=MSRP_US, BUCKET=color, $
TYPE=DATA, COLUMN=SIZE1, BUCKET=size, $
*GRAPH_JS
"series": [
    {"series": 0, "marker": {"visible": false}},
    {"series": 1, "marker": {"visible": true, "shape": "circle"}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

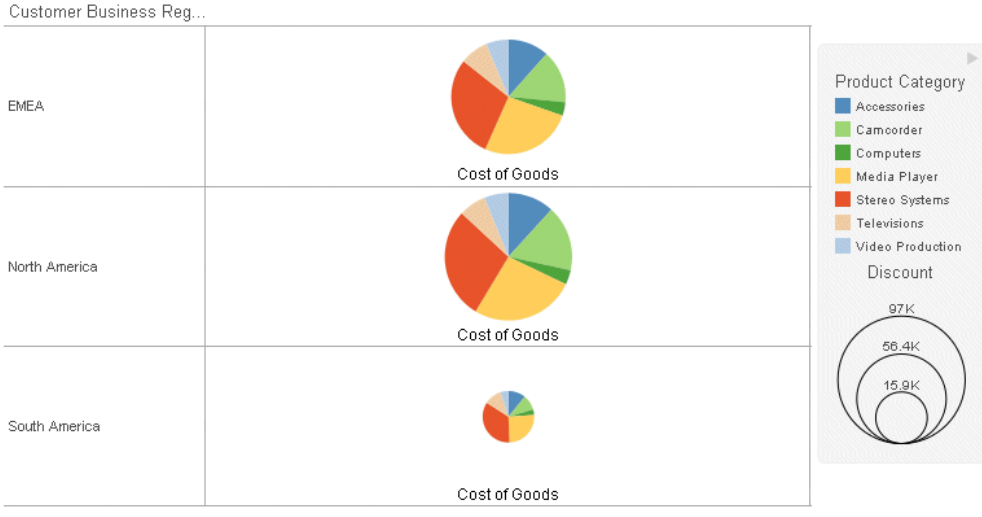


例　マトリックスでの円グラフのサイズ設定

次のリクエストは、マトリックスで円グラフを生成します。マトリックスの行がビジネス地方区分 (BUSINESS_REGION) を表します。各行の円グラフは、DISCOUNT_US メジャーに基づいてサイズが決定されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=row, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=measure, $
TYPE=DATA, COLUMN=DISCOUNT_US, BUCKET=size, $
ENDSTYLE
END
```

出力結果は次のとおりです。



スライダコントロールを使用した時間推移のアニメーション化

グラフのスライダコントロールでは、スライダバーコントロールの移動に合わせて動作するアニメーション効果を作成することができます。スライダバー上でコントロールをドラッグすることも、スライダ再生ボタンをクリックしてスライダが値から値へ自動的に移動するように設定することもできます。スライダは、1 つのソートフィールドのみに制限され、YEAR などの時間に関連するフィールドに適しています。

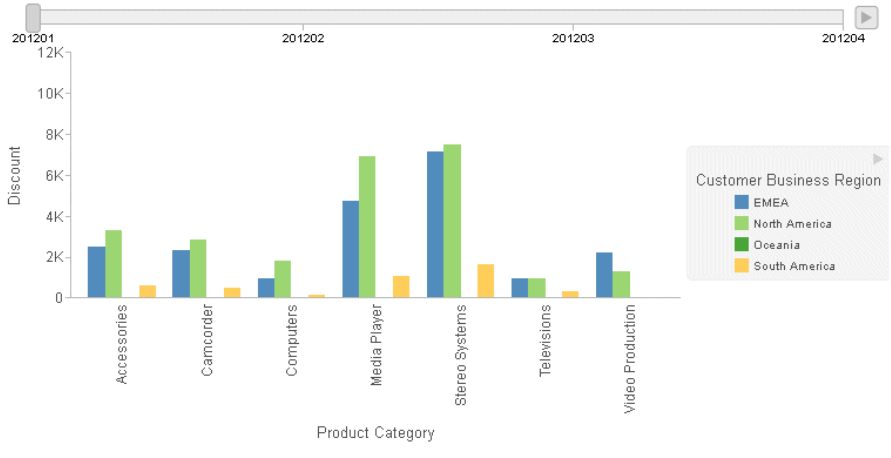
再生ボタンをクリックした際のスライダの移動速度を制御することができます。詳細は、769 ページの 「 その他の機能 」 を参照してください。

例　グラフへのスライダコントロールの追加

次のリクエストは、製品区分 (PRODUCT_CATEGORY) ごとに 1 つのメジャーを持つ棒グラフを生成します。TIME_YEARQTR フィールドにはスライダコントロールが割り当てられ、BUSINESS_REGION ディメンションには色フィールドが割り当てられています。これらのフィールドは、最上位ソートフィールドとしてリクエストに追加されています。スライダを各四半期にドラッグするたびに、新しい棒グラフが生成されます。

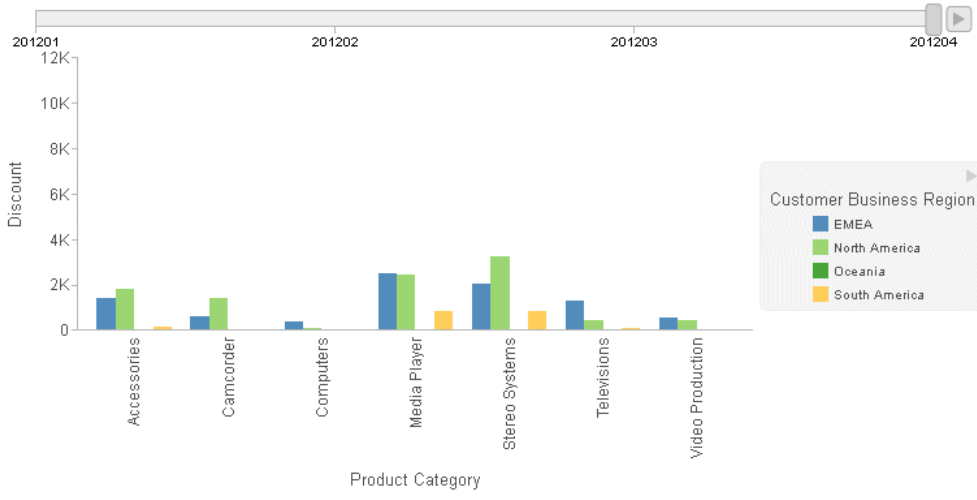
```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US
BY WF_RETAIL_TIME_SALES.TIME_YEARQTR
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=DISCOUNT_US , BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
TYPE=DATA, COLUMN=WF_RETAIL_TIME_SALES.TIME_YEARQTR, BUCKET=slider, $
ENDSTYLE
END
```

出力結果は次のとおりです。スライダは最初に一番左側に配置されます。



ソートフィールド値の変化をアニメーションで見るには、コントロールをクリックしてドラッグするか、右矢印をクリックします。

下図は、第 4 四半期までコントロールをドラッグした状態を示しています。



ツールヒントへのフィールドの追加

グラフコンポーネントを表すフィールドは、デフォルト設定のツールヒントに自動的に含まれます。tooltip 属性カテゴリを使用することで、ツールヒントにフィールドを追加することができます。

tooltip の値のフォーマットは、メタデータおよびリクエストのフィールドフォーマットから取得されます。

tooltip 属性カテゴリには、任意の順序で項目を追加することができます。これらの項目は、自動的に表示されるフィールド以下に、tooltip カテゴリに追加した順序に従って表示されます。

tooltip 属性カテゴリにソートフィールドを追加した場合、ツールヒントに表示されるソートフィールドの値は、ツールヒントが表示されるライザの値と一致します。

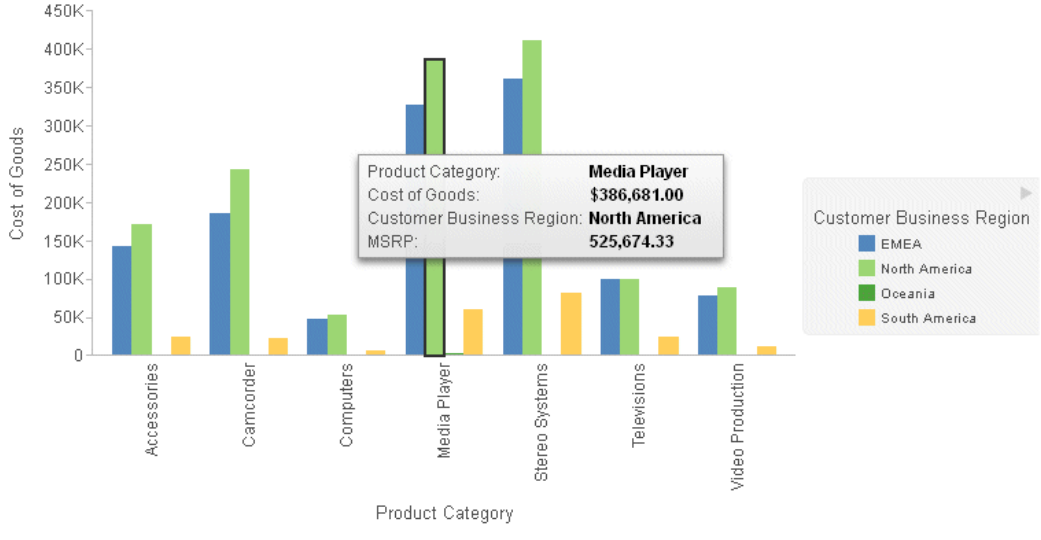
注意: グラフコンポーネントを表すソートフィールドについてはツールヒントに自動的に含まれるため、追加する必要はありません。これには page、row、column の属性カテゴリに割り当てられるソートフィールドが含まれます。

例　ツールヒントへのフィールドの追加

次の棒グラフは、各製品区分 (PRODUCT_CATEGORY) の売上原価 (COGS_US) を示しており、ビジネス地方区分 (BUSINESS_REGION) ごとに色分けされています。ツールヒントには、希望小売価格 (MSRP_US) が追加されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=MSRP_US, BUCKET=tooltip,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。MSRP の値はツールヒントに表示されますが、グラフ上の棒としては表示されません。



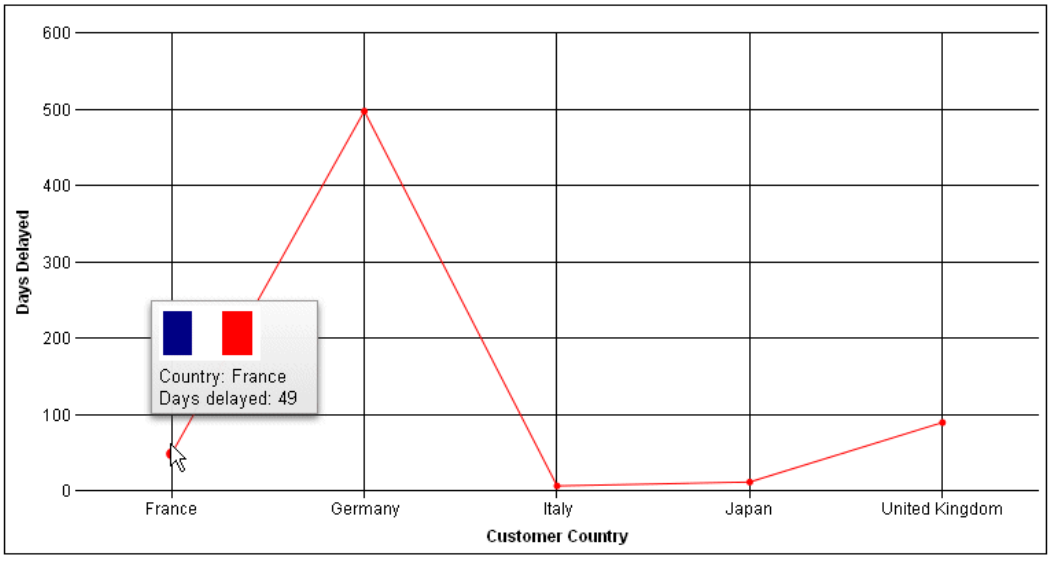
例　グラフテンプレートエンジンを使用したツールヒントへの画像の追加

次のリクエストは、ツールヒントに各国の国旗の画像を追加します。この場合、画像の名前とデータソースの国名を関連付けるために、一時項目 (DEFINE) を作成します。これらが同一の場合、このステップは不要です。リクエスト内の tooltip 属性カテゴリに、FLAG という名前の一時項目 (DEFINE) が追加されます。この項目は tooltip カテゴリの 1 番目のフィールドであるため、この値はマクロ {{tooltip1}} を使用してツールヒントに挿入することができます。グラフテンプレートエンジンおよび組み込みマクロについての詳細は、61 ページの 「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

HTML img タグが、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックのカスタムツールヒントに追加されます。

```
DEFINE FILE WF_RETAIL_LITE
FLAG/A12=
DECODE COUNTRY_NAME ( 'United Kingdom' 'uk' 'Germany' 'germany'
    'Italy' 'italy' 'France' 'france' 'Japan' 'japan' );
END
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED FLAG
BY COUNTRY_NAME
WHERE COUNTRY_NAME EQ 'United Kingdom' OR 'Italy' OR 'Germany' OR 'France' OR
'Japan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=country_name, bucket=x-axis, $
type=data, column=n2, bucket=y-axis, $
type=data, column=flag, bucket=tooltip, $
*GRAPH_JS_FINAL
"series": [{"series": 0,
    "tooltip": "<img src=¥"http://ecl.informationbuilders.com/jschart/
        {{tooltip1}}.gif¥"><br>Country: {{group_label}}<br>Days delayed: {{value}}"
    }]
END
```

下図では、フランスに関するツールヒントが表示されています。



ドリルダウンリンク付き画像の見出しへの追加

グラフ属性構文では、埋め込み見出しに HTML フィールドを使用することができます。埋め込み見出しは、ON GRAPH SET EMBEDHEADING ON コマンドで実行されます。

HTML フィールドは、埋め込み見出しに含めることができます。この場合、埋め込み見出しとして HTML のグラフタイトルが生成されます。

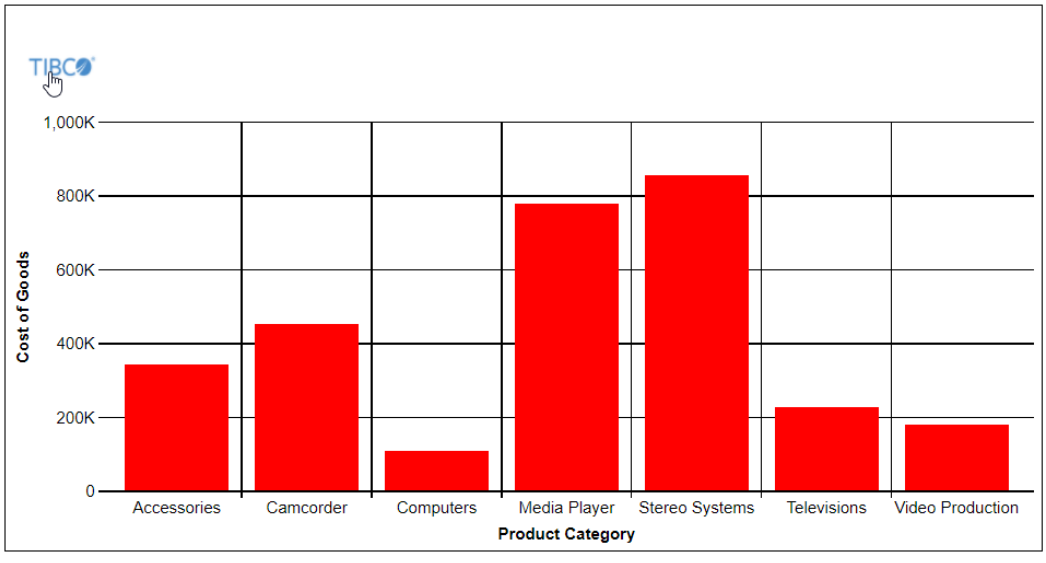
例　リンク付き画像のグラフタイトルへの追加

次のリクエストは、HTML 画像および TIBCO Web サイトへのリンクを含むフィールドを定義します。次にこのフィールドを見出しに追加します。見出しは、ON GRAPH SET EMBEDHEADING ON コマンドを使用してグラフに埋め込まれます。

注意：以下の例では、この文書のページ幅によって HTML コードが複数行に折り返されています。このサンプルリクエストを実行する場合は、一重引用符 (') 内の改行を削除する必要があります。

```
DEFINE FILE WF_RETAIL_LITE
IMG_TAG/A300 =
'<a href="http://www.tibco.com"><img src="http://ecl.informationbuilders.com/jschart/
tibco.png" width=70 height=40></a>';
END
GRAPH FILE WF_RETAIL_LITE
HEADING
" "
"<IMG_TAG"
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=x-axis, $
type=data, column=cogs_us, bucket=y-axis, $
END
```

出力結果は次のとおりです。マウスポインタを画像上に置くと表示される指のカーソルは、この画像にリンクが設定されていることを示します。



グラフのフィールド値への色のバインド

色のバインドを使用すると、BY フィールド値を特定の色に関連付け (バインド) できるため、フィールド値の順序やフィルタ条件に関係なく、複数のグラフで同一値が同一色で表示されます。色のバインドを指定する BY フィールドは、color 属性カテゴリに割り当てる必要があります。たとえば、BY フィールドが COUNTRY の場合、このフィールドを color 属性カテゴリに割り当てます。次に、レポートの各ページのすべてのグラフで、COUNTRY フィールド値の「France」が赤色で表示されるよう指定します。凡例マーカーも、この色にバインドされます。

構文　フィールド値への色のバインド

```
type=data, marker-color=color, when=field eq value, $
```

説明

`color`

WHEN 句の条件が満たされた場合にグラフおよび凡例マーカーに適用する色を指定します。色名、RGB 色値、16 進数色値がサポートされます。

`field`

color 属性カテゴリに割り当てる BY フィールドの名前です。このフィールドの値が特定の色にバインドされます。

`value`

marker-color 属性で指定された色にバインドする値です。

参照　色のバインドに関する注意

- 色のバインドは、color 属性カテゴリに割り当てられたフィールドが 1 つの場合にのみサポートされます。
- WHEN 条件でサポートされる演算子は EQ のみです。
- 色のバインドは、グラフ属性構文でのみサポートされます。
- COLUMN 属性は、一般に条件付きスタイルをグラフに適用する際に必要な属性ですが、色のバインドではオプションです。この属性を使用すると、複数のプロシジャで色マッピングの繰り返し使用が容易になります。
- 色が明示的に指定されていない BY 値が存在する場合、その値の色はスタイルシートで指定されているシリーズ色になります (スタイルシートが存在しない場合はグラフエンジンのデフォルト色)。

例　棒グラフのフィールド値への色のバインド

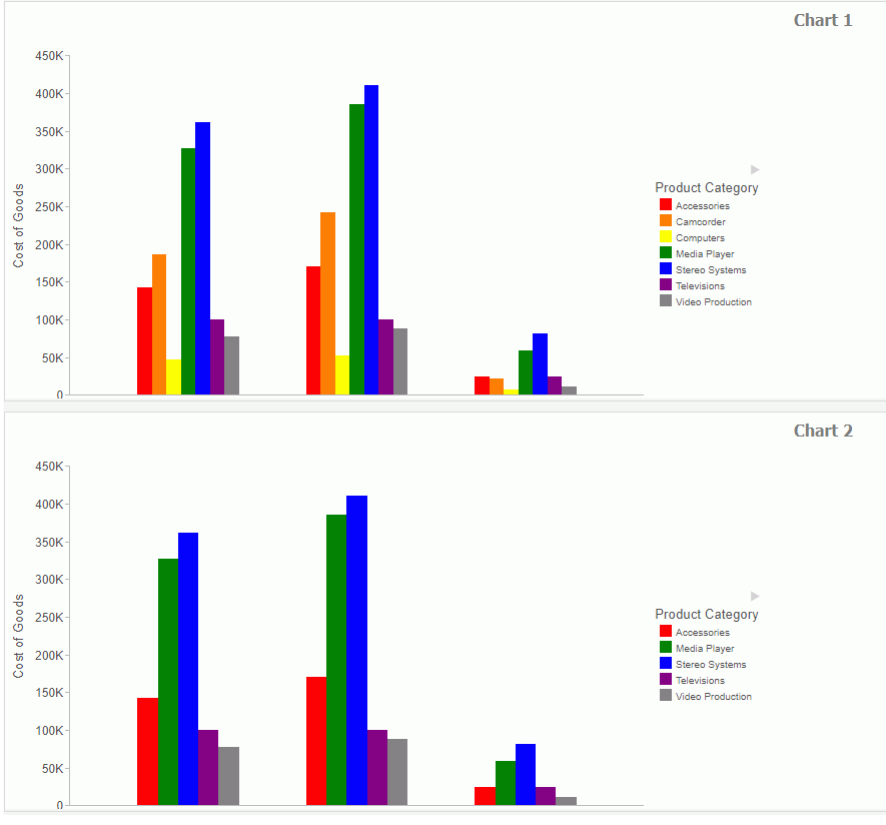
次のリクエストは、PRODUCT_CATEGORY フィールドの各フィールド値に特定の色をバインドします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=X-AXIS, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=COLOR, $
TYPE=DATA, MARKER-COLOR=red, WHEN=PRODUCT_CATEGORY EQ 'Accessories',$
TYPE=DATA, MARKER-COLOR=orange, WHEN=PRODUCT_CATEGORY EQ 'Camcorder',$
TYPE=DATA, MARKER-COLOR=yellow, WHEN=PRODUCT_CATEGORY EQ 'Computers',$
TYPE=DATA, MARKER-COLOR=green, WHEN=PRODUCT_CATEGORY EQ 'Media Player',$
TYPE=DATA, MARKER-COLOR=blue, WHEN=PRODUCT_CATEGORY EQ 'Stereo Systems',$
TYPE=DATA, MARKER-COLOR=purple, WHEN=PRODUCT_CATEGORY EQ 'Televisions',$
TYPE=DATA, MARKER-COLOR=gray, WHEN=PRODUCT_CATEGORY EQ 'Video Production',$
ENDSTYLE
END
```

次のリクエストは、PRODUCT_CATEGORY フィールドの各フィールド値に上記リクエストと同一の色をバインドしますが、値が「C」で始まる製品区分を除外します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
WHERE PRODUCT_CATEGORY NOT LIKE 'C%'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=X-AXIS, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=COLOR, $
TYPE=DATA, MARKER-COLOR=red, WHEN=PRODUCT_CATEGORY EQ 'Accessories',$
TYPE=DATA, MARKER-COLOR=orange, WHEN=PRODUCT_CATEGORY EQ 'Camcorder',$
TYPE=DATA, MARKER-COLOR=yellow, WHEN=PRODUCT_CATEGORY EQ 'Computers',$
TYPE=DATA, MARKER-COLOR=green, WHEN=PRODUCT_CATEGORY EQ 'Media Player',$
TYPE=DATA, MARKER-COLOR=blue, WHEN=PRODUCT_CATEGORY EQ 'Stereo Systems',$
TYPE=DATA, MARKER-COLOR=purple, WHEN=PRODUCT_CATEGORY EQ 'Televisions',$
TYPE=DATA, MARKER-COLOR=gray, WHEN=PRODUCT_CATEGORY EQ 'Video Production',$
ENDSTYLE
END
```

下図は、両方のグラフを示しています。Media Player はグラフ 1 では 4 つ目のライザとして、グラフ 2 では 2 つ目のライザとして表示されていますが、両方のライザ色は緑色です。



グラフ属性構文のグラフタイプ

グラフ属性構文を使用しない場合、縦方向と横方向ではグラフタイプが異なります。たとえば、縦棒グラフには VBAR が使用され、横棒グラフには HBAR が使用されます。

グラフ属性構文を使用した場合、グラフタイプ BAR は棒グラフのすべてのタイプに使用され、WebFOCUS スタイルシートの TYPE=REPORT 宣言の ORIENTATION 属性で縦棒グラフと横棒グラフが区別されます。詳細は、191 ページの 「 グラフ方向の指定 」 を参照してください。

積み上げ棒グラフ、折れ線グラフ、面グラフも同じように、WebFOCUS スタイルシートの TYPE=REPORT 宣言の CHART-SERIES-LAYOUT 属性を使用して、並列グラフと区別されます。詳細は、193 ページの 「 棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定 」 を参照してください。

以下は使用可能なグラフタイプ (LOOKGRAPH 値) です。

- **面グラフ (AREA)** 面グラフは、数値データの動きを表すという点では折れ線グラフと似ています。ただし、面グラフでは、データ線とゼロ線 (軸線) との間の領域に色またはテクスチャが適用されている点が異なります。
- **棒グラフ (BAR)** 棒グラフは、スケールを基準にした長方形のブロックを表示することによって、数値データを表します。棒の長さは、各データの値または数量に相当します。棒の長さを対比することにより、データ間の比較結果を明確なイメージとして頭の中に描くことができます。
- **ボックスプロット (BOXPLOT)** ボックスプロットは、データを四分位に分けて示します。ボックスプロットでは、各シリーズとグループに 5 つの値が必要です。特定のシリーズとグループボックスでは、1 つ目の値が最小値 (下側のハット)、2 つ目の値がボックス下端 (第 1 四分位) 、3 つ目の値が中央値 (第 2 四分位) 、4 つ目の値がボックス上端 (第 3 四分位) 、5 つ目の値が最大値 (上側のハット) をそれぞれ定義します。
- **バブルグラフ (BUBBLE)** バブルグラフは散布図を拡張したグラフで、3 つ目のメジャー値に比例して各マーカーのサイズが変化します。
- **バブルマップ (BUBBLEMAP)** バブルマップは、プロポーショナルシンボルマップグラフで、マーカー (バブル) は地理的なポイントやエリアに配置され、そのサイズはメジャー値に基づきます。また、メジャーまたはディメンションフィールドによって色分けされます。
- **コロプレスマップ (CHOROPLETH)** コロプレスマップは、地理位置を示す多角形の領域が、メジャーまたはディメンションに基づいて色分けされたマップグラフです。
- **データグリッド (DATAGRID)** データグリッドは、行、列、セルで表す表形式のグラフです。
- **じょうごグラフ (FUNNEL)** じょうごグラフは基本的には円グラフですが、一度に表示するデータグループは 1 つです。グループに属する各シリーズは、じょうご内に積み重ねられ、最初のシリーズが最上部に、最後のシリーズが最下部に表示されます。
- **メータグラフ (GAUGE)** メータグラフでは、一般に円形で表示される軸で単一値を識別します。メータグラフはダッシュボードでよく使用され、進捗率や数量を表示します。
- **ヒートマップ (HEATMAP)** ヒートマップは、データ値に応じて長方形をさまざまな色で表示し、これらの長方形を行と列のマトリックスで表したグラフです。
- **折れ線グラフ (LINE)** 折れ線グラフは、時間の経過とともに変化する数値データの動向や傾向を強調する際に役立ちます。これは、過去の状態を観察したり補間したりすることによって、特定の点の変動を追跡できるためです。高低のレベル、遅速の動き、安定に向けての傾向は、折れ線グラフが適するトレンドタイプです。

- **マトリックスマーカーグラフ (MARKER)** マトリックスマーカーグラフは、表形式のデータをグラフで表現したものです。このグラフでは、マトリックスの各セルにマーカーが表示されます。1 つ目のメジャーに基づいてマーカーのサイズが決定され、2 つ目のメジャーに基づいてマーカーの色が決定されます。
- **マリメッコチャート (MEKKO)** マリメッコチャート (別名、メッコチャート) は、パーセント棒グラフに類似していますが、積み上げ棒の全体値に基づいて棒ライザの幅が決定されるという点が異なります。ライザの幅は降順で自動的にソートされます。
- **円グラフ (PIE)** 円グラフは、特定のデータが全体の中でどの程度の割合を占めるかを強調します。全体に対する割合は、円項目のサイズで表されます。穴のサイズがゼロの場合は円グラフが生成されます。穴のサイズがゼロより大きい場合は、リンググラフが生成されます。
- **散布図 (SCATTER)** 散布図は、一般に数値メジャー間の関係を示します。散布図は、特定のポイント周辺のデータ値の密集度を視覚的に表示したり、データのパターンを明示したりする場合に使用します。グラフ属性構文では、ディメンションのほかメジャーもサポートされます。
- **統計ボックスプロット (STAT-BOXPLOT)** 統計ボックスプロットでは、四分位と異常値が 1 つ以上のメジャーと 1 つのディメンションから自動的に計算され、より簡単にボックスプロットが生成できます。グラフタイプは、STAT-BOXPLOT です。
- **ストリームグラフ (STREAM)** ストリームグラフは、積み上げ面グラフを簡素化したグラフです。ストリームグラフでは X 軸が無効化される場合が多く、罫線やフレームが省略されます。ストリームグラフでは、データテキストラベルは使用されません。
- **タグクラウド (TAGCLOUD)** タグクラウドは、グループのラベルをグラフ化したものです。データ値に比例して、各ラベルのサイズが変化します。タグクラウドはソーシャルメディアで使用され、語句の発生頻度を計測して感情を定量化します。
- **ツリーマップ (TREEMAP)** ツリーマップには、階層データが、ネストされた長方形の集合体として表示されます。

グラフ属性構文を使用したグラフの例

ここでは、グラフ属性構文をサポートする各グラフタイプについて簡単に説明し、グラフ例を示します。

グラフタイプごとに挙げた属性のほか、すべてのグラフはマトリックス配列での生成、ツールヒントへのフィールド追加、別ページとしての生成が可能です。また、一部のグラフタイプではスライダコントロールを備えたグラフ生成が可能です。

グラフ属性構文を使用した面グラフ

面グラフは、数値データの動きを示すという点で折れ線グラフに類似していますが、データ線とゼロの線 (軸線) との間の領域に色またはテクスチャが適用されている点が異なります。面グラフでは、1 つのデータの上に別のデータを積み上げることができます。積み上げにより、あるデータシリーズが 2 つ目のデータシリーズにどのように近づくか、追い越すかを表示し、データ間の関係を強調することができます。

グラフタイプは AREA (面グラフ) です。AREA には次の属性カテゴリがあります。

- **x-axis** - YEAR など通常は時間ベースのソートフィールドです。
- **y-axis** - メジャーです。
- **color** - エリアおよび凡例の色です (オプション)。

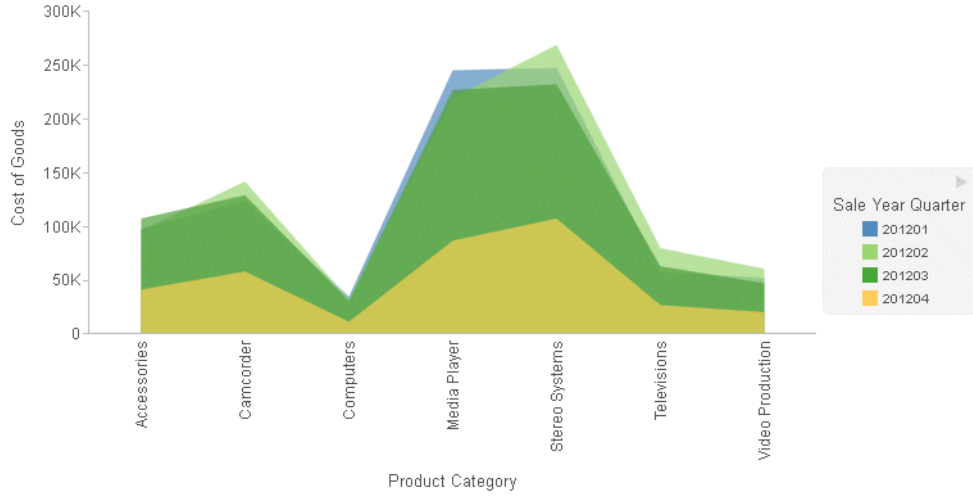
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。複数のグラフがサポートされます。

例　グラフ属性構文を使用した面グラフの例

次のリクエストは、面グラフを生成します。ソートフィールド (TIME_YEARQTR) は color、ソートフィールド (PRODUCT_CATEGORY) は x-axis のカテゴリに割り当てられ、メジャーフィールドは y-axis に割り当てられています。グラフタイプは AREA (面グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY WF_RETAIL_TIME_SALES.TIME_YEARQTR
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH AREA
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
TYPE=DATA, COLUMN=WF_RETAIL_TIME_SALES.TIME_YEARQTR, BUCKET=color, $
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用した棒グラフ

棒グラフは、スケールを基準にした長方形のブロックを表示することによって、数値データを表します。棒の長さは、各データの値または数量に相当します。棒の長さを対比することにより、データ間の比較結果を明確なイメージとして頭の中に描くことができます。棒グラフは、数値データを表示してデータを対比させる場合に適しています。

グラフタイプは BAR (棒グラフ) です。BAR には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです。
- **y-axis** - メジャーです。
- **color** - 棒および凡例の色です (オプション)。

  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。複数のグラフがサポートされます。
- **size** - 棒の幅です (オプション)。

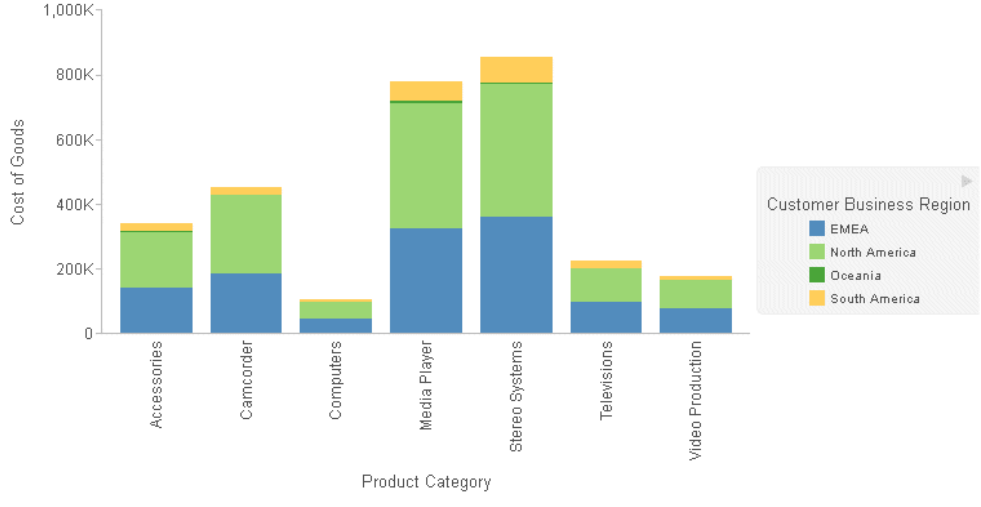
例　グラフ属性構文を使用した積み上げ棒グラフの例

次のリクエストは、積み上げ棒グラフを生成します。ソートフィールド (PRODUCT_CATEGORY) は x-axis、ソートフィールド (BUSINESS_REGION) は color のカテゴリに割り当てられ、メジャーは y-axis に割り当てられています。積み上げ棒グラフの作成についての詳細は、193 ページの 「 棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定 」を参照してください。

グラフタイプは BAR (棒グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=stacked,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用したボックスプロット

ボックスプロットは、データを四分位に分けて示します。ボックスプロットでは、各シリーズとグループに 5 つの値が必要です。特定のシリーズとグループボックスでは、1 つ目の値が最小値 (下側のハット)、2 つ目の値がボックス下端 (第 1 四分位)、3 つ目の値が中央値 (第 2 四分位)、4 つ目の値がボックス上端 (第 3 四分位)、5 つ目の値が最大値 (上側のハット) をそれぞれ定義します。ボックスプロットの塗りつぶし色および境界線は、シリーズの色および境界線で定義されます。

グラフタイプは BOXPLOT (ボックスプロット) です。BOXPLOT には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです。
- **min** - 最小値を表すメジャーです。
- **lower** - ボックス下端を表すメジャーです。
- **median** - 中央値を表すメジャーです。
- **upper** - ボックス上端を表すメジャーです。
- **max** - 最大値を表すメジャーです。
- **color** - ボックスおよび凡例の色に使用されるソートフィールドです (オプション)。

  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。

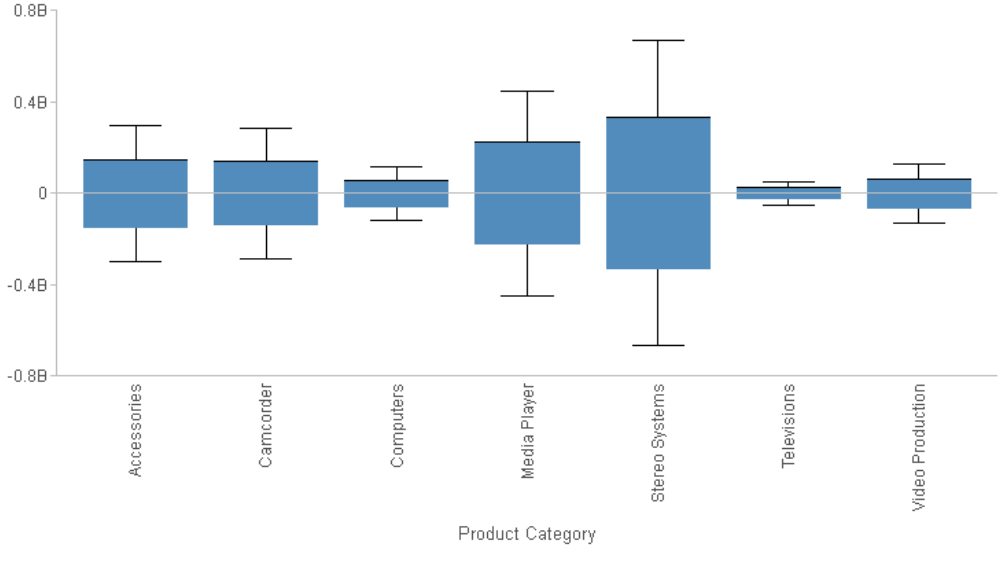
四分位値が自動的に計算される統計ボックスプロットについての詳細は、182 ページの 「 グラフ属性構文を使用した統計ボックスプロット 」 を参照してください。

例　　グラフ属性構文を使用したボックスプロットの例

次のリクエストは、ボックスプロットを生成します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=diff2, bucket=min,$
type=data, column=diff1, bucket=lower,$
type=data, column=c4, bucket=median,$
type=data, column=diff3, bucket=upper,$
type=data, column=diff4, bucket=max,$
*GRAPH_JS
"boxPlotProperties": {"drawHatAsBox": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用したバブルグラフ

バブルグラフは散布図を拡張したグラフで、メジャー値に比例して各マーカーのサイズが変化します。そのため、各バブルマーカーを描画するには、バブルグラフに 3 つの値 (X 軸値、Y 軸値、サイズ) が必要です。

グラフタイプは BUBBLE (バブルグラフ) です。BUBBLE には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです。ソートフィールドがメジャーの場合、X 軸上のバブルの切片を指定します。
- **y-axis** - Y 軸上のバブルの切片を表すメジャーです。
- **size** - バブルのサイズを表すメジャーです。
- **color** - バブルおよび凡例の色に使用されるフィールドです (オプション)。

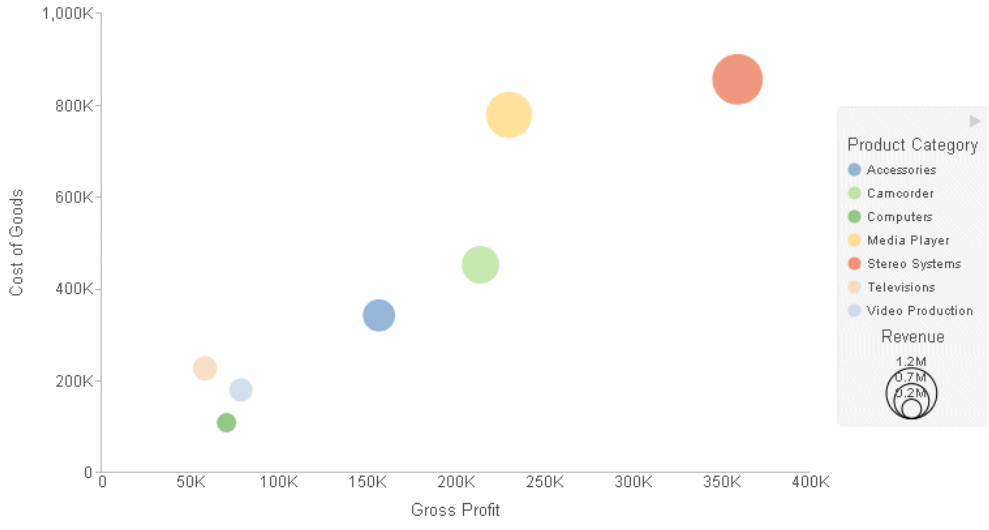
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。
- **detail** - データポイントをグラフに追加するために使用される下位ソートフィールドです (オプション)。

例　グラフ属性構文を使用したバブルグラフの例

次のリクエストは、バブルグラフを生成します。ソートフィールド (PRODUCT_CATEGORY) は color の属性カテゴリに割り当てられ、凡例を生成し、1 番目のメジャー (COGS_US) は y-axis、2 番目のメジャー (GROSS_PROFIT_US) は x-axis に割り当てられます。また、3 番目のメジャー (REVENUE_US) は size のカテゴリに割り当てられ、凡例領域のサイズインジケータを生成します。グラフタイプは BUBBLE (バブルグラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=x-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=size, $
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用したバブルマップ

バブルマップは、プロポーショナルシンボルマップグラフで、マーカー (バブル) が地理的なポイントやエリアに配置され、そのサイズはメジャー値に比例します。

グラフタイプは BUBBLEMAP (バブルマップ) です。BUBBLEMAP には次の属性カテゴリがあります。

- **location** - 地図上にシンボルが表示される地理位置情報を含むソートフィールドです。使用されるのは location または latitude および longitude のいずれか一方です。
- **latitude** - シンボルの緯度を含むソートフィールドまたはメジャーです。使用されるのは location または latitude および longitude のいずれか一方です。
- **longitude** - シンボルの経度を含むソートフィールドまたはメジャーです。使用されるのは location または latitude および longitude のいずれか一方です。
- **size** - シンボルのサイズを表すメジャーです。
- **color** - シンボルおよび凡例の色を表示するために使用するフィールドです (オプション)。

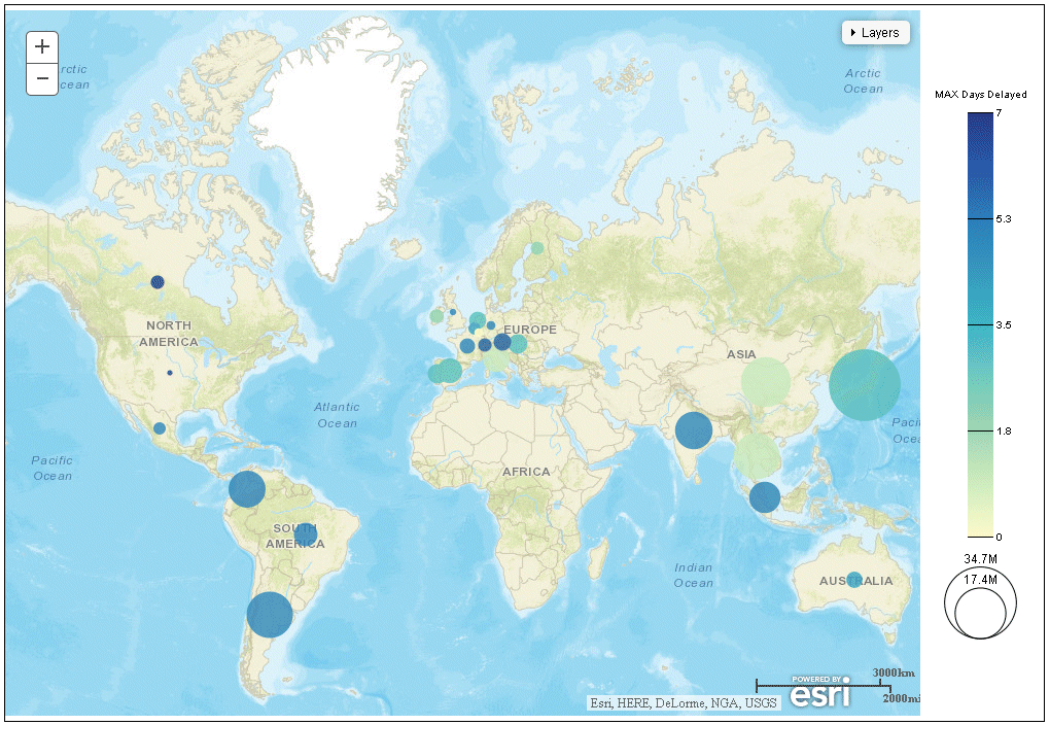
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。

例　グラフ属性構文に位置情報を使用したバブルマップの例

次のリクエストは、ベースレイヤとして ESRI マップを使用したプロポーショナルシンボルマップグラフを生成します。マーカーは、COUNTRY_NAME ソートフィールドを使用して配置されます。また、MIN.CITY_POPULATION メジャーによって色分けされ、MAX.DAYSDELAYED メジャーによってサイズが決められます。JSON プロパティ bubbleMarker:maxSize は、マーカーを 10% まで拡大します。スタイルシートのその他のプロパティは、マップタイプ、マップレイヤなどのマッププロパティを指定します。詳細は、871 ページの「マップサポート」を参照してください。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION MAX.DAYSDELAYED
BY COUNTRY_NAME
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
TYPE=DATA, COLUMN=MAX.DAYSDELAYED, BUCKET=color, $
*GRAPH_JS
"legend": {
    "visible": true
},
 "bubbleMarker": {"maxSize": "10%" },

"extensions": {
    "com.esri.map": {
        "overlayLayers":             [
                {
                    "title": "Population",
                    "layerType": "bubble",
                    "geometrySourceType": "esri",
                    "geometryLocateField": ["Country"],
                    "geometryDataField": "name",
                    "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/
arcgis/rest/services/World_Countries_(Generalized)/FeatureServer/0"
                }
            ],
        "baseLayer": {
            "basemap": "streets"
        }
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



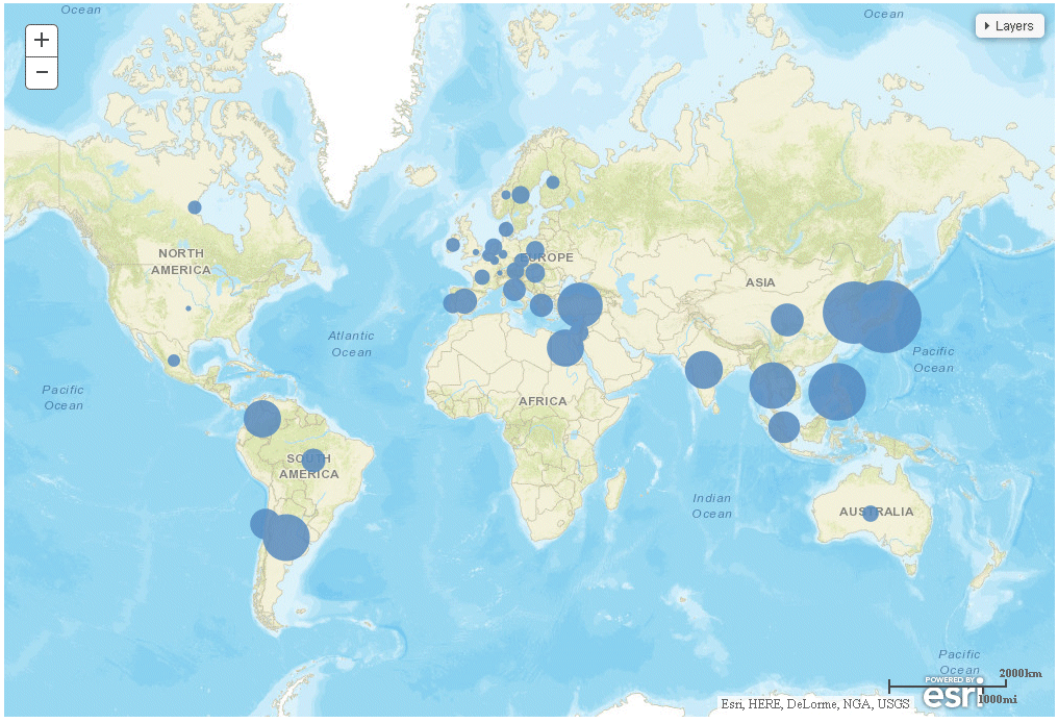
地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

例　グラフ属性構文に緯度と経度を使用したバブルマップの例

次のリクエストは、ベースレイヤとして ESRI マップを使用したプロポーショナルシンボルマップグラフを生成します。マーカーは、COUNTRY_LATITUDE および COUNTRY_LONGITUDE ソートフィールドを使用して配置され、MIN.CITY_POPULATION メジャーによってサイズが決められます。JSON プロパティ bubbleMarker:maxSize は、マーカーを 10% まで拡大します。スタイルシートのその他のプロパティは、マップタイプ、マップレイヤなどのマッププロパティを指定します。詳細は、871 ページの 「 マップサポート 」 を参照してください。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY COUNTRY_LATITUDE
BY COUNTRY_LONGITUDE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=COUNTRY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
  "bubbleMarker":{"maxSize":"10%"},

          "legend": {
              "visible": false
          },
          "extensions": {
              "com.esri.map": {
                  "overlayLayers":            [
                          {
                              "title": "Population",
                              "layerType": "bubble",
                              "geometryXY": {
                                  "y": "lat",
                                  "x": "lng"
                              },
                              "geometrySourceType": "seriesdata"
                          }
                      ],
                  "baseLayer": {
                      "basemap": "streets"
                  }
              }
          }

          *END
          ENDSTYLE
          END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

グラフ属性構文を使用したコロプレスマップ

コロプレスマップは、地図上の場所に表示された多角形の領域が、メジャーまたはディメンションに基づいて色分けされたマップグラフです。

グラフタイプは CHOROPLETH (コロプレスマップ) です。CHOROPLETH には次の属性カテゴリがあります。

- **location** - 地図上にシンボルが表示される地理位置情報を含むソートフィールドです。
- **color** - 多角形の色を指定します。

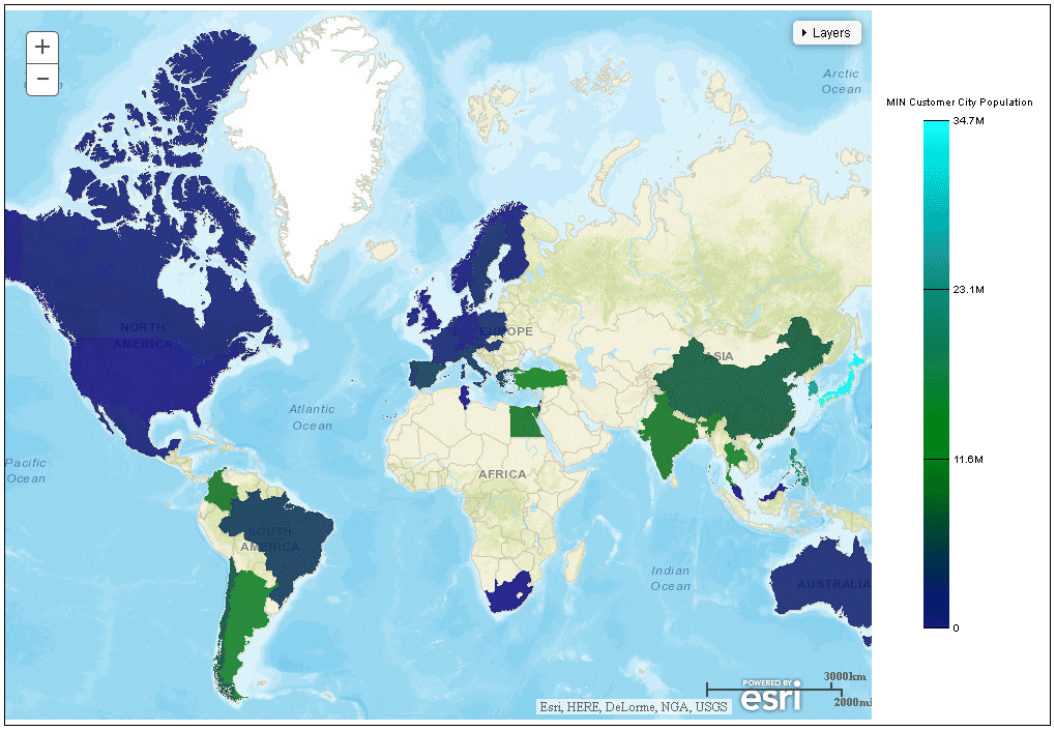
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。

例　グラフ属性構文を使用したコロプレスマップの例

次のリクエストは、ベースレイヤとして ESRI マップを使用したコロプレスマップを生成します。色分けされた多角形は、COUNTRY_NAME ソートフィールドを表示し、MIN.CITY_POPULATION メジャーによって色分けされています。JSON プロパティ colorScale は、多角形のカラースケールを生成します。スタイルシートのその他のプロパティは、マップタイプ、マップレイヤなどのマッププロパティを指定します。詳細は、871 ページの「マップサポート」を参照してください。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY COUNTRY_NAME
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=color, $
*GRAPH_JS
legend: {
    "visible": true
},
 "colorScale": {"colors":["navy", "green", "teal", "cyan"]},

"extensions": {
    "com.esri.map": {
        "overlayLayers":            [
                {
                    "title": "WF_RETAIL_LITE Countries",
                    "layerType": "choropleth",
                    "geometrySourceType": "esri",
                    "geometryLocateField": ["Country"],
                    "geometryDataField": "name",
                    "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/
arcgis/rest/services/World_Countries_(Generalized)/FeatureServer/0"
                }
            ],
        "baseLayer": {
            "basemap": "streets"
        }
    }
}

*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

グラフ属性構文を使用したデータグリッド

データグリッドは、行と列の表形式でデータを表示します。データグリッドは、WebFOCUS TABLE FILE コマンドで生成される表形式レポートと異なり、PCHOLD FORMAT JSCHART を使用した GRAPH FILE リクエストで生成されます。

データグリッドのプロパティを使用して、グリッドのフォーマットを設定したり、グリッド内のメジャーの列合計を表示したりすることができます。

グラフタイプは DATAGRID (データグリッド) です。DATAGRID には次の属性カテゴリがあります。

- **measure** - グリッドのセルに表示される値のメジャーです。

- **row** - グリッドの行を表すメジャーまたはディメンションフィールドです。
- **column** - グリッドの列を表すメジャーまたはディメンションフィールドです。

例 データグリッドの生成

次のリクエストは、データグリッドを生成し、列合計を表示します。行は、製品区分 (PRODUCT_CATEGORY) と製品区分 (詳細) (PRODUCT_SUBCATEG) を示します。列は、当該年度の第 1 四半期と第 2 四半期、および北米と南米のビジネス地方区分を示します。メジャーは、売上原価 (COGS_US) と粗利益 (GROSS_PROFIT_US) です。デフォルト設定では、グリッド全体が描画領域に入りきらない場合にスクロールバーが追加されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY TIME_QTR
BY BUSINESS_REGION
WHERE (BUSINESS_REGION NE 'Oceania' OR 'EMEA') AND (TIME_QTR EQ 1 OR 2)
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=time_qtr,          bucket=column, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,           bucket=measure, $
type=data, column=gross_profit_us,   bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"columnTotals": {"visible": true}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Sale Quarter		1				2			
Customer Business Region		North America		South America		North America		South America	
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Prof
Accessories	Charger	$1,445.00	$1,441.10	$170.00	$157.98	$945.00	$901.25	$232.00	$242.9
	Headphones	$23,243.00	$11,691.55	$2,659.00	$1,551.87	$23,680.00	$12,076.97	$5,266.00	$1,987.5
	Universal Remote Controls	$20,671.00	$7,428.05	$5,034.00	$1,577.96	$19,995.00	$8,044.73	$2,559.00	$695.8
Camcorder	Handheld	$12,172.00	$12,772.12	$3,170.00	$3,172.94	$12,692.00	$13,031.36	$1,644.00	$1,411.6
	Professional	$23,570.00	$7,811.15			$31,030.00	$1,850.75		
	Standard	$31,199.00	$13,095.73	$7,277.00	$3,065.45	$36,694.00	$13,544.14	$3,113.00	$1,187.1
Computers	Smartphone	$15,052.00	$8,932.73	$3,720.00	$2,514.65	$15,568.00	$10,256.59	$1,321.00	$679.8
Media Player	Blu Ray	$113,791.00	$34,313.74	$25,734.00	$7,551.09	$107,000.00	$29,846.63	$16,449.00	$4,340.9
	DVD Players	$7,743.00	$4,663.02	$2,184.00	$1,012.87				
	Streaming	$1,748.00	$1,187.62	$184.00	$135.96	$1,380.00	$983.70	$46.00	$33.9
Stereo Systems	Home Theater Systems	$36,291.00	$17,731.46	$7,527.00	$3,509.50	$37,369.00	$18,277.90	$4,609.00	$2,183.6
	Receivers	$20,082.00	$8,774.40	$3,992.00	$1,425.23	$23,382.00	$8,922.36	$6,542.00	$2,709.4
	Speaker Kits	$54,032.00	$17,453.00	$12,007.00	$4,001.42	$45,503.00	$14,251.98	$11,014.00	$3,570.9
	iPod Docking Station	$17,786.00	$10,859.56	$2,888.00	$1,373.32	$17,902.00	$9,982.36	$2,907.00	$1,770.5
Televisions	Flat Panel TV	$29,889.00	$8,931.25	$3,829.00	$883.92	$33,167.00	$9,311.83	$15,847.00	$4,310.9
Video Production	Video Editing	$27,278.00	$12,561.31	$3,026.00	$1,550.34	$29,025.00	$11,446.98	$4,592.00	$1,760.2
TOTAL		$435,992.00	$179,647.79	$83,401.00	$33,484.50	$435,332.00	$162,729.53	$76,141.00	$26,885.8

グラフ属性構文を使用したじょうごグラフ

じょうごグラフは基本的には円グラフですが、一度に表示するデータグループは 1 つです。グループに属する各シリーズは、じょうご内に積み重ねられ、最初のシリーズが最上部に、最後のシリーズが最下部に表示されます。じょうごグラフでは、表示フィールドはグループとして機能し、ソートフィールドはシリーズとして機能します。

グラフタイプは FUNNEL (じょうごグラフ) です。FUNNEL には次の属性カテゴリがあります。

- **color** - ソートフィールドです。
- **measure** - じょうご項目の生成のために使用されるメジャーです。

例　グラフ属性構文を使用したじょうごグラフの例

次のリクエストは、じょうごグラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=measure, $
ENDSTYLE
END
```

出力結果は次のとおりです。

グラフ属性構文を使用したメータグラフ

メータグラフでは、一般に円形で表示される軸で単一値を識別します。メータグラフはダッシュボードでよく使用され、進捗率や数量を表示します。

グラフタイプは GAUGE (メータグラフ) です。GAUGE には次の属性カテゴリがあります。

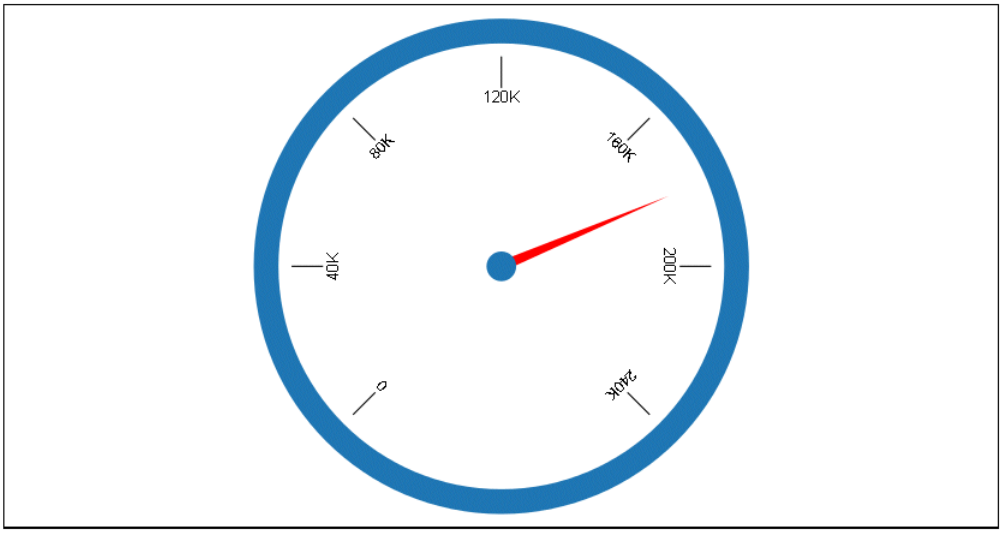
- **measure** - メータに表示される値を含むフィールドです。

例　グラフ属性構文を使用したメータグラフの例

次のリクエストは、メータグラフを生成します。グラフタイプは GAUGE (メータグラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
    "groupLabel": {"visible": false},
    "totalLabel": {"visible": false}
    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用したヒートマップ

ヒートマップは、データ値に応じて長方形をさまざまな色で表示し、これらの長方形を行と列のマトリックスで表したグラフです。ヒートマップでは、棒グラフ、折れ線グラフ、面グラフで使用されるデータと同種類のデータが使用されます。

グラフタイプは HEATMAP (ヒートマップ) です。HEATMAP には次の属性カテゴリがあります。

- **y-axis** - 1 番目のソートフィールドです。
- **x-axis** - 2 番目のソートフィールドです。
- **color** - メジャーです。

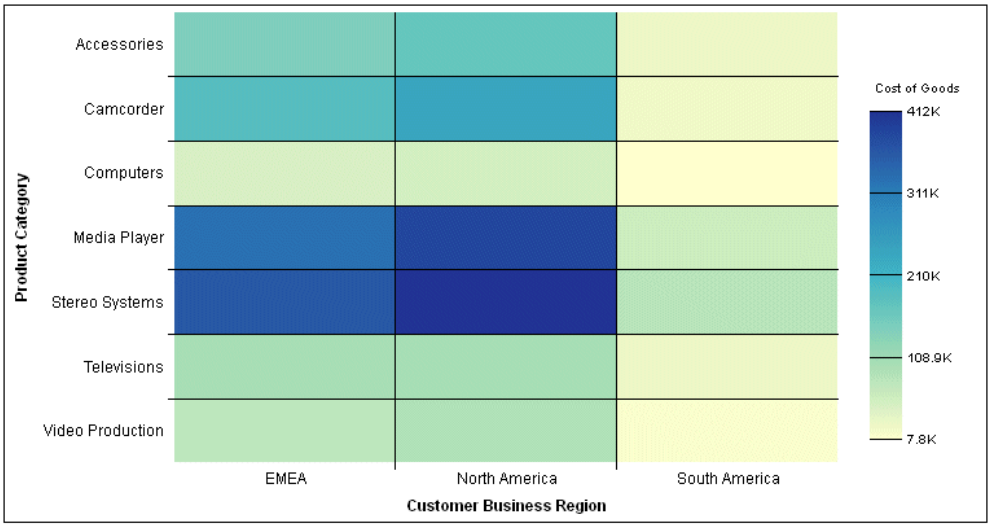
  長方形の色は、このメジャーの値に基づきます。この場合、色はグラデーションとして表示されます。

例　グラフ属性構文を使用したヒートマップの例

次のリクエストは、ヒートマップを生成します。長方形はメジャー COGS_US (color カテゴリ) によって色分けされ、行は 1 番目の BY フィールド (y-axis カテゴリ) を表し、列は 2 番目の BY フィールド (x-axis カテゴリ) を表します。グラフタイプは HEATMAP (ヒートマップ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH SET LOOKGRAPH HEATMAP
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=y-axis, $
type=data, column=business_region, bucket=x-axis, $
type=data, column=cogs_us, bucket=color, $
END
```

出力結果は次のとおりです。



グラフ属性構文を使用した折れ線グラフ

折れ線グラフは、時間の経過とともに変化する数値データの動向や傾向を強調する際に役立ちます。これは、過去の状態を観察したり補間したりすることによって、特定の点の変動を追跡できるためです。高低のレベル、遅速の動き、安定に向けての傾向は、折れ線グラフが適するトレンドタイプです。

また、折れ線グラフに複数の目盛りを使用して、異なる期間においての同一データ値または複数データ値を比較することもできます。グラフに表示できるスケールの数は、選択するグラフのタイプによって異なります。

グラフタイプは LINE (折れ線グラフ) です。LINE には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです。通常は YEAR など時間に基づくディメンションです。
- **y-axis** - メジャーです。
- **color** - 線、マーカー、および凡例の色です (オプション)。

  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。複数のグラフがサポートされます。
- **size** - マーカーまたは線の幅を表すメジャーです (オプション)。

  マーカーが表示される場合、このメジャーはマーカーのサイズを変えるために使用されます。マーカーが表示されない場合、このメジャーは線の幅を変えるために使用されます。

例　折れ線グラフの例

次のリクエストは、折れ線グラフを作成します。ソートフィールド TIME_YEARMTH が x-axis カテゴリに割り当てられています。メジャー COGS_US が y-axis カテゴリに割り当てられています。メジャー DISCOUNT_US が color カテゴリに割り当てられ、線、マーカー、凡例の色に使用されます。色フィールドがメジャーであるため、色はグラデーションとして表示されます。グラフタイプは LINE (折れ線グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY WF_RETAIL_TIME_SALES.TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=WF_RETAIL_TIME_SALES.TIME_YEARMTH, BUCKET=x-axis,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis,$
TYPE=DATA, COLUMN=DISCOUNT_US, BUCKET=color,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。

グラフ属性構文を使用したマトリックスマーカーグラフ

マトリックスマーカーグラフは、マトリックスの各セル内にマーカーを表示するマトリックスグラフです。

グラフタイプは MARKER (マトリックスマーカーグラフ) です。MARKER には次の属性カテゴリがあります。

- **row** - マトリックスの行を表すフィールドです。
- **column** - マトリックスの列を表すフィールドです。
- **size** - マーカーのサイズを表すメジャーです。
- **color** - マーカーの色を表すメジャーです。

  この場合、色はグラデーションとして表示されます。

例　グラフ属性構文を使用したマトリックスマーカーグラフの例

次のリクエストは、マトリックスマーカーグラフを生成します。行は、製品区分 (PRODUCT_CATEGORY) を示します。列は、ビジネス地方区分 (BUSINESS_REGION) を示します。各セルのマーカーは、メジャー GROSS_PROFIT_US に基づいてサイズが決定され、メジャー COGS_US に基づいて色が決定されます。マーカーと凡例の色は、グラデーションとして表示されます。

JSON プロパティでマーカーの形状が正方形に変更されています。グラフタイプは MARKER (マトリックスマーカーグラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
WHERE PRODUCT_CATEGORY LE 'D'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH MARKER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=size, $
*GRAPH_JS
"series": [{"series": 0, "marker": {"shape": "square"}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

グラフ属性構文を使用したマリメッコチャート

マリメッコチャート (別名、メッコチャート) は、パーセント棒グラフに類似していますが、積み上げ棒の全体値に基づいて棒ライザの幅が決定されるという点が異なります。ライザの幅は降順で自動的にソートされます。デフォルト設定では、ツールヒントにグループ内のメジャーの値とパーセンテージが表示されます。

グラフタイプは MEKKO (マリメッコチャート) です。MEKKO には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです。
- **y-axis** - メジャーです。
- **color** - 棒および凡例の色です (オプション)。

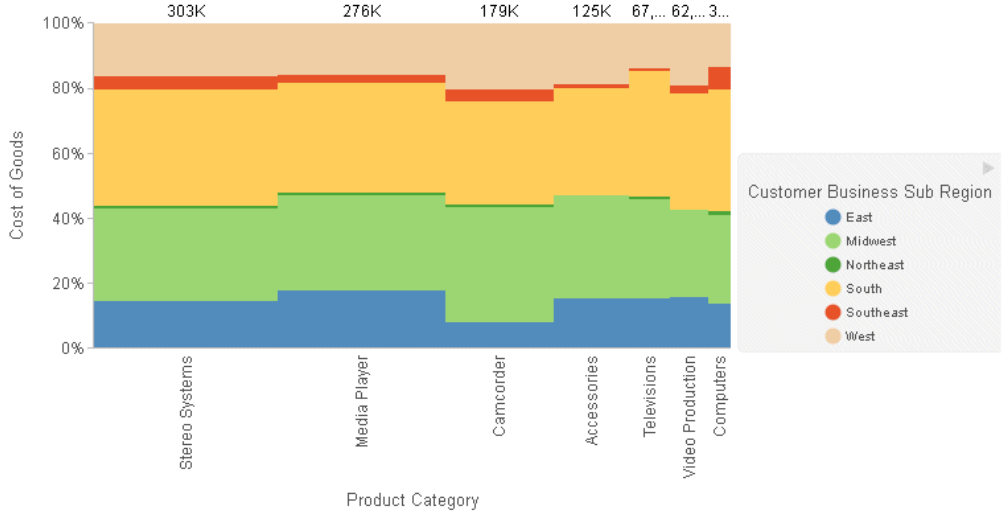
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。複数のグラフがサポートされます。

例　グラフ属性構文を使用したマリメッコチャートの例

次のリクエストは、マリメッコチャートを生成します。x-axis カテゴリには 1 つのメジャーと 1 つのソートフィールド、color カテゴリには 1 つのソートフィールドが割り当てられています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_SUB_REGION
BY PRODUCT_CATEGORY
WHERE COUNTRY_NAME EQ 'United States'
ON TABLE PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH MEKKO
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=BUSINESS_SUB_REGION, bucket=color,$
type=data, column=product_category, bucket=x-axis,$
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用した円グラフ

円グラフは、特定のデータが全体の中でどの程度の割合を占めるかを強調します。全体に対する割合は、円項目のサイズで表されます。円グラフは、データが少数のグループに分類される場合に適しています。グループ数が多すぎると、円が小さいセグメントに分割されるため、判読が困難になります。視覚的なコントラストを付けるには、個々のセグメントに色またはテクスチャを使用します。

穴のサイズがゼロの場合は円グラフが生成されます。穴のサイズがゼロより大きい場合は、リンググラフが生成されます。

グラフタイプは PIE (円グラフ) です。PIE には次の属性カテゴリがあります。

- **color** - ソートフィールドです。
- **measure** - 円項目の生成のために使用されるメジャーです。
- **size** - マトリックス配列で生成された場合に、円グラフのサイズを決めるために使用されるメジャーです (オプション)。

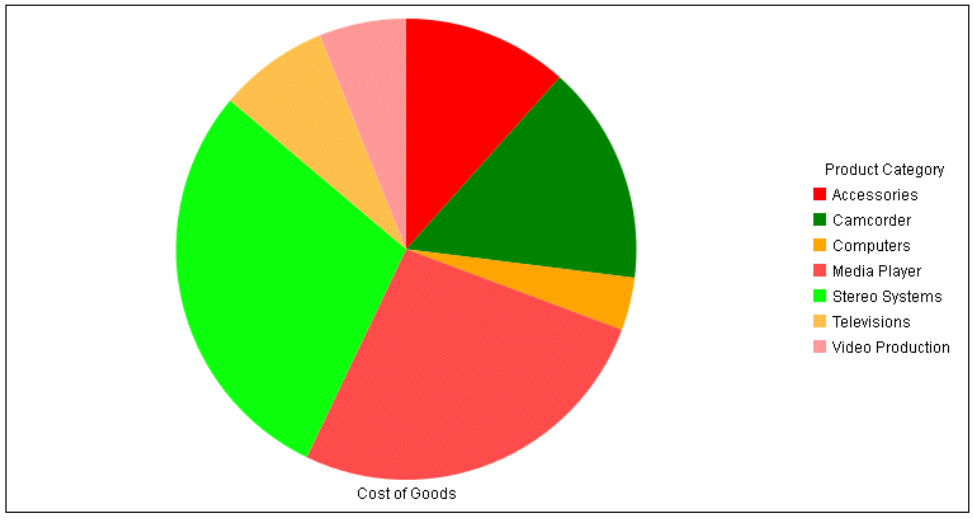
注意：measure カテゴリにメジャーを追加し、color カテゴリに BY フィールドを追加して作成した円グラフは、単一の円グラフになり、グラフ下部にメジャー名が表示されます。

例　グラフ属性構文を使用した円グラフの例

円グラフの基本的なコンポーネントは、メジャーとソートフィールドです。ソートフィールドは、項目および凡例の色を生成します。グラフタイプは PIE (円グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=measure, $
ENDSTYLE
END
```

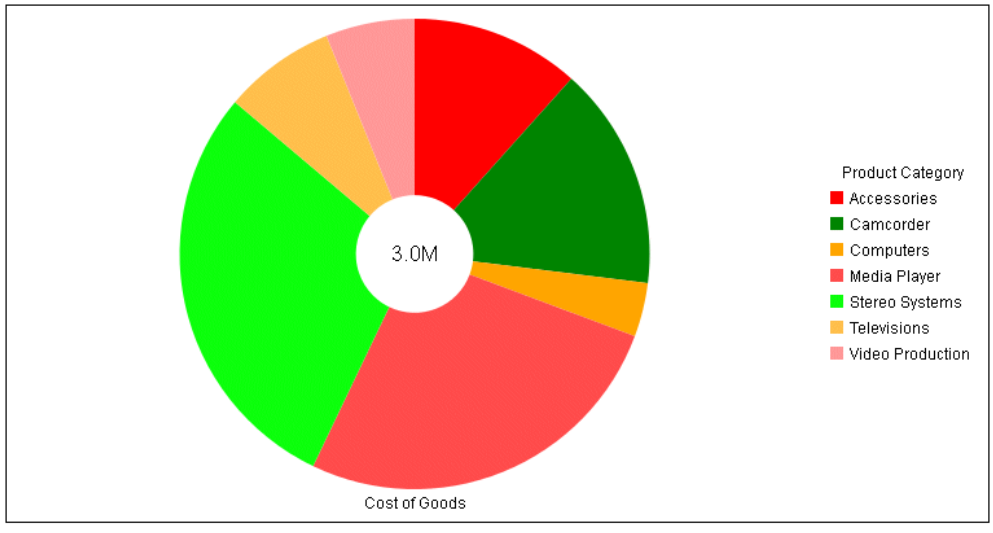
出力結果は次のとおりです。



穴開きの円グラフを作成するには、JSON holeSize プロパティを追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=measure, $
*GRAPH_JS
"pieProperties": {"holeSize": "25%"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。デフォルト設定では、穴の中に合計ラベルが表示されます。



グラフ属性構文を使用した散布図

散布図は、2 つの数値メジャーの関係を示します。グラフ属性構文では、メジャーとともにディメンションもサポートされます。散布図は、特定のポイント周辺のデータ値の密集度を視覚的に表示したり、データのパターンを明示したりする場合に使用します。

グラフタイプは SCATTER (散布図) です。SCATTER には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです (メジャーまたはディメンションのいずれか)。
- **y-axis** - マーカーの Y 軸上の切片を定義するメジャーまたはディメンションです。
- **size** - マーカーのサイズを表すメジャーです (オプション)。
- **color** - マーカーの色を表すフィールドです (オプション)。

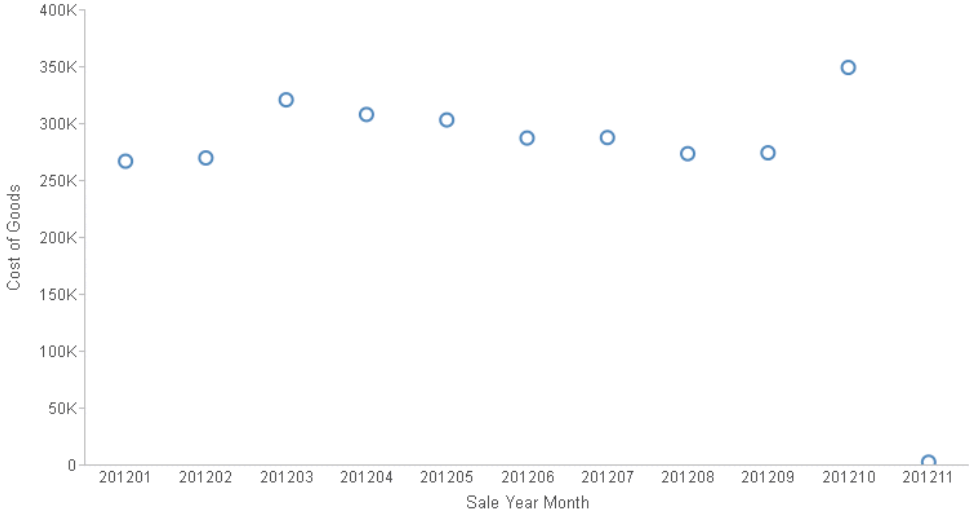
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。
- **detail** - グラフ上に追加のデータポイントを生成するために使用される下位ソートフィールドです (オプション)。

例　グラフ属性構文を使用した散布図の例

次のリクエストは、散布図を作成します。ソートフィールド TIME_YEARMTH が x-axis カテゴリに割り当てられています。メジャー COGS_US が y-axis カテゴリに割り当てられています。グラフタイプは SCATTER (散布図) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY WF_RETAIL_TIME_SALES.TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=WF_RETAIL_TIME_SALES.TIME_YEARMTH, BUCKET=x-axis,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用した統計ボックスプロット

統計ボックスプロットでは、四分位と異常値が 1 つ以上のメジャーと 1 つのディメンションから自動的に計算され、より簡単にボックスプロットが生成できます。これに対して、従来のボックスプロット (BOXPLOT) では、ユーザが事前に四分位を計算し、5 つの値として提供する必要があります。また、異常値は表示することができません。

統計ボックスプロットでは、異常値が個々の点として描画されます。

ボックスプロットの塗りつぶし色および境界線は、シリーズの色および境界線で定義されます。シリーズマーカーのプロパティが、異常値を表す点の外観を制御します。

グラフタイプは STAT-BOXPLOT です。STAT-BOXPLOT には次の属性カテゴリがあります。

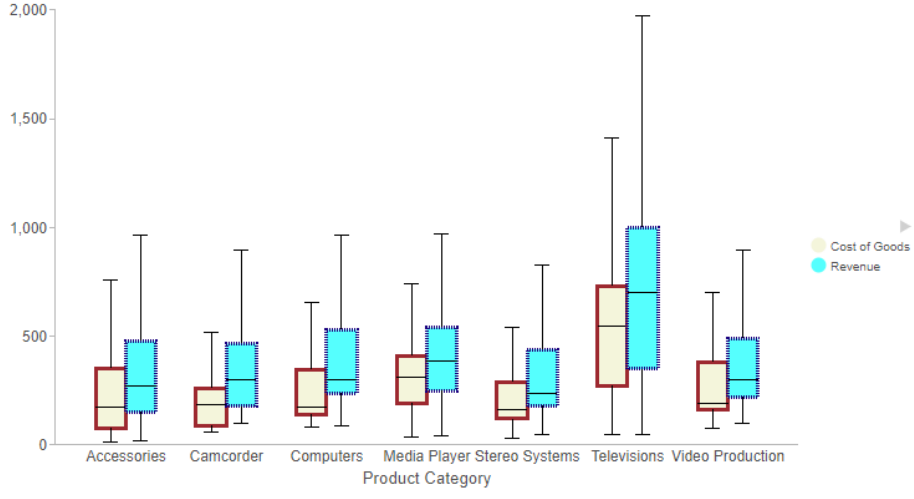
- **x-axis** - ソートフィールドです (ディメンション)。
- **y-axis** - シリーズフィールドです (メジャー)。

例　統計ボックスプロットの例

次のリクエストは、2 つのメジャーを持つ統計的なボックスプロットを生成します。シリーズのプロパティは、ボックスの塗りつぶし色および境界線を定義します。

```
GRAPH FILE wf_retail_lite
PRINT COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STAT-BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=product_category, bucket=x-axis, $
type=data, column=cogs_us, bucket=y-axis, $
type=data, column=revenue_us, bucket=y-axis, $
*GRAPH_JS
"axisList":{
"y1": {"max": 2000, "min": 0}},
"series":[
{"series":0, "color": "beige", "border": {"color": "brown", "width": 3}},
{"series":1, "color": "cyan", "border": {"color": "navy", "width": 3,
"dash": "1 1"}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

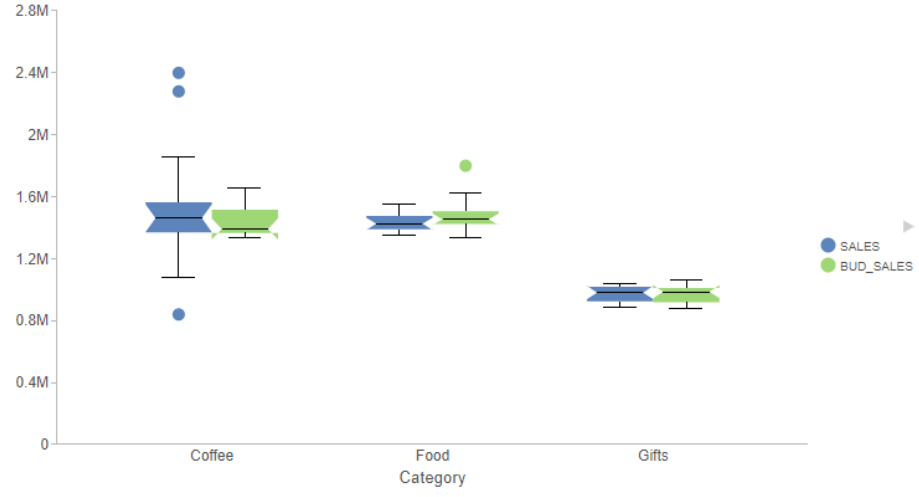


例　異常値を含む統計ボックスプロットの生成

次のリクエストは、各ボックスの統計的測定値から外れた点を生成します。

```
DEFINE FILE GGSALES
SALES/D12 = IF ((CATEGORY EQ 'Coffee') AND (STCD EQ 'R1019')) THEN 19000
  ELSE IF ((CATEGORY EQ 'Coffee') AND (STCD EQ 'R1020')) THEN 20000
  ELSE IF ((CATEGORY EQ 'Coffee') AND (STCD EQ 'R1040')) THEN 7000
  ELSE DOLLARS;
BUD_SALES/I12 = IF ((CATEGORY EQ 'Food') AND (STCD EQ 'R1019')) THEN 15000
ELSE BUDDOLLARS;
END

GRAPH FILE GGSALES
SUM SALES
BUD_SALES
BY CATEGORY
BY STCD
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STAT-BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=n1, bucket=x-axis, $
type=data, column=n2, bucket=detail, $
type=data, column=n3, bucket=y-axis, $
type=data, column=n4, bucket=y-axis, $
*GRAPH_JS
"series": [{"series": 0, "marker": {"visible": true}},
           {"series": 1, "marker": {"visible": true}}],
"boxPlotProperties": {
  "notch": {"width": "70%"}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。異常値は、ボックスの上または下に表示される個々の点です。



グラフ属性構文を使用したストリームグラフ

ストリームグラフは、積み上げ面グラフを簡素化したグラフです。ストリームグラフでは X 軸が無効化される場合が多く、罫線やフレームが省略されます。ストリームグラフでは、データテキストラベルは使用されません。ストリームグラフの描画に必要なデータのフォーマットは、面グラフの描画に必要なフォーマットと同一です。ただし、一般にストリームグラフでは多くのシリーズ (10 以上) が使用され、各シリーズに多くのデータポイント (100 以上) が存在します。通常、ストリームグラフには 20 個のシリーズと、シリーズごとに 400 個のデータポイントが含まれます。

グラフタイプは STREAM (ストリームグラフ) です。STREAM には次の属性カテゴリがあります。

- **x-axis** - ソートフィールドです。
- **y-axis** - メジャーです。
- **color** - エリアおよび凡例の色です (オプション)。

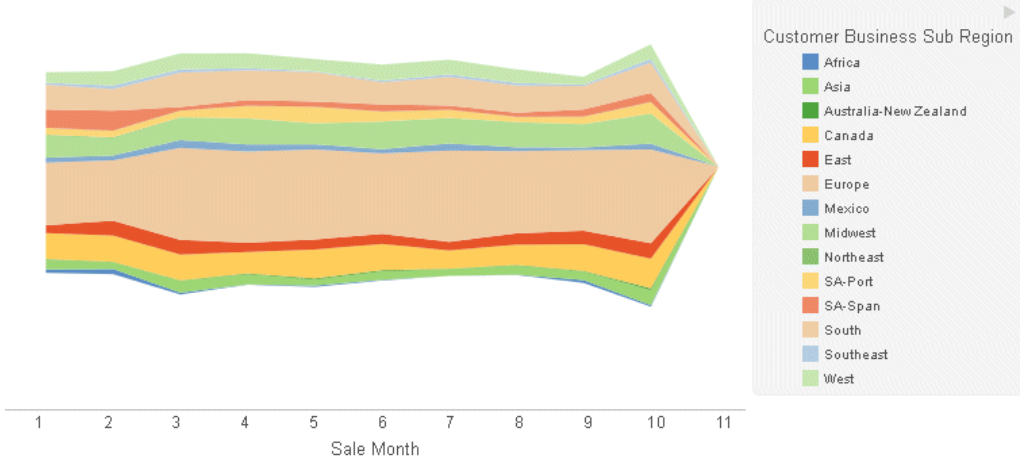
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。複数のグラフがサポートされます。

例　ディメンションで色分けされたストリームグラフの例

次のリクエストは、ストリームグラフを生成します。エリアは BUSINESS_SUB_REGION ディメンションで色分けされています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_SUB_REGION
BY TIME_MTH
ON TABLE PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STREAM
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=time_mth, bucket=x-axis,$
type=data, column=business_sub_region, bucket=color,$
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用したタグクラウド

タグクラウドは、グループのラベルをグラフ化したものです。データ値に比例して、各ラベルのサイズが変化します。タグクラウドはソーシャルメディアで使用され、語句の発生頻度を計測して感情を定量化します。

グラフタイプは TAGCLOUD (タグクラウド) です。TAGCLOUD には次の属性カテゴリがあります。

- **detail** - 表示されるタグを含むソートフィールドです。
- **size** - タグのサイズを制御するメジャーです。
- **color** - タグの色を制御するメジャーです。タグクラウドは複数のシリーズや凡例をサポートしないため、このカテゴリはソートフィールドに割り当てることができません。

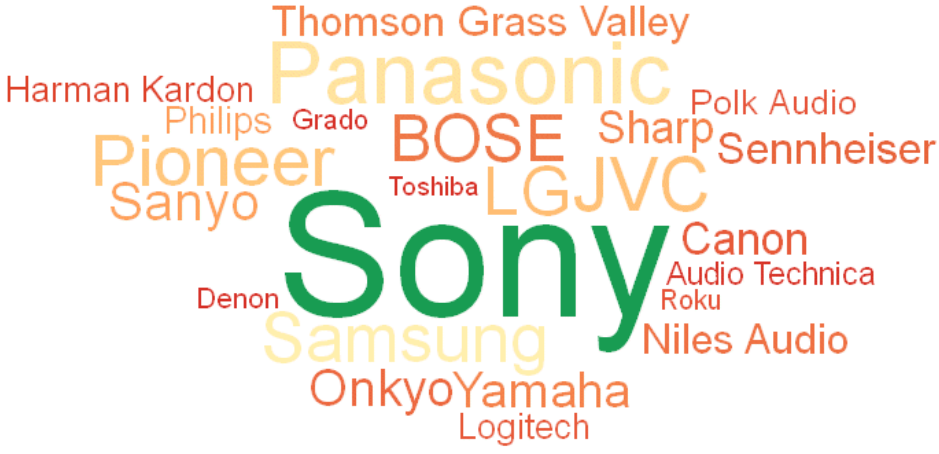
タグクラウドでは、マトリックス配列を使用することができません。

例　グラフ属性構文を使用したタグクラウドの例

次のリクエストは、タグクラウドを生成します。この場合、タグはブランド名 (BRAND)、タグの色は販売数量 (QUANTITY_SOLD) で制御され、タグのサイズは粗利益 (GROSS_PROFIT_US) で制御されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD  GROSS_PROFIT_US
BY BRAND
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TAGCLOUD
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=brand, bucket=detail,$
type=data, column=quantity_sold, bucket=color,$
type=data, column=gross_profit_us, bucket=size,$
*GRAPH_JS
"tagcloudProperties": {"engine": "new"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ属性構文を使用したツリーマップ

ツリーマップには、階層データが、ネストされた長方形の集合体として表示されます。

オブジェクトは、任意の深さまでネストすることができます。

グラフタイプは TREEMAP (ツリーマップ) です。TREEMAP には次の属性カテゴリがあります。

- **detail** - 階層を定義するソートフィールドです。
- **size** - 長方形のサイズを決定するために使用されるメジャーです (オプション)。
- **color** - エリアの色を表すフィールドです (オプション)。

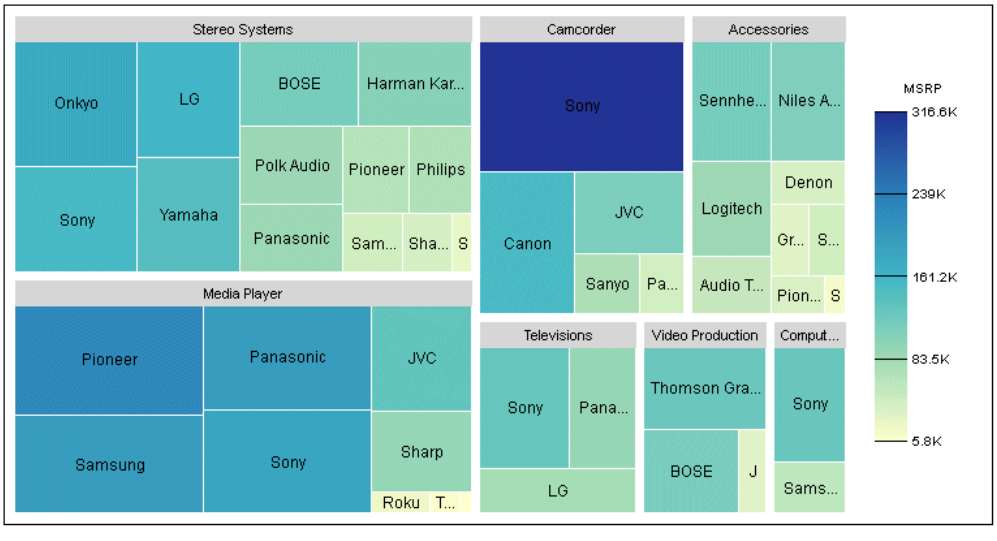
  フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。

例　メジャーに基づいて色分けされたツリーマップの例

次のサンプルリクエストは、ツリーマップを生成します。メジャー MSRP_US が色を決定し、メジャー COGS_US がサイズを決定しています。色フィールドがメジャーであるため、長方形および凡例の色はグラデーションとして表示されます。グラフタイプは TREEMAP (ツリーマップ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
BY BRAND
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=report, column=COGS_US,    bucket=size, $
type=report, column=MSRP_US,    bucket=color, $
type=report, column=PRODUCT_CATEGORY, bucket=detail, $
type=report, column=BRAND,    bucket=detail, $
END
```

出力結果は次のとおりです。

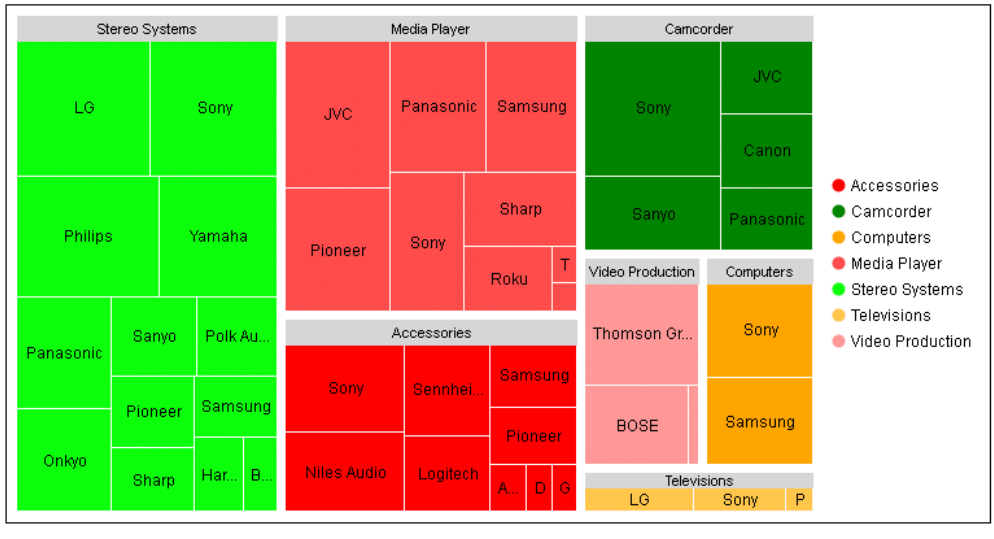


例 ディメンションに基づいて色分けされたツリーマップの例

次のサンプルリクエストは、ツリーマップを生成します。この場合、BY フィールドが色を決定しています。色フィールドがディメンションであるため、長方形および凡例は不連続の色で生成されます。2 番目の BY フィールド (detail 属性カテゴリ) が、色分けされたボックスの下位区分に相当します。リクエスト内で使用される演算接頭語は、フィールドをカテゴリに割り当てる際にも使用されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM CNT.COGS_US
BY PRODUCT_CATEGORY
BY BRAND
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=report, column=product_category, bucket=color, $
type=report, column=brand, bucket=detail, $
type=report, column=cnt.cogs_us, bucket=size, $
END
```

出力結果は次のとおりです。



グラフ方向の指定

グラフ方向を指定する場合は、WebFOCUS スタイルシートの TYPE=REPORT 宣言に CHART-ORIENTATION 属性を含めます。デフォルトでは、グラフの向きは縦方向に設定されています。

グラフ方向は、棒グラフ、折れ線グラフ、面グラフ、マリメッコチャート、ボックスプロットでサポートされています。

構文　グラフ方向の指定

```
TYPE=REPORT, CHART-ORIENTATION=orientation,$
```

説明

`orientation`
グラフ方向を指定します。有効な値には、次のものがあります。

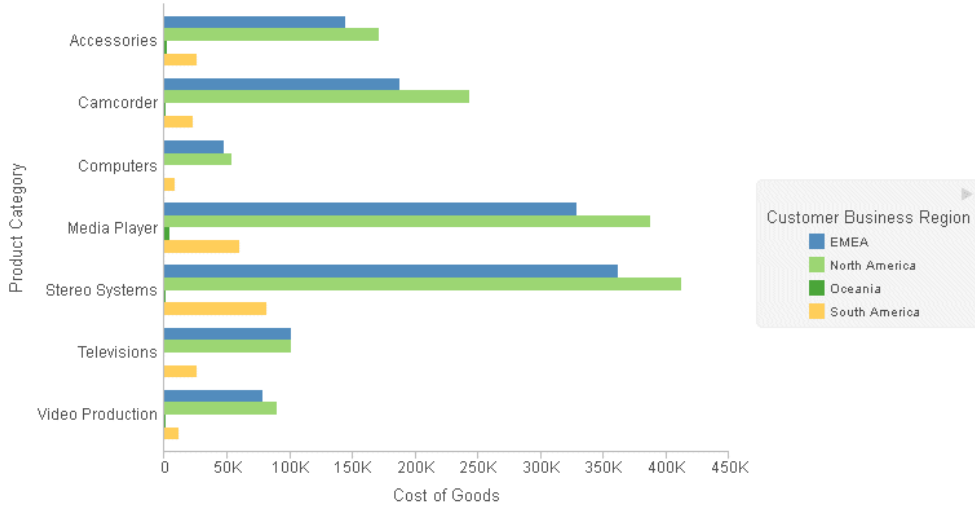
- **vertical** - X 軸をグラフの下側に、Y 軸をグラフの左側に描画します。これがデフォルト値です。
- **horizontal** - X 軸をグラフの左側に、Y 軸をグラフの下側に描画します。

例　グラフ方向の指定

次のリクエストは、横方向の棒グラフを生成します。グラフタイプは BAR (棒グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-ORIENTATION=horizontal,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定

棒グラフ、折れ線グラフ、面グラフのレイアウト (サブタイプ) を指定する場合は、WebFOCUS スタイルシートの TYPE=REPORT 宣言に CHART-SERIES-LAYOUT 属性を含めます。レイアウトとは、ライザの相対的位置関係のことです。

- 棒グラフは、並列、積み上げ、絶対、パーセントのレイアウトをサポートします。
- 面グラフは、絶対、積み上げ、パーセントのレイアウトをサポートします。
- 折れ線グラフは、絶対、積み上げ、パーセントのレイアウトをサポートします。

デフォルト設定のグラフレイアウトは、棒グラフでは並列、面グラフおよび折れ線グラフでは絶対です。

構文　　グラフ方向およびレイアウトの指定

```
TYPE=REPORT, CHART-SERIES-LAYOUT=layout,$
```

説明

layout
棒グラフ、折れ線グラフ、面グラフのライザの配置を定義します。有効な値には、次のものがあります。

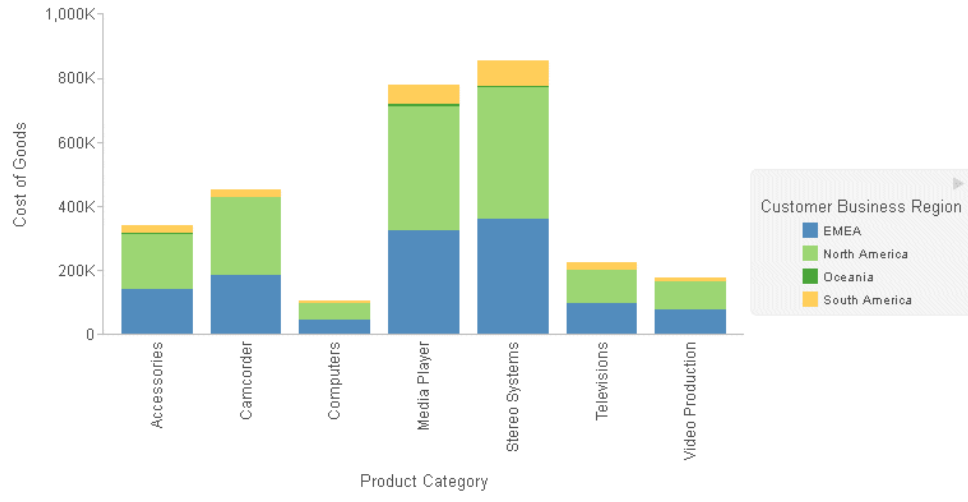
- **side-by-side** - ライザを別のライザの横に並べて配置 (縦棒グラフの場合)、またはライザを別のライザの上に並べて配置 (横棒グラフの場合) します。デフォルト値は side-by-side です。並列レイアウトは、棒グラフのみでサポートされます。
- **stacked** - (棒グラフ、折れ線グラフ、面グラフの場合) ライザを別のライザの上に積み重ねます。各ライザの長さは、データ値を表します。積み上げレイアウトは、棒グラフ、折れ線グラフ、面グラフでサポートされます。
- **absolute** - 縦グラフの場合、各ライザを同一の X 軸値の位置に描画します。横グラフの場合、各ライザを同一の Y 軸値の位置に描画します。各ライザに異なる幅を設定すると、グラフ上でライザを区別することができます。これがデフォルト設定のグラフレイアウトです。絶対レイアウトは、折れ線グラフおよび面グラフでサポートされます。
- **percent** - ライザを別のライザの上に積み重ねます。各ライザの長さは、そのライザの合計に対するパーセント値を表し、積み上げの合計が 100% になります。パーセントレイアウトは、棒グラフ、折れ線グラフ、面グラフでサポートされます。

例　グラフレイアウトの指定

次のリクエストは、積み上げレイアウトの棒グラフを生成します。グラフタイプは BAR (棒グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=stacked,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



絶対棒グラフのインセット幅の制御

絶対棒グラフでは、各棒の中心線が X 軸 (縦棒の場合) または Y 軸 (横棒の場合) の同一点を通過するよう描画されます。リクエストで最初に指定されたシリーズの棒が最背面に描画され、後続の各シリーズの棒はそれぞれの棒の前面 (手前側) に描画されます。グラフに描画される各棒の幅は自動的にサイズ調整されるため、背面の棒は両端部分のみが表示されます。absoluteInset プロパティを使用して、最前面の棒の幅 (最小幅) を最背面の棒の幅 (最大幅) のパーセントとして定義することができます。その他の棒の幅は、それに応じてサイズ調整されます。

構文　絶対棒グラフのインセット幅の制御

```
"blaProperties": {
   "absoluteInset": number
          }
```

説明

`absoluteInset:number`

最前面の棒の幅 (最小幅) を最背面の棒の幅 (最大幅) のパーセントとして定義する 0 (ゼロ) から 1 までの数値です。デフォルト値は 0.7 です。これは 70% を表します。

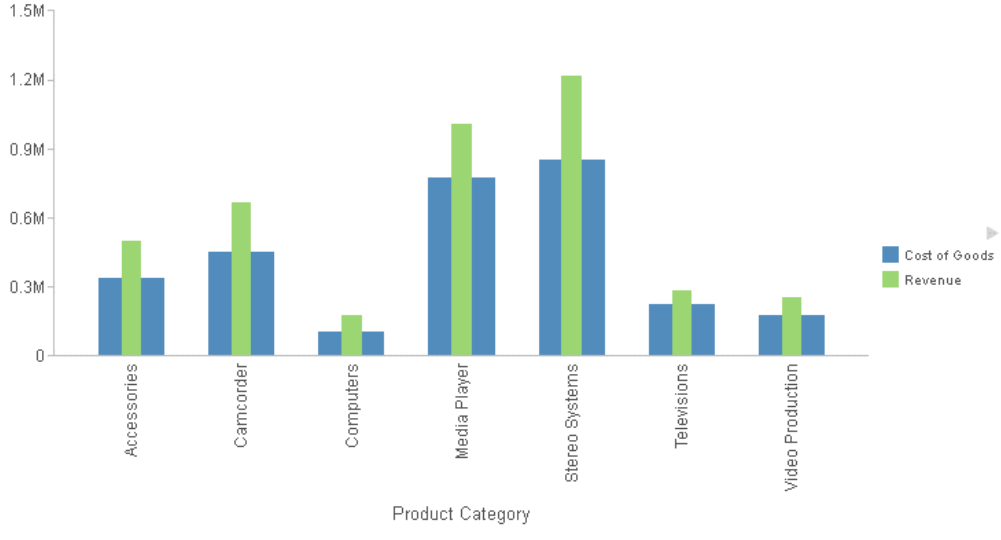
注意：absolueInset 値を 1 に設定することで、インセット効果を無効にすることができます。この場合、すべての棒が同一幅になり、棒が互いに重なり合います。

例　絶対棒グラフのインセット指定

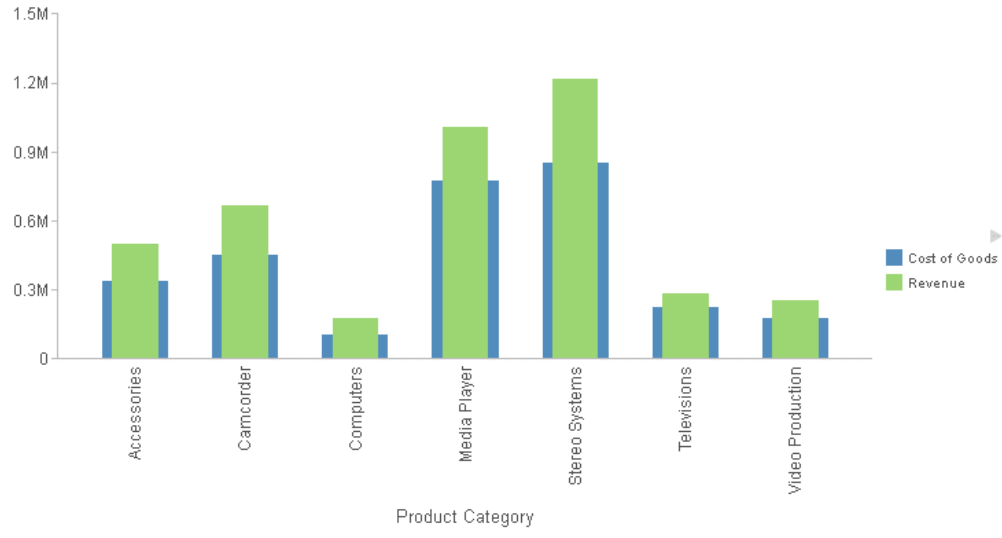
次のリクエストは、最前面の棒の幅 (最小幅) を最背面の棒の幅 (最大幅) の 30% に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=report, chart-series-layout=absolute,$
type=data, column=product_category, bucket=x-axis, $
type=data, column=cogs_us, bucket=y-axis, $
type=data, column=revenue_us, bucket=y-axis, $
*GRAPH_JS
"blaProperties": {"absoluteInset": 0.3}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



下図は、absoluteInset プロパティを指定しない場合の棒グラフを示しています。



グラフデータの属性カテゴリへの割り当て

グラフリクエスト内の各フィールドは、メジャー (集計される値) またはディメンション (メジャーを分類するフィールドで、グラフリクエストではソートフィールドとして使用される) のいずれかです。

属性カテゴリには、ソートフィールドのみに割り当てられるもの、メジャーのみに割り当てられるもの、また両方に割り当てられるものがあります。

リクエスト内のソートフィールドの順序および生成されるグラフタイプによって、各フィールドに割り当てられるカテゴリが決まります。

グラフ統合パラメータの GRMERGE、GRMULTIGRAPH、GRLEGEND、GRXAXIS は、フィールドの属性カテゴリへの割り当てに置き換えられます。統合パラメータを含むリクエストのグラフ属性構文への変換についての詳細は、1061 ページの「 リクエストのグラフ属性構文への変換 」 を参照してください。

構文　グラフデータの属性カテゴリへの割り当て

```
TYPE=DATA, COLUMN=colspec, bucket=category,$
```

説明

`COLUMN=colspec`

WebFOCUS スタイルシートの有効な列参照です。

たとえば、COLUMN=N1 は、出力の 1 列目を表します。これは上位の BY フィールドになります。詳細は、125 ページの「 列順序番号についての理解 」 を参照してください。別の方法として、フィールド名を指定することもできます (例、COLUMN=PRODUCT_CATEGORY)。

`bucket=category`

グラフ内でこのフィールドが割り当てられた役割を表します。

有効な属性カテゴリの値はグラフタイプによって異なります。以下はすべての属性カテゴリのリストです。

- **row** - マトリックスまたはグリッド内の行にフィールド値を割り当てます。
- **column** - マトリックスまたはグリッド内の列にフィールド値を割り当てます。
- **color** - 凡例にフィールドを割り当てます。フィールドがメジャーの場合は、グラデーション表示の凡例を生成します。
- **measure** - グラフ上のデータ値としてフィールド値を割り当てます。軸との関連付けはありません (円グラフの場合と同様)。
- **y-axis** - 縦軸にフィールド値を割り当てます (CHART-SERIES LAYOUT にデフォルト値を使用した場合)。
- **x-axis** - 横軸にフィールド値を割り当てます (CHART-SERIES LAYOUT にデフォルト値を使用した場合)。
- **size** - マーカー (バブルグラフ) のサイズまたはサイズフィールドにフィールド値を割り当てます。
- **tooltip** - ツールヒントにフィールド値を追加します。リクエスト内のフィールドはすべてツールヒントに追加することができます。
- **detail** - 追加の下位ソートフィールドにフィールド値を割り当てます。これによって追加のデータポイントが生成されます。このフィールドは、リクエスト内の最下位フィールドとして追加する必要があります。

- **slider** - スライダコントロールにフィールド値を割り当てます。スライダは、スライダバーコントロールの移動に合わせて動作するアニメーション効果を作成することができます。スライダは、1 つのソートフィールドのみに制限され、YEAR など時間や順序に関連するフィールドに適しています。
- **page** - 独立したページ (別個のグラフ) にフィールド値を割り当てます。page 属性カテゴリは、ReportCaster を使用しバーストや配信にも使用することができます。
- **latitude** - バブルマップ上のシンボルを表示する緯度にフィールド値を割り当てます。この属性は、バブルグラフの y-axis 属性に相当します。
- **longitude** - バブルマップ上のシンボルを表示する経度にフィールド値を割り当てます。この属性は、バブルグラフの x-axis 属性に相当します。
- **location** - コロプレスマップ上の色分けされた多角形やバブルマップ上のシンボルにロケーション値としてフィールド値を割り当てます。

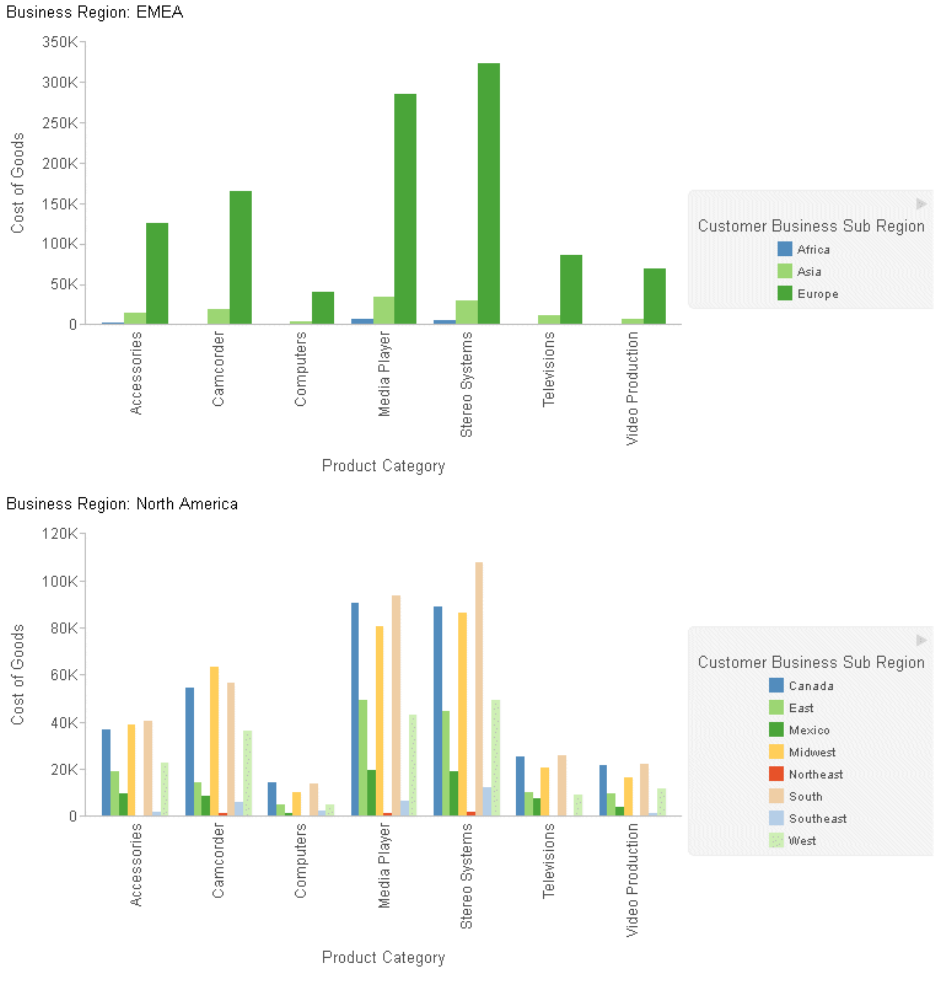
属性カテゴリによっては、グラフ出力に表示したくない場合や、生成されるグラフタイプに適用されない場合には省略することができます。また、リクエスト内の複数フィールドに割り当てることができる属性カテゴリもあります (棒グラフや折れ線グラフの y-axis に複数のフィールドが割り当てられる場合など)。

例　グラフページの生成

次のリクエストは、ビジネス地方区分 (BUSINESS_REGION) ごとに別個の棒グラフを生成します。BUSINESS_REGION フィールドは、1 番目のソートフィールドに追加され、page カテゴリに割り当てられています。BUSINESS_REGION の値は見出しにも追加され、グラフに埋め込まれます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING
"Business Region: <BUSINESS_REGION "
SUM COGS_US
BY BUSINESS_REGION
BY BUSINESS_SUB_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=X-AXIS, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=page,$
TYPE=DATA, COLUMN = BUSINESS_SUB_REGION, BUCKET=color,$
ENDSTYLE
END
```

出力結果は次のとおりです。ビジネス地方区分ごとに別個のグラフが生成されます。



ビジネス地方区分を示す見出しを追加する代わりに、page カテゴリのほかに tooltip カテゴリにもフィールドを追加することができます。詳しい説明と例については、205 ページの 「 複数の属性カテゴリへのフィールドの割り当て 」 を参照してください。

カテゴリ内のフィールド順序の割り当て

カテゴリによっては複数のフィールドが割り当てられます。デフォルト設定では、スタイルシートに表示される割り当て順序でフィールドがカテゴリに割り当てられます。たとえば、1 番目に割り当てられた y-axis フィールドが、縦棒グラフの一番左側のメジャーになります。カテゴリに割り当てられるフィールドの順序を指定する場合は、インデックス番号の接尾語を使用します。たとえば、次の宣言を使用した場合、UNITS はリクエスト内の順序にかかわらず 1 番目になります。ここでは分かりやすいように UNITS の接尾語 (.1) を BUDUNITS の接尾語 (.2) の下に置いています。

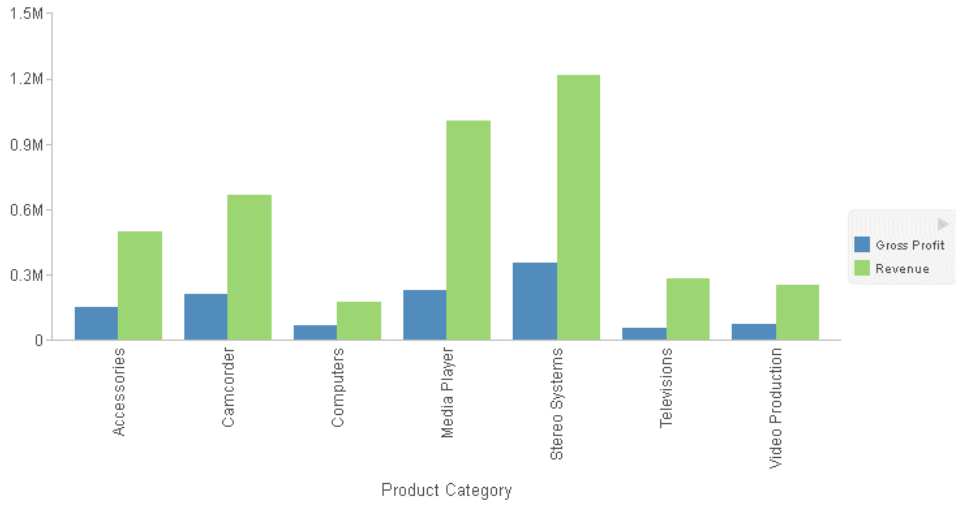
```
type=data, column=budunits, bucket=y-axis.2, $
type=data, column=units, bucket=y-axis.1, $
```

例　フィールド順序の指定

次のリクエストは、棒グラフを生成します。棒は、GROSS_PROFIT_US および REVENUE_US のメジャーを表します。スタイルシート宣言では、GROSS_PROFIT_US が REVENUE_US より前に指定されているため、GROSS_PROFIT_US が 1 番目に表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM
GROSS_PROFIT_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis, $
type=data, column=gross_profit_us, bucket=y-axis, $
type=data, column=revenue_us, bucket=y-axis, $
ENDSTYLE
END
```

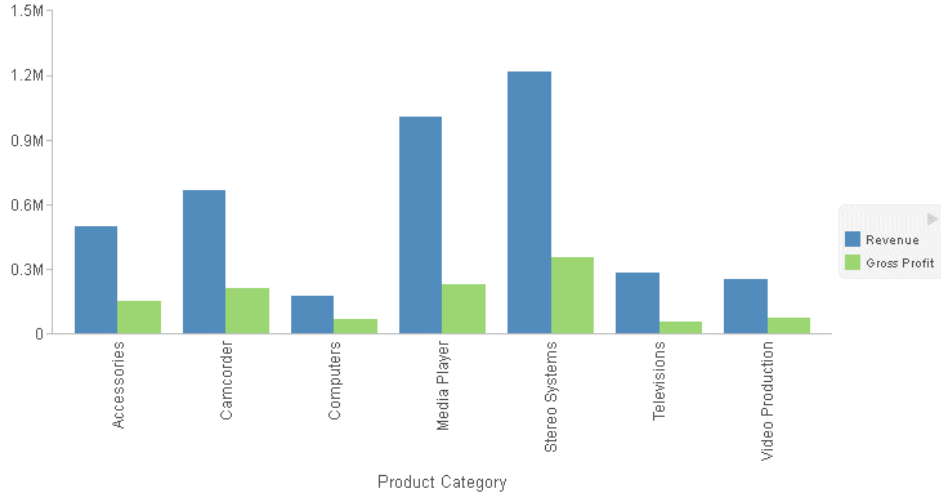
出力結果は次のとおりです。



同じリクエストでも次のバージョンでは、REVENUE_US フィールドの属性カテゴリに接尾語 .1 が追加されているため、REVENUE_US を示す棒が 1 番目に表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM
GROSS_PROFIT_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis, $
type=data, column=gross_profit_us, bucket=y-axis, $
type=data, column=revenue_us, bucket=y-axis.1, $
ENDSTYLE
END
```

出力結果は次のとおりです。



複数 X 軸および複数 Y 軸の指定

複数のバージョンが存在する場合、属性カテゴリには括弧 ( ) 内にサブスクリプト値を含めることができます。たとえば、y2-axis を示すには以下を使用します。

```
y-axis(2)
```

サブスクリプト値を使用して Y 軸または X 軸の数を任意に指定することができます。

棒グラフ、折れ線グラフ、面グラフで複数 Y 軸を使用できるかどうかは、blaProperties:splitY プロパティの設定 (true または false) によります。詳細は、530 ページの 「 棒グラフ、折れ線グラフ、面グラフの複数 Y 軸表示の制御 」 を参照してください。

バブルグラフまたは散布図では、サブスクリプト値を使用してフィールドを適切な軸に割り当てることで、分割 X 軸、分割 Y 軸、またはその両方を使用することができます。以下はその例です。

```
type=data, column=COGS_US,      bucket=x-axis, $
type=data, column=REVENUE_US,   bucket=x-axis(2), $
type=data, column=DISCOUNT_US,  bucket=y-axis, $
type=data, column=MSRP_US,      bucket=y-axis(2), $
```

カテゴリには、サブスクリプト値と接尾語の両方を含めることができます。たとえば、y2-axis カテゴリを 1 番目のフィールドに指定するには以下を使用します。

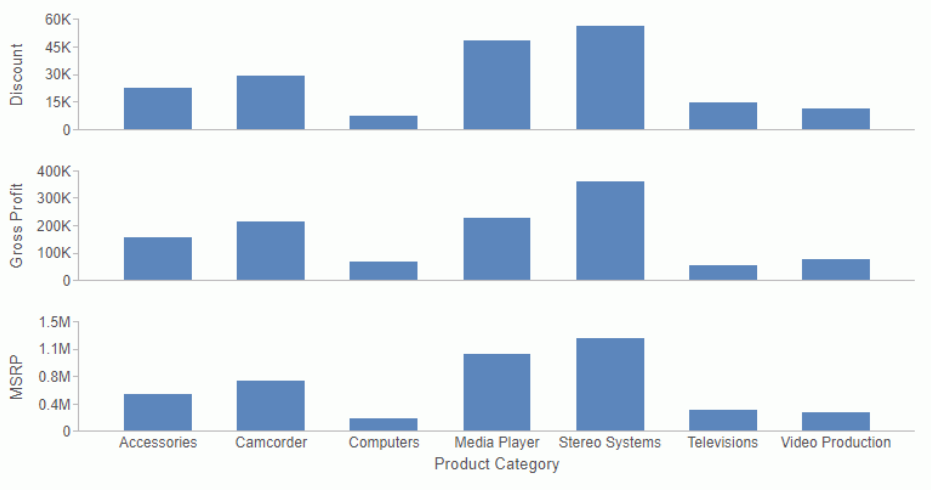
```
y-axis(2).1
```

例　棒グラフでの分割 Y 軸の生成

次のリクエストは、メジャー DISCOUNT_US を y-axis、メジャー GROSS_PROFIT_US を y2-axis、メジャー MSRP_US を y3-axis に割り当てています。ここでは、blaProperties:splitY プロパティが true に設定されているため、分割 Y 軸が描画されます (積み上げ)。chart-color-measures=off スタイルシート属性により、メジャーはすべて同一色で表示されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=discount_us, bucket=y-axis,$
type=data, column=GROSS_PROFIT_US, bucket=y-axis(2),$
type=data, column=MSRP_US, bucket=y-axis(3),$
type=report, chart-color-measures=off,$
*GRAPH_JS
"blaProperties": {"splitY": true},
"legend": {"visible": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

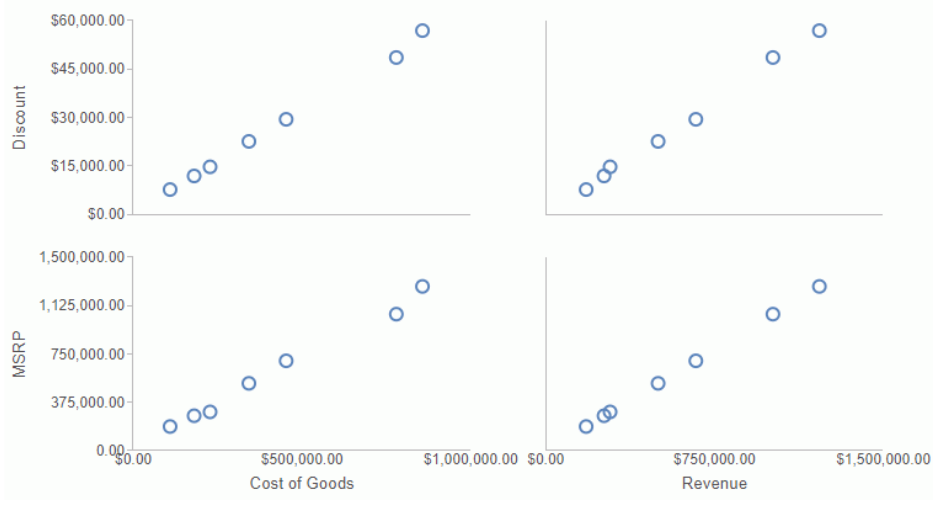


例　散布図での分割 X 軸および 分割 Y 軸の生成

次のリクエストは、分割 X 軸および分割 Y 軸を使用した散布図を生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,    $
type=data, column=PRODUCT_CATEGORY,   bucket=detail,    $
type=data, column=COGS_US,            bucket=x-axis,    $
type=data, column=REVENUE_US,         bucket=x-axis(2), $
type=data, column=DISCOUNT_US,        bucket=y-axis,    $
type=data, column=MSRP_US,            bucket=y-axis(2), $
END
```

出力結果は次のとおりです。



複数の属性カテゴリへのフィールドの割り当て

複数の属性カテゴリにフィールドを割り当てるには、括弧 ( ) 内にカテゴリのリストを含め、これらをブランクで区切ります。

```
TYPE=DATA, COLUMN=colspec, bucket=(category1 category2 ...),$
```

たとえば、次の宣言は page と tooltip の両カテゴリに 1 つのフィールドを追加します。

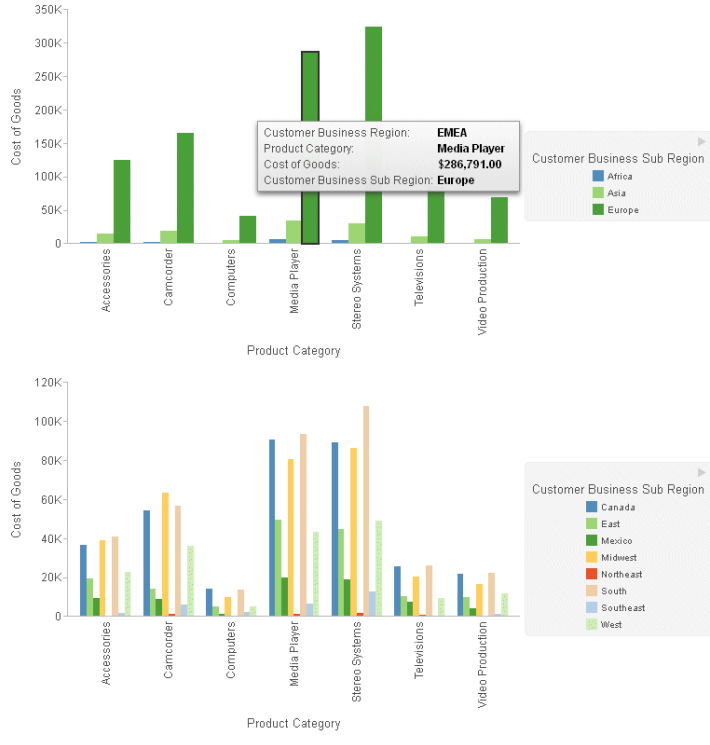
```
TYPE=DATA, COLUMN=N1, BUCKET=(page tooltip),$
```

例　ページおよびツールヒントカテゴリへのフィールドの割り当て

次のリクエストは、ビジネス地方区分 (BUSINESS_REGION) ごとに別個の棒グラフを生成します。BUSINESS_REGION フィールドが 1 番目のソートフィールドとして追加され、page と tooltip のカテゴリに割り当てられています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY BUSINESS_SUB_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_SUB_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=(page tooltip),$
ENDSTYLE
END
```

下図は、BUSINESS_REGION が新しいページを生成し、ツールヒントに追加されたことを示しています。



ソートフィールドに対する属性カテゴリの割り当て順序

ソートフィールドに対する属性カテゴリの割り当ては、次の順序に従います。

1. page
2. slider
3. row
4. column
5. color
6. detail
7. location (マップの場合) または x-axis (マップ以外の場合)

注意

- バブルマップの latitude と longitude の属性は、バブルグラフの x-axis と y-axis の属性に対応します。

- ヒートマップの場合、y-axis はソートディメンションになり、リクエスト内で x-axis ソートフィールドの前に挿入する必要があります。
- バブルグラフおよび散布図の場合、x-axis と y-axis の両方がソートディメンションになります。これらは、リクエスト内に任意の順序で指定することができます。

グラフタイプ別のサポート可能な属性カテゴリの概要

マップグラフとタグクラウド以外のグラフタイプはすべて、複数行列 の BY フィールドの割り当てをサポートします。また、すべてのグラフタイプで複数ページおよびツールヒントの BY フィールドの割り当てが可能です。slider カテゴリに割り当てられる BY フィールドは 1 つのみです。

グラフ属性構文では、ACROSS ソートフィールドはサポートされません。ソートフィールドはすべて BY フィールドです。

下表は、サポート可能なその他の属性カテゴリをグラフタイプ別に示しています。

グラフタイプ	属性カテゴリおよび説明
面グラフ	**x-axis** ソートフィールドです。通常は YEAR など時間ベースのディメンションです。 **y-axis** メジャーです。複数がサポートされます。2 つの Y 軸がサポートされます。 **color** オプションです。面の色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。複数がサポートされます。

グラフタイプ	属性カテゴリおよび説明
棒グラフ	**x-axis** ソートフィールドです。 **y-axis** メジャーです。複数がサポートされます。複数の Y 軸がサポートされます。 **color** オプションです。棒の色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。複数がサポートされます。 **size** オプションです。棒の幅は、size 属性に割り当てられたメジャー値に基づきます。
ボックスプロット	**x-axis** ソートフィールドです。 **min** メジャーです。最小値です。 **lower** メジャーです。第 1 四分位です。 **median** メジャーです。中央値です。 **upper** メジャーです。第 3 四分位です。 **max** メジャーです。最大値です。 **color** オプションです。ボックスの色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。複数がサポートされます。

グラフタイプ	属性カテゴリおよび説明
バブルグラフ	**x-axis** メジャーまたはソートフィールドです。 **y-axis** メジャーまたはソートフィールドです。複数がサポートされます。 **color** オプションです。シンボルの色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。 **size** メジャーです。バブルのサイズは、size 属性に割り当てられたメジャー値に基づきます。 **detail** ソートフィールドです (オプション)。グラフにデータを追加します。複数がサポートされます。
バブルマップ	**location** ディメンションソートフィールドです。マップ上でシンボルが表示される地理位置情報を含みます。使用されるのは location または latitude および longitude のいずれか一方です。 **latitude** ソートフィールドまたはメジャーフィールドです。シンボルの緯度を含みます。使用されるのは location または latitude および longitude のいずれか一方です。 **longitude** ソートまたはメジャーフィールドです。シンボルの経度を含みます。使用されるのは location または latitude および longitude のいずれか一方です。 **size** メジャーフィールドです。シンボルのサイズを決めるのに使用されます。 **color** オプションです。シンボルの色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。

グラフタイプ	属性カテゴリおよび説明
コロプレスマップ	**location** ディメンションソートフィールドです。マップ上でシンボルが表示される地理位置情報を含みます。 **color** 多角形の色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。
データグリッド	**row** ソートディメンションまたはメジャーフィールドです。値はグリッドの行を表します。複数がサポートされます。 **column** ソートディメンションまたはメジャーフィールドです。値はグリッドの列を表します。複数がサポートされます。 **measure** グリッドのセルに表示される値のフィールドです。複数がサポートされます。
じょうごグラフ	**measure** じょうご項目のサイズを決めるメジャーフィールドです。 **color** ソートフィールドです。
メータグラフ	**measure** レイアウトに基づいてメータ針の位置およびバンドの幅を決めるフィールドです。
ヒートマップ	**y-axis** ソートディメンションフィールドです。リクエスト内で、x-axis フィールドより前に指定する必要があります。 **x-axis** ソートフィールドです。 **color** メジャーです。長方形の色は、このメジャーの値に基づきます。この場合、色はグラデーションとして表示されます。

グラフタイプ	属性カテゴリおよび説明
折れ線グラフ	**x-axis** ソートフィールドです。通常は YEAR など時間ベースのディメンションです。 **y-axis** メジャーです。複数がサポートされます。複数の Y 軸がサポートされます。 **color** メジャーです (オプション)。線の色 (およびマーカーが表示される場合はマーカーの色) は、color 属性に割り当てられたメジャーに基づきます。この場合、色はグラデーションとして表示されます。複数がサポートされます。 **size** メジャーです (オプション)。マーカーが表示される場合、マーカーのサイズは size 属性に割り当てられたメジャー値に基づきます。マーカーが表示されない場合、線の幅が size メジャーを表します。
マトリックスマーカーグラフ	**size** メジャーです。各セルのマーカーのサイズは、size 属性に割り当てられたメジャー値に基づきます。 **color** メジャーです。マーカーの色は、color 属性に割り当てられたメジャーに基づきます。この場合、色はグラデーションとして表示されます。
マリメッコチャート	**x-axis** ソートフィールドです。 **y-axis** メジャーです。複数がサポートされます。 **color** オプションです。棒の色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。複数がサポートされます。
円グラフ	**measure** 各項目のサイズを決めます。 **color** ソートフィールドです。項目の色分けに使用されます。 **size** マトリックスグラフのみでサポートされます。割り当てられたメジャーに基づいて、各セルの円のサイズを拡大縮小します。

グラフタイプ	属性カテゴリおよび説明
散布図	**x-axis** メジャーまたはディメンションソートフィールドです。 **y-axis** メジャーまたはディメンションです。 **color** マーカーの色は、color 属性に割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがソートフィールドの場合、色は不連続の色で表示されます。このカテゴリにフィールドが割り当てられていない場合、マーカーはすべて同一色になります。 **size** メジャーです (オプション)。マーカーのサイズは、size 属性に割り当てられたメジャー値に基づきます。 **detail** ソートフィールドです (オプション)。グラフにデータを追加します。複数がサポートされます。
統計ボックスプロット	**x-axis** ソートフィールドです (ディメンション)。 **y-axis** メジャーです。複数がサポートされます。
ストリームグラフ	**x-axis** ソートフィールドです。 **y-axis** メジャーです。複数がサポートされます。 **color** エリアおよび凡例の色です (オプション)。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。複数がサポートされます。
タグクラウド	**detail** 表示されるタグを含むソートディメンションです。 **size** タグのサイズを制御するメジャーです。 **color** タグの色を制御するメジャーです。タグクラウドは複数のシリーズや凡例をサポートしないため、このカテゴリはソートフィールドに割り当てることができません。

グラフタイプ	属性カテゴリおよび説明
ツリーマップ	**color**　メジャーまたはソートフィールドです。長方形の色は、このメジャーまたはディメンションの値に基づきます。色は、メジャーの場合はグラデーションとして表示され、ディメンションの場合は不連続の色で表示されます。 **size**　メジャーです。長方形のサイズは、このメジャーの値によって決まります。 **detail**　ディメンションソートフィールドです。複数がサポートされます。

5

グラフ全体のプロパティ

ここでは、グラフの全体的な外観を制御する方法について説明します。

トピックス

- グラフ全体のプロパティの概要
- グラフ境界線のフォーマット設定
- JavaScript エラーの処理方法の定義
- グラフフレームのフォーマット設定
- 1 行に表示するグラフ数の制御
- グラフに描画するデータ範囲の定義
- グラフへの奥行きの適用
- 描画領域への色またはグラデーションの適用
- グループラベルの定義
- ライザ、マーカー、円項目へのベベル効果の適用
- 連続するライザへの濃淡効果の適用
- 3D グラフのライザ間隔の制御
- シリーズとグループの位置の入れ替え
- シリーズ、グループ、ラベルの位置の入れ替え

グラフ全体のプロパティの概要

ここでは、グラフの全体的な外観を制御するプロパティについて説明します。

次のコードセグメントは、デフォルト設定を示しています。

```
"catchErrors": true,
"fill": {
"color": "white"
},

"border": {
    "width": 1,
    "color": "black",
    "dash": ""
    },
```

```
"chartFrame": {
    "fill": {
        "color": "transparent"
    },
    "border": {
        "width": 0,
        "color": "transparent",
        "dash": ""
    }
},

"dataSubset": {
    "startGroup": "undefined",
    "stopGroup": "undefined"
},

"swapData": false,

"swapDataAndLabels": false,

"riserCycleEndLightness": 0.8,

"chartsPerRow": undefined,

"depth": undefined,
"riserDepthGap": 0.2,
"riserBevel": undefined,

"groupLabels": "ABCDEFGHIJKLMNOPQRSTUVWXYZ".split(''),
```

グラフ境界線のフォーマット設定

border プロパティは、描画領域の境界線を定義します。

グラフ境界線は、描画エリアを囲みます。描画エリアには、次の項目が含まれています。

- グラフとフレーム
- 軸ラベル
- 凡例
- グラフタイトル
- サブタイトル
- 脚注

構文　グラフ境界線のフォーマット設定

```
"border": {
   "width": number,
   "color": "color",
   "dash": "string"
}
```

説明

`"width": number`

描画領域を囲む境界線の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です (境界線なし)。

`"color": "color"`

次のいずれかの値を指定します。

色名または数値指定で色を定義する文字列、またはグラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "transparent" です。

グラデーションを定義する JSON オブジェクト。

色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"dash": "string"`

境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

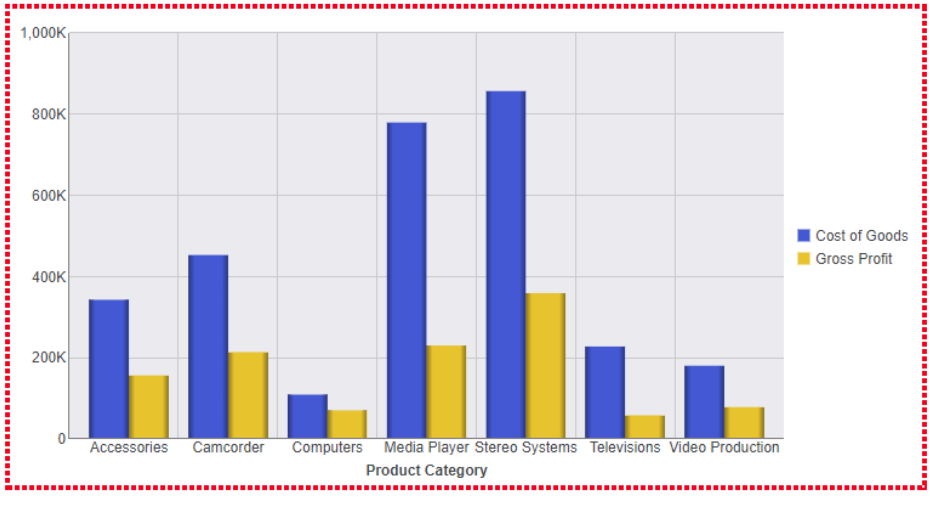
破線を定義する文字列では、複数の破線スタイルを指定することができます。たとえば、「"2 4 4 2"」と指定すると、境界線の破線スタイルが交互に入れ替わります。

例　グラフ境界線のフォーマット設定

次のリクエストは、赤色の 4 ピクセル幅で境界線を生成し、破線スタイルを破線長さ 4 ピクセル、破線間隔 2 ピクセルに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=product_category, bucket = x-axis,$
type=data, column=COGS_US, bucket = y-axis,$
type=data, column=GROSS_PROFIT_US, bucket = y-axis,$
*GRAPH_JS
"border": {
  "width": 4,
  "color": "red",
  "dash": "4 2"
         }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



JavaScript エラーの処理方法の定義

catchErrors プロパティは、JavaScript エラーの処理方法を定義します。代表的なエラーには、プロパティ設定が無効なためにエンジンが JavaScript を解析できない場合に発生するエラーがあります。catchErrors が false の場合は、JavaScript コンソールを使用して、エラーの正確な発生場所を特定することができます。JavaScript コンソールを表示するには、たとえば Chrome では Ctrl+Shift+J キーを押します。Firefox では F12 キーを押し、[コンソール] タブをクリックします。

構文　JavaScript エラーの処理方法の指定

```
"catchErrors": boolean
```

説明

`"catchErrors": boolean`
次のいずれかの値を指定します。

- true - エラーを HTML テキストとして表示します。エラーが発生しない場合、グラフが描画されます。これがデフォルト値です。呼び出し元アプリケーションの後続の JavaScript コードは引き続き実行されます。
- false - 呼び出し元アプリケーションにエラーを転送します。呼び出し元アプリケーションがエラーを検出しない場合、後続の JavaScript 実行がすべて終了します。

グラフフレームのフォーマット設定

chartFrame プロパティは、グラフフレームの塗りつぶし色および境界線を定義します。

グラフの表示領域は、グラフフレームで囲まれます。グラフフレーム内には、軸ラベル、凡例、グラフのタイトル、サブタイトル、脚注は含まれません。

注意：円グラフではグラフフレームがサポートされません。代わりに、背景の塗りつぶしを使用します。

構文　グラフフレームのフォーマット設定

```
"chartFrame": {
   "fill": {
      "color": "color"
           },
   "border": {
      "width": number,
      "color": "string",
      "dash": "string"
           }
             }
```

説明

`"fill": {"color": "color" }`
塗りつぶし色です。次のいずれかの値を指定します。

色名または数値指定の文字列で色を定義する文字列、あるいはグラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "transparent" です。

グラデーションを定義する JSON オブジェクト。

色およびグラデーションの指定についての詳細は、63 ページの「 色およびグラデーション 」 を参照してください。

`"border":`
フレーム境界線のプロパティを定義します。

`"width": number`

フレーム境界線の幅をピクセル単位で指定します。デフォルト値は 0 (ゼロ) です (境界線なし)。

`"color": "string"`
フレームを囲む境界線の色です。次のいずれかの値を指定します。

色名または数値指定の文字列で色を定義する文字列、あるいはグラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "transparent" です。

グラデーションを定義する JSON オブジェクト。

`"dash": "string"`

境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

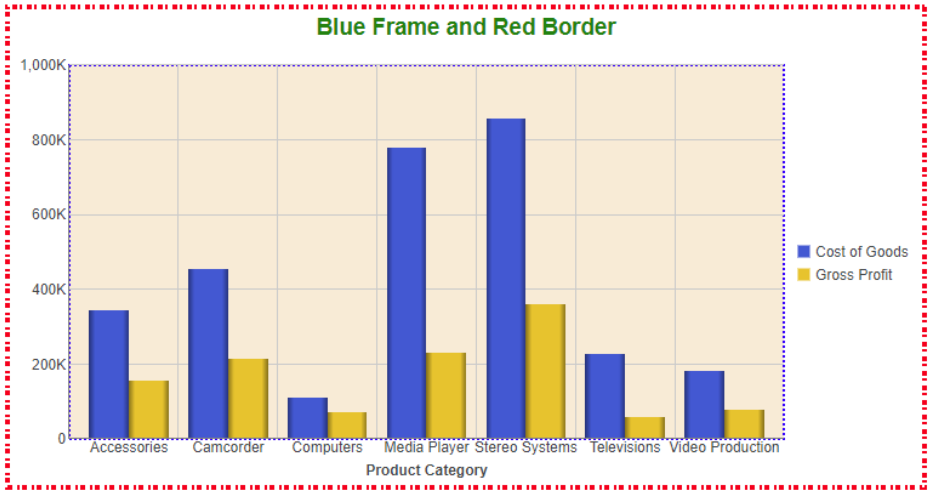
破線を定義する文字列では、複数の破線スタイルを指定することができます。たとえば、「"2 4 4 2"」と指定すると、境界線の破線スタイルが交互に入れ替わります。

例　グラフフレームのフォーマット設定

次のリクエストは、グラフフレームを生成し、グラフフレームの塗りつぶし色をアンティークホワイト、境界線を青色の破線に設定します。また、描画領域の境界線を赤色の破線で生成し、サイズの異なる 2 種類の破線スタイルを交互に入れ替えます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=product_category, bucket = x-axis,$
type=data, column=COGS_US, bucket = y-axis,$
type=data, column=GROSS_PROFIT_US, bucket = y-axis,$
*GRAPH_JS
"title": {"visible": true,
   "text": "Blue Frame and Red Border",
   "color": "green",
   "font": "bold 14pt Sans-Serif"
        },
"chartFrame": {"fill": {"color": "antiquewhite"},
   "border": {
      "width": 2,
      "color": "blue",
      "dash": "2 2"
       }},
"border": {
  "width": 4,
  "color": "red",
  "dash": "2 4 4 2"
        }
*END
ENDSTYLE
END
```

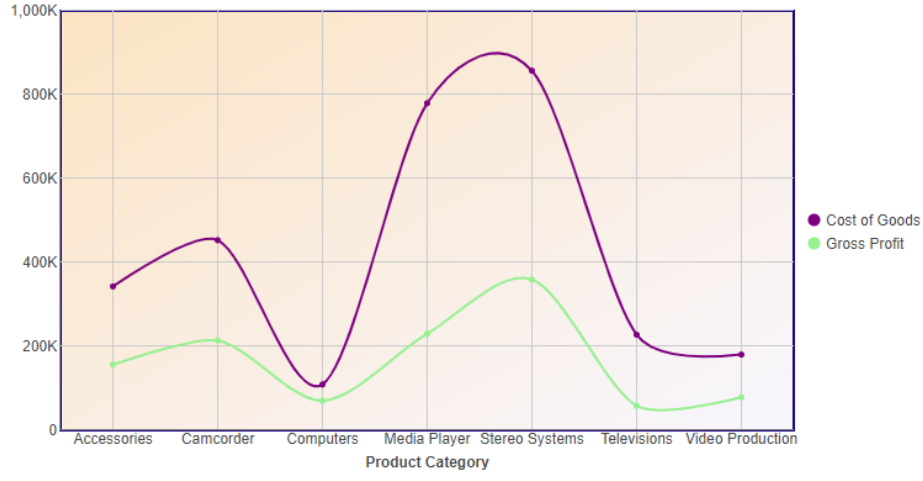
出力結果は次のとおりです。



次の例では、グラフフレームにビスクからゴーストホワイトに変化する線状グラデーションを適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=product_category, bucket = x-axis,$
type=data, column=COGS_US, bucket = y-axis,$
type=data, column=GROSS_PROFIT_US, bucket = y-axis,$
*GRAPH_JS
"blaProperties": {"lineConnection": "curved"},
"chartFrame": {"fill": {"color":
   {"type": "linear",
      "start": {"x": "0%", "y": "0%"},
      "end": {"x": "100%", "y": "100%"},
      "stops": [[0, "bisque"], [1, "ghostwhite"]]}},
"border": {"width": 2, "color": "navy"}},
"series": [
   {"series": 0, "color": "purple"},
   {"series": 1, "color": "lightgreen"},
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



1 行に表示するグラフ数の制御

円グラフ、メータグラフ、じょうごグラフでは 1 つの描画領域に複数のグラフを表示できますが、chartsPerRow プロパティを使用することで、1 行あたりに描画するグラフ数を定義することができます。

構文　1 行に表示するグラフ数の制御

```
"chartsPerRow": number
```

説明

`"chartsPerRow": number`

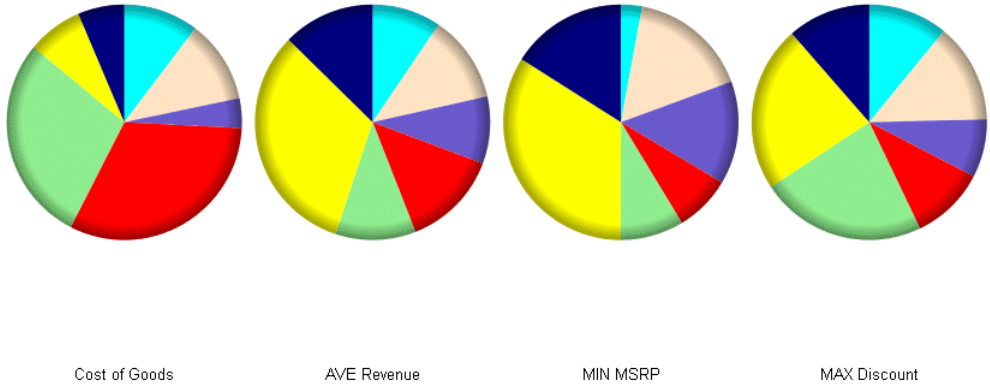
1 行に描画するグラフ数です。デフォルト値は undefined です。

例　　1 行に表示するグラフ数の制御

次のリクエストは、4 つの円グラフを生成します。chartsPerRow プロパティで、すべてのグラフが 1 行に配置されるよう設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US AVE.REVENUE_US MIN.MSRP_US MAX.DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=product_category, bucket = color,$
type=data, column=COGS_US, bucket=measure,$
type=data, column=AVE.REVENUE_US, bucket=measure,$
type=data, column=MIN.MSRP_US, bucket=measure,$
type=data, column=MAX.DISCOUNT_US, bucket=measure,$
*GRAPH_JS
"chartsPerRow": 4,
"legend": {"visible": false},
"series": [
   {"series": 0, "color": "cyan"},
   {"series": 1, "color": "bisque"},
   {"series": 2, "color": "slateblue"},
   {"series": 3, "color": "red"},
   {"series": 4, "color": "lightgreen"},
   {"series": 5, "color": "yellow"},
   {"series": 6, "color": "navy"},
   {"series": 7, "color": "lavender"},
   {"series": 8, "color": "limegreen"},
   {"series": 9, "color": "red"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフに描画するデータ範囲の定義

dataSubset プロパティは、グラフに描画するデータの範囲を定義します。これらのプロパティを interaction プロパティ (例、mouseDrag: pan) とともに使用して、表示するデータ範囲を mouseDrag ユーザ操作で定義するよう設定することができます。

構文　グラフに描画するデータ範囲の定義

```
"dataSubset": {
   "startGroup": number,
   "stopGroup": number,
}
```

説明

`"startGroup": number`

0 (ゼロ) を基準とする開始グループ番号です。このグループ番号からデータを表示します。デフォルト値は undefined です。

`"stopGroup": number`

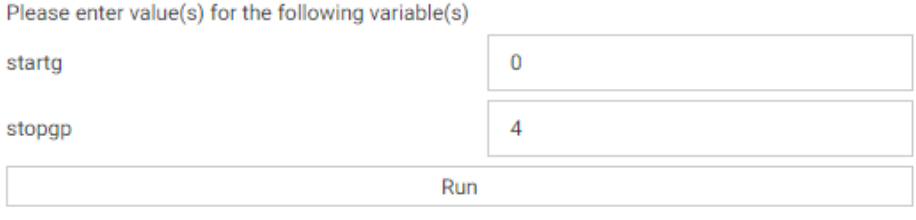
0 (ゼロ) を基準とする終了グループ番号です。このグループ番号からデータを非表示にします。デフォルト値は undefined です。

例　グラフに表示するデータ範囲の定義

次のリクエストでは、ダイアログマネージャ変数を使用して、描画するデータのサブセットをユーザが指定するよう設定します。

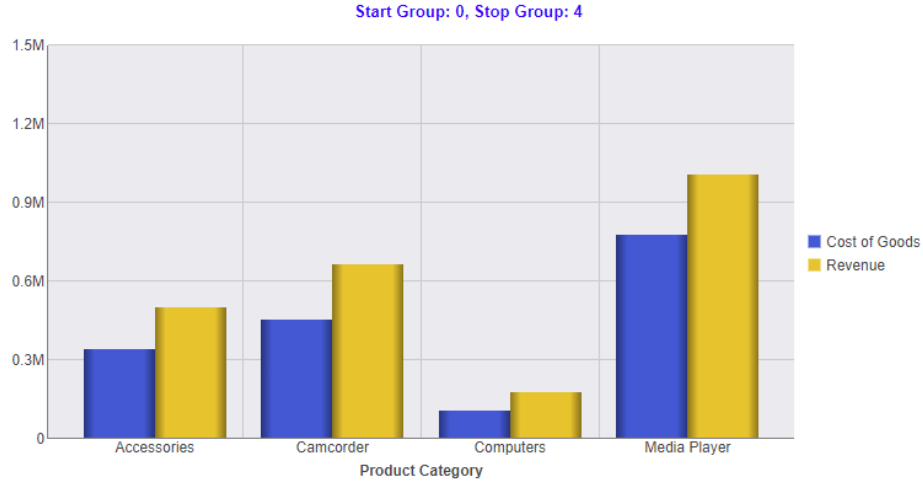
```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=y-axis,$
*GRAPH_JS
"title": {"visible": true,
   "text": "Start Group: &startg, Stop Group: &stopgp",
   "color": "blue"},
"dataSubset": {
  "startGroup": "&startg",
  "stopGroup": "&stopgp"
   }
*END
ENDSTYLE
END
```

リクエストの実行後、ユーザが開始グループ番号と終了グループ番号を入力します。この例では、開始グループ番号は 0 (ゼロ)、終了グループ番号は 4 です。



注意：WebFOCUS のオートプロンプト機能を使用して、プロシジャの実行に必要な変数値の入力をユーザに要求する起動ページを作成することができます。この機能を使用すると、受容値の動的または静的なドロップダウンリストを生成したり、ユーザが入力した値を検証したりすることができます。詳細は、『ibi™ WebFOCUS® アプリケーション作成ガイド』を参照してください。

ユーザが [実行] をクリックすると、下図のグラフが生成され、グループ 0 からグループ 3 までが表示されます。



グラフへの奥行きの適用

depth プロパティは、棒グラフ、折れ線グラフ、面グラフ、円グラフのほか、棒状のグラフに 2.5D 奥行き効果を適用します。

構文　　グラフへの奥行きの適用

```
"depth": number
```

説明

`"depth": number`

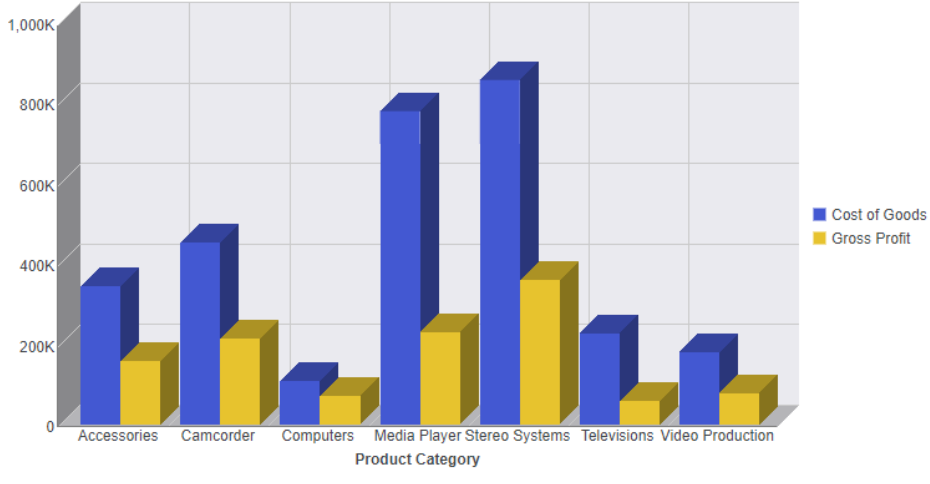
ライザおよびグラフフレームに適用する奥行きを、表示可能領域に対するパーセント値として 0 (ゼロ) から 100 までの数値を指定します。2.5D 奥行き効果を適用しない場合は、0 (ゼロ) または undefined を指定します。デフォルト値は undefined です。

例　　グラフへの奥行きの適用

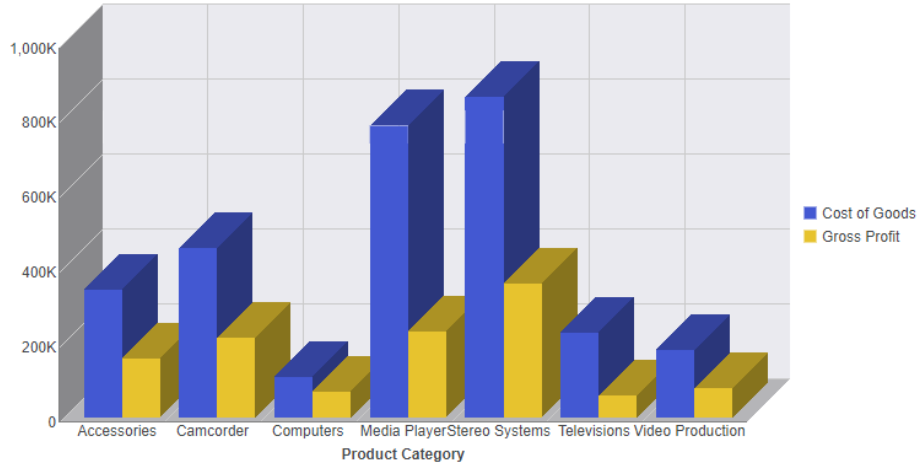
次のリクエストは、縦棒グラフを生成し、depth プロパティを 25 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=Y-AXIS,$
*GRAPH_JS
  "depth": 25
*END
ENDSTYLE
END
```

出力結果では、グラフが 2.5D 奥行き効果で描画されています。



depth プロパティの値を 50 に変更すると、さらに奥行きのある 2.5D 効果がグラフに適用されます。



描画領域への色またはグラデーションの適用

fill プロパティは、描画領域に適用する色またはグラデーションを定義します。

構文　描画領域への色またはグラデーションの適用

```
"fill": {
        "color": "color"}
```

説明

`"color": "color"`

塗りつぶし色を定義します。次のいずれかの値を指定します。

色名または数値指定で色を定義する文字列、あるいはグラデーションを定義する文字列。デフォルト値は "white" です。

グラデーションを定義する JSON オブジェクト。

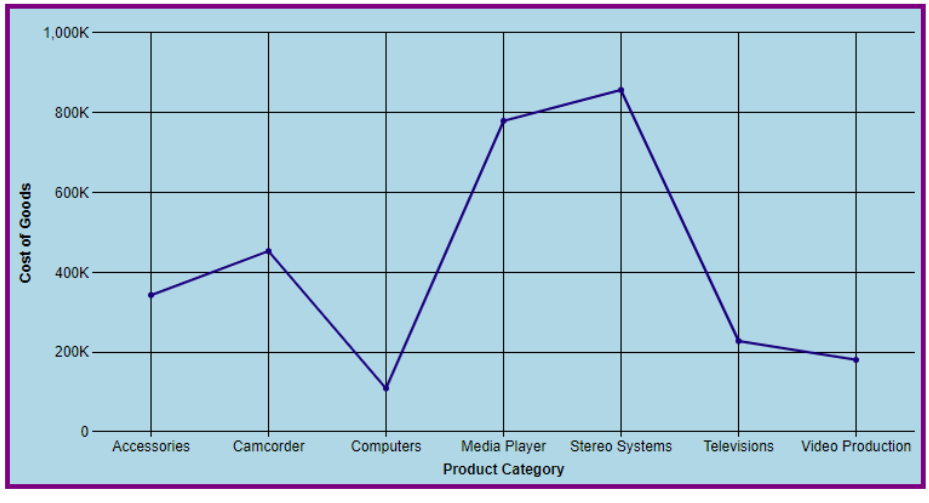
色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　描画領域の塗りつぶし色またはグラデーションの指定

次のリクエストは、縦折れ線グラフを生成し、塗りつぶし色を水色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=x-axis,$
type=data, column=cogs_us, bucket=y-axis,$
*GRAPH_JS
"fill": {"color": "lightblue"},
"border": {"width": 4, "color": "purple"},
"series": [
   {"series": 0, "color": "navy"},
   ]
*END
ENDSTYLE
END
```

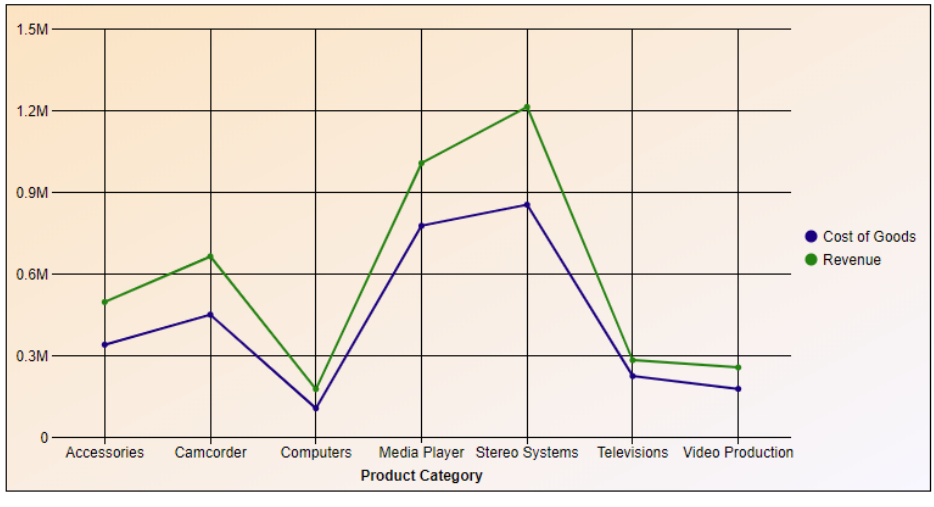
出力結果は次のとおりです。



次のリクエストでは、描画領域に線状グラデーションを適用する JSON オブジェクトを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=Y-AXIS,$
*GRAPH_JS
"fill": {"color": {"type": "linear",
   "start": {"x": "0%", "y": "0%"},
   "end": {"x": "100%", "y": "100%"},
   "stops": [[0, "bisque"], [1, "ghostwhite"]]}
   },
"series": [
   {"series": 0, "color": "navy"},
   {"series": 1, "color": "green"}
   ]
*END
ENDSTYLE
END
```

ビスクからゴーストホワイトに変化するグラデーションが描画領域に適用されています。



グループラベルの定義

すべてのグラフタイプでは (ヒストグラムおよびタグクラウドを除く)、groupLabels プロパティを使用して、X 軸に表示するグループラベルを定義することができます。タグクラウドでは、このプロパティを使用して、グラフのデータ値に基づいて表示されるラベルを定義します。すべてのグラフタイプでは (タグクラウドを除く)、グループラベルを文字列または文字列配列として定義することができます。タグクラウドでは、グループラベルを文字列配列として定義する必要があります。

構文　グループラベルの定義

```
"groupLabels": "string"
```

または

```
"groupLabels": ["string", ..., "string"]
```

説明

`"groupLabels": "string"`

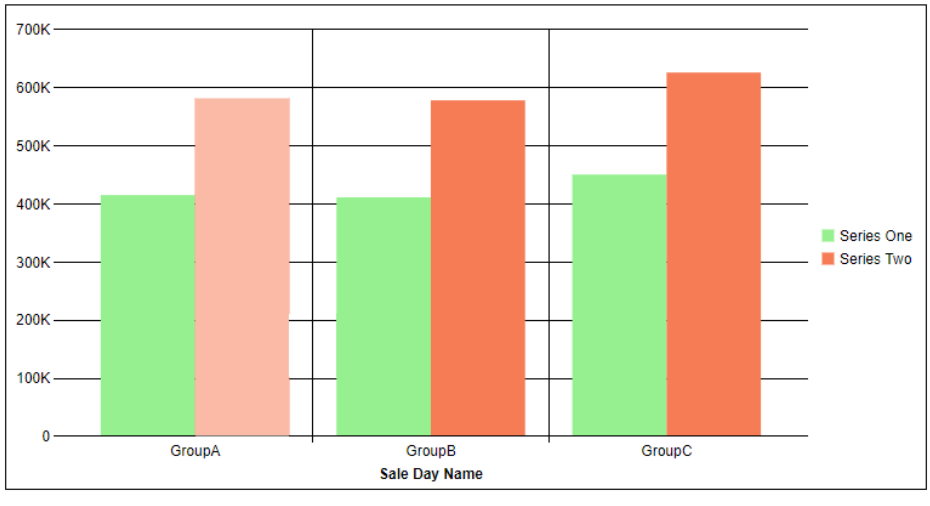
文字列または文字列配列です。文字列にブランクを含めると、文字列がその位置で分割され、文字列配列になります。デフォルト値は "ABCDEFGHIJKLMNOPQRSTUVWXYZ".split('') です。

例　グループラベルを文字列として定義

次のリクエストでは、グループラベルを「"GroupA GroupB GroupC"」という文字列として指定します。文字列内のブランクの位置で、前のグループラベルが終了し、次のグループラベルが開始されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'FRI' OR 'SAT' OR 'SUN'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=time_dayname, bucket=x-axis,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=revenue_us, bucket=y-axis,$
*GRAPH_JS_FINAL
"groupLabels": "GroupA GroupB GroupC",
"series": [
   {"series": 0, "label": "Series One", "color": "lightgreen"},
   {"series": 1, "label": "Series Two", "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

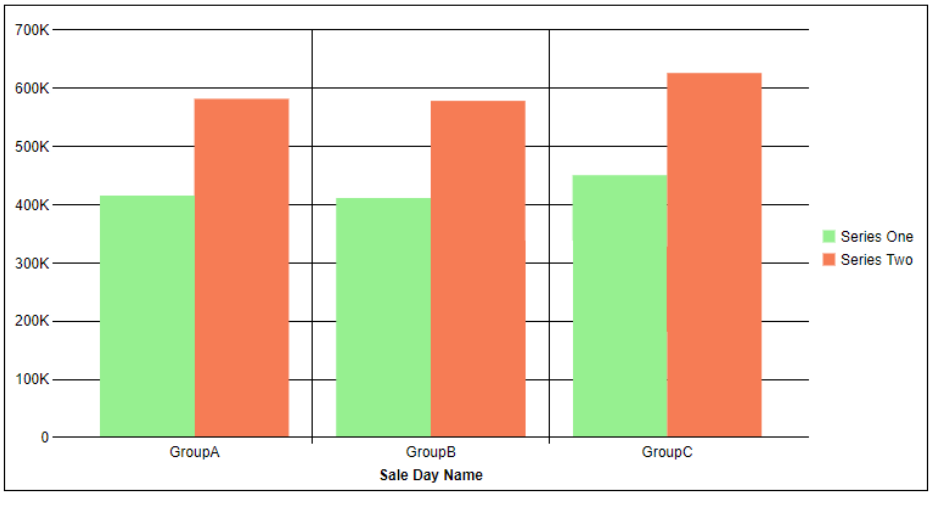


例　グループラベルを配列として定義

次のリクエストでは、グループラベルを「"GroupA", "GroupB", "GroupC"」という配列として指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'FRI' OR 'SAT' OR 'SUN'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=time_dayname, bucket=x-axis,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=revenue_us, bucket=y-axis,$
*GRAPH_JS_FINAL
"groupLabels": ["GroupA", "GroupB", "GroupC"],
"series": [
   {"series": 0, "label": "Series One", "color": "lightgreen"},
   {"series": 1, "label": "Series Two", "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ライザ、マーカー、円項目へのベベル効果の適用

riserBevel プロパティは、棒グラフのライザ、バブルグラフおよび散布図のマーカー、円グラフの円項目にベベル効果を適用します。

構文　　ライザ、マーカー、円項目へのベベル効果の適用

```
"riserBevel": "string"
```

説明

`"riserBevel": "string"`

ベベルタイプを定義する文字列です。デフォルト値は undefined です。

- **棒グラフ**　次のいずれかの値を指定します。

`"bevel"`

ライザの両端に若干の傾斜を付けることで、ベベル効果が適用された外観にします。

`"cylinder"`

色のグラデーションを適用して円筒形のようなライザを描画します。

`"darken"`

縦ライザの場合は下向きに暗くなるグラデーション、横ライザの場合は左向きに暗くなるグラデーションをライザに適用します。

`"darkenInverted"`

縦ライザの場合は上向きに暗くなるグラデーション、横ライザの場合は右向きに暗くなるグラデーションをライザに適用します。

`"lighten"`

縦ライザの場合は下向きに明るくなるグラデーション、横ライザの場合は左向きに明るくなるグラデーションをライザに適用します。

`"lightenInverted"`

縦ライザの場合は上向きに明るくなるグラデーション、横ライザの場合は右向きに明るくなるグラデーションをライザに適用します。

- **バブルグラフおよび散布図**　次のいずれかの値を指定します。

`"darken"`

元のマーカー色から暗い色に変化する放射状グラデーションをマーカーに適用します。

`"lighten"`

元のマーカー色から明るい色に変化する放射状グラデーションをマーカーに適用します。

- **円グラフ**　次のいずれかの値を指定します。

  `"bevel"`

  円の外縁に角度を付けます。

  `"cylinder"`

  暗い色に変化するグラデーション効果を使用して、円を円筒形のような外観にします。

  `"donut"`

  円の中央に穴がある場合にドーナツ形状の円を描画します。

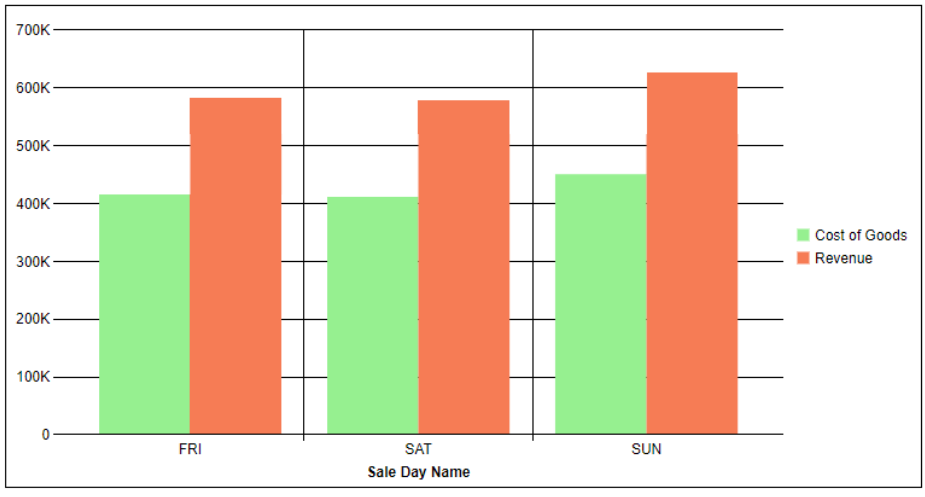
ベベルタイプの darken および lighten は、凡例マーカーにも適用されます (凡例マーカーが表示されている場合)。

例　棒グラフへのベベル効果の適用

次のリクエストは、棒グラフを生成し、ベベル効果を適用しません。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'FRI' OR 'SAT' OR 'SUN'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=N2, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N3, BUCKET=Y-AXIS,$
*GRAPH_JS
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"}
    ]
*END
ENDSTYLE
END
```

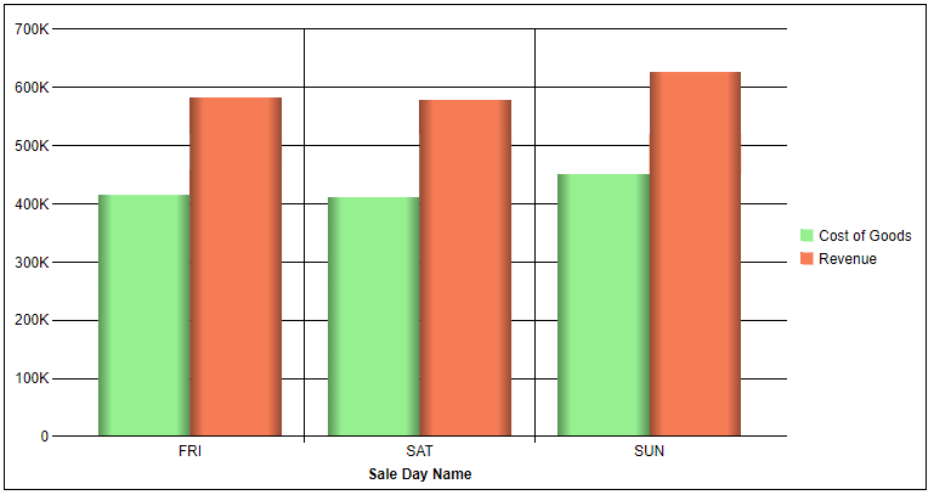
出力結果は次のとおりです。



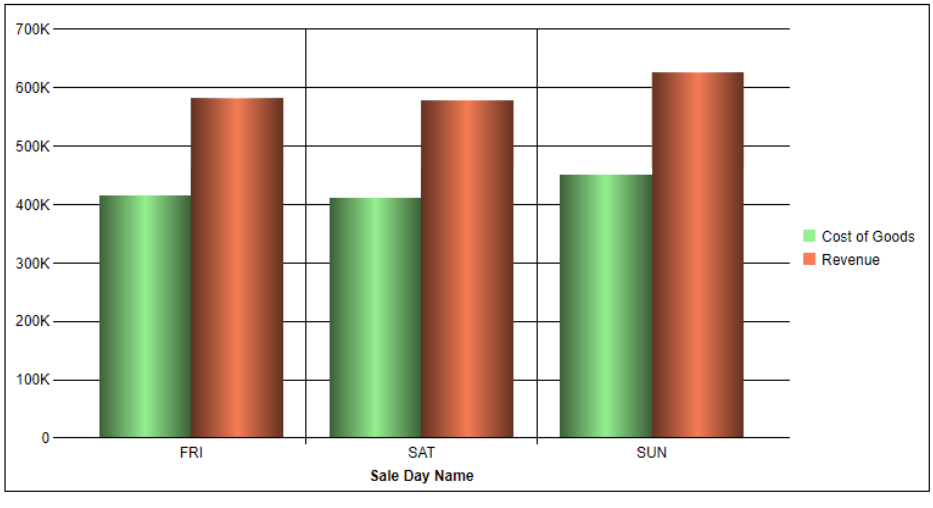
次のリクエストでは、ベベル効果を適用します ("riserBevel": "bevel")。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'FRI' OR 'SAT' OR 'SUN'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=N2, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N3, BUCKET=Y-AXIS,$
*GRAPH_JS
"riserBevel": "bevel",
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"}
    ]
*END
ENDSTYLE
END
```

出力結果では、ライザは完全な平面ではありません。ライザの両端に若干の傾斜が付いています。



このプロパティを "riserBevel": "cylinder" に変更すると、下図のグラフが生成され、色のグラデーションが適用された円筒形のようなライザが描画されます。

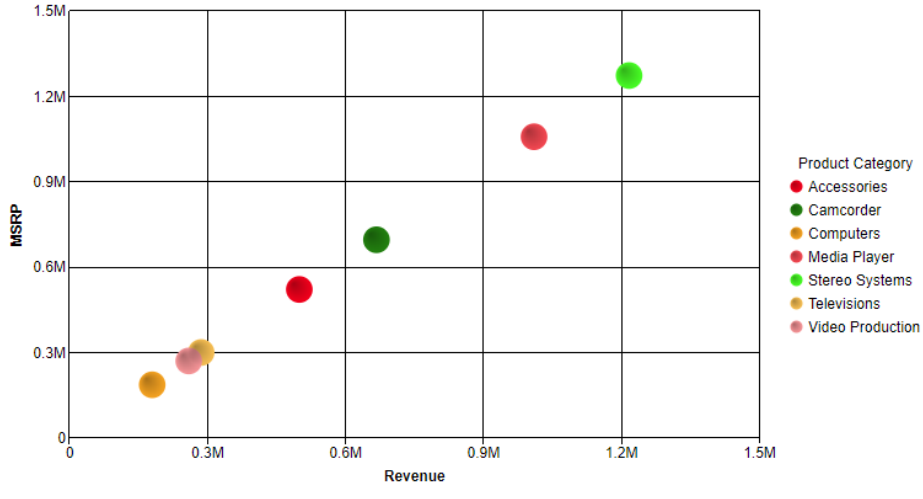


例　バブルグラフへのベベル効果の適用

次のリクエストでは、ベベル効果の "darken" をバブルマーカーおよび凡例マーカーに適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US MSRP_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 0},
"series": [{"series": "all",    "marker": {"shape": "circle"}}],
"riserBevel": "darken"
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



連続するライザへの濃淡効果の適用

グラフエンジンは、色が定義されたシリーズ数よりグラフに存在するシリーズ数が多い場合に、定義された色を残りのシリーズに周期的に割り当てます。riserCycleEndLightness プロパティは、定義された最後のライザ色を開始色として、残りのライザに周期的に割り当てる濃淡効果を定義します。

```
"riserCycleEndLightness": number
```

説明

```
riserCycleEndLightness: number
```

色が定義されていないシリーズライザに割り当てる色の明るさを 0 (ゼロ) から 1 までの数値で制御します。値を 0.5 より小さくすると、各シリーズの色が順に暗くなります。値を 0.5 より大きくすると、各シリーズの色が順に明るくなります。デフォルト値は 0.8 です。

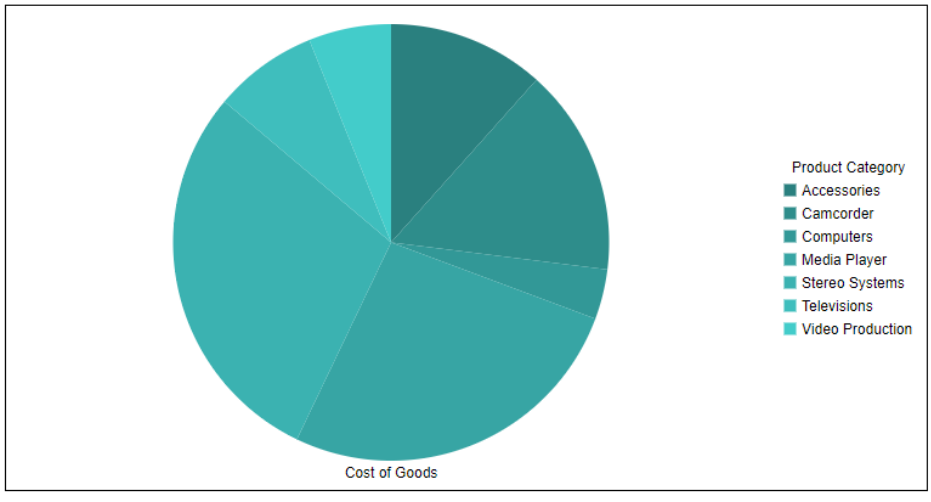
注意：WebFOCUS ではシリーズ色にデフォルト値が割り当てられるため、シリーズの配列に {"series": "reset", "color": undefined} を含めることで、これらのデフォルト値を削除する必要があります。次に、周期的な色の割り当ての開始色を定義します。

例　連続する円項目への濃淡効果の適用

次のリクエストは、シリーズ 0 の色を青緑色に設定し、この色を開始色として後続の円項目に濃淡効果を適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN = N1, BUCKET=COLOR,$
TYPE=DATA, COLUMN=N2, BUCKET=MEASURE,$
*GRAPH_JS
"riserCycleEndLightness": 0.4,
"series": [
   {"series": "reset", "color": undefined},
   {"series": 0, "color": "teal"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



3D グラフのライザ間隔の制御

3D グラフおよび 2.5D グラフ (depth プロパティで奥行きが適用されたグラフ) では、riserDepthGap プロパティを使用して、手前のライザと背後のライザとの間に隙間を追加することができます。

構文　3D グラフのライザ間隔の制御

```
"riserDepthGap": number
```

説明

`"riserDepthGap": number`

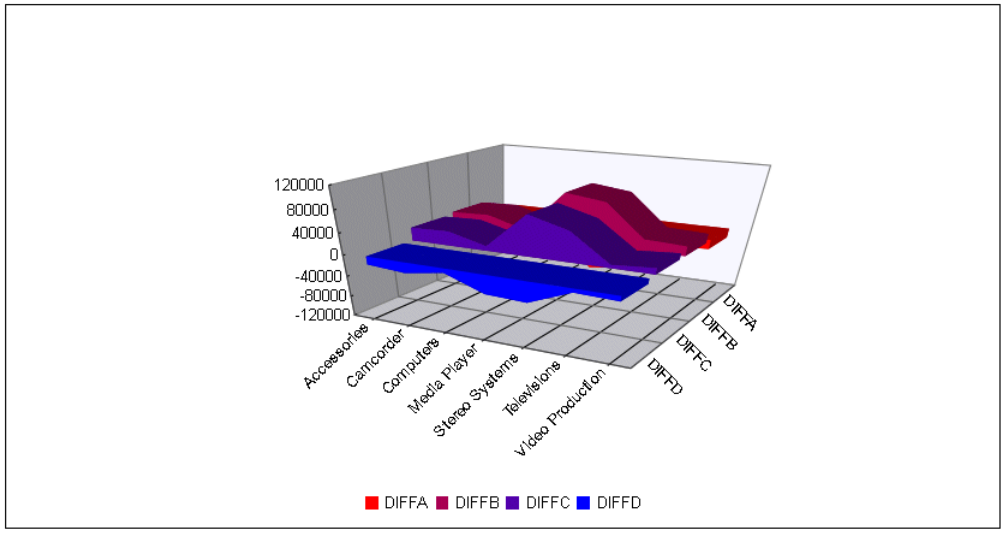
ライザ間の間隔を奥行きの倍数で定義する 0 (ゼロ) から 1 までの数値です。デフォルト値は 0.2 です。

例　3D グラフのライザ間隔の制御

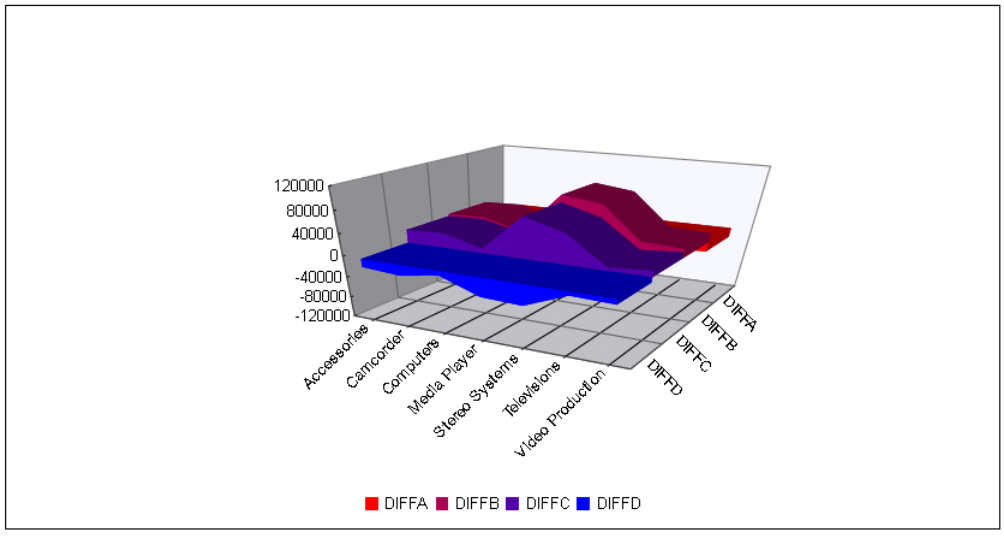
次のリクエストは、3D 面グラフを生成し、ライザ間の間隔にデフォルト値を使用します。

```
DEFINE FILE WF_RETAIL_LITE
DIFFA = GROSS_PROFIT_US - REVENUE_US;
DIFFB = REVENUE_US - GROSS_PROFIT_US;
DIFFC = COGS_US - (COGS_US * DISCOUNT_US)/100;
DIFFD = COGS_US - MSRP_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFFA DIFFB DIFFC DIFFD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH 3DAREAS
ON GRAPH SET STYLE *
*GRAPH_JS
"riserDepthGap": 0.2,
"chartFrame": {
    "border": {"width": 1, "color": "grey"},
    "fill": {"color": "ghostwhite"}},
"colorMode": {
    "mode": "byInterpolation",
    "colorList": ["red", "blue"]
    }
*END
ENDSTYLE
END
```

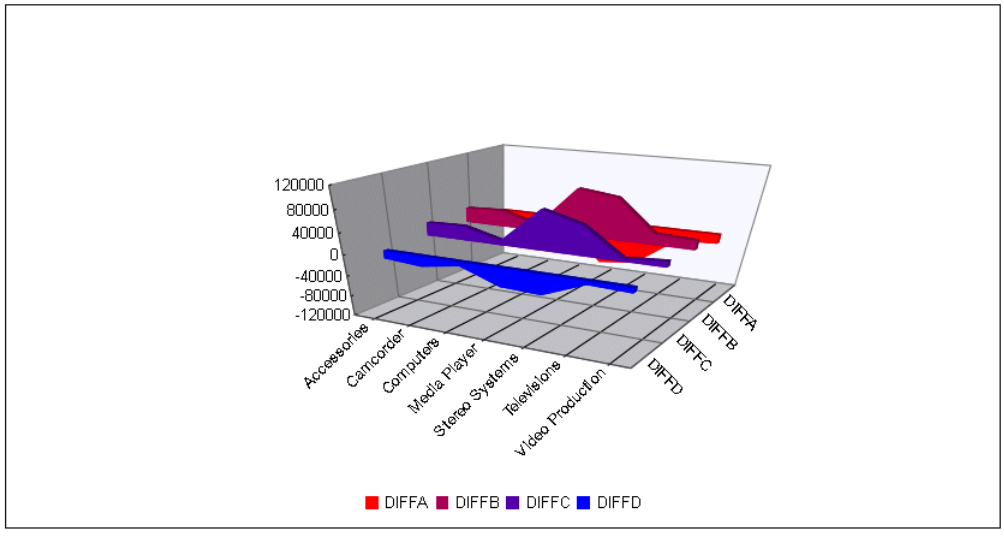
出力結果は次のとおりです。



riserDepthGap プロパティを 0 (ゼロ) に変更すると、下図のグラフが生成されます。



riserDepthGap プロパティ値を 0.9 に変更すると、下図のグラフが生成されます。



シリーズとグループの位置の入れ替え

swapData プロパティを使用して、シリーズとグループの位置を入れ替えることができます。false (デフォルト) の場合、データセットの行の値がシリーズとして表示されます。true の場合、データセットの列の値がシリーズとして表示されます。

構文　シリーズとグループの位置の入れ替え

```
"swapData": boolean
```

説明

`"swapData": boolean`

次のいずれかの値を指定します。

- true - シリーズとグループの位置を入れ替えます。
- false - シリーズとグループの位置を入れ替えません。これがデフォルト値です。

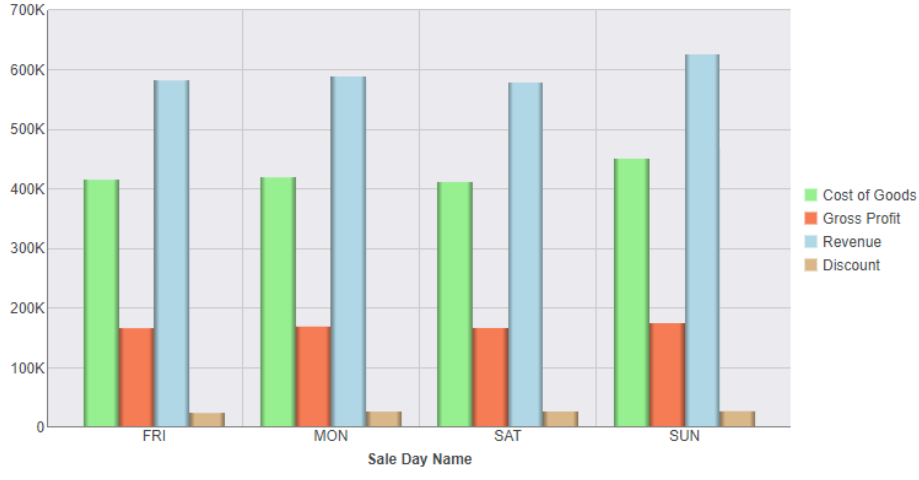
例　シリーズとグループの位置の入れ替え

次のリクエストは、縦棒グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'FRI' OR 'SAT' OR 'SUN' OR 'MON'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=N2, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N3, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N4, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N5, BUCKET=Y-AXIS,$
*GRAPH_JS
"swapData": false,
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"},
   {"series": 2, "color": "lightblue"},
   {"series": 3, "color": "burlywood"}
   ]
*END
ENDSTYLE
END
```

下図は、出力結果を示しています。

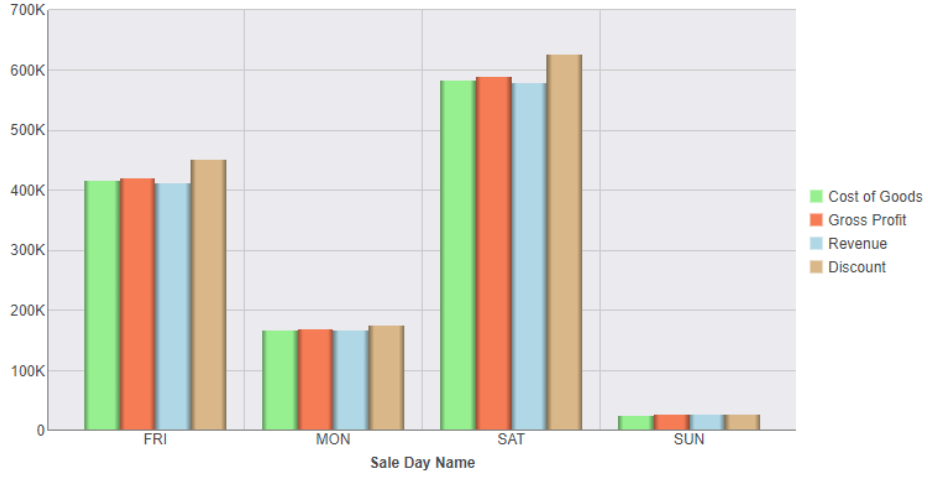
- 棒の第 1 セットは、第 1 グループ値 (曜日 FRI) のすべてのシリーズ値 (売上原価、粗利益、収益、値引) を表します。
- 棒の第 2 セットは、第 2 グループ値 (曜日 MON) のすべてのシリーズ値を表します。
- 棒の第 3 セットは、第 3 グループ値 (曜日 SAT) のすべてのシリーズ値を表します。
- 棒の第 4 セットは、第 4 グループ値 (曜日 SUN) のすべてのシリーズ値を表します。



swapData プロパティを true に変更すると、下図のグラフが生成されます。

- 棒の第 1 セットは、各曜日 (グループ値) の売上原価 (COGS_US) (シリーズ 0) を表します。
- 棒の第 2 セットは、各曜日 (グループ値) の粗利益 (GROSS_PROFIT_US) (シリーズ 1) を表します。
- 棒の第 3 セットは、各曜日 (グループ値) の収益 (REVENUE_US) (シリーズ 2) を表します。
- 棒の第 4 セットは、各曜日 (グループ値) の値引 (DISCOUNT_US) (シリーズ 3) を表します。

ラベルは変更されていませんが、シリーズとグループの位置が入れ替わっています。



シリーズとグループ以外にラベルの位置を入れ替える場合は、246 ページの 「 シリーズ、グループ、ラベルの位置の入れ替え 」 を参照してください。

シリーズ、グループ、ラベルの位置の入れ替え

swapDataAndLabels プロパティを使用して、シリーズとグループの位置を入れ替える以外に、ラベルの位置を入れ替えることができます。false (デフォルト) の場合、データセットの行の値がシリーズとして表示されます。このプロパティは swapData プロパティに類似していますが、このプロパティを使用すると、シリーズとグループのラベルの位置も入れ替えられます。true の場合、データセットの列の値がシリーズとして表示されます。

構文　シリーズ、グループ、ラベルの位置の入れ替え

```
"swapDataAndLabels": boolean
```

説明

`"swapDataAndLabels": boolean`

次のいずれかの値を指定します。

true - シリーズ、グループ、ラベルの位置を入れ替えます。

false - シリーズ、グループ、ラベルの位置を入れ替えません。これがデフォルト値です。

例　シリーズ、グループ、ラベルの位置の入れ替え

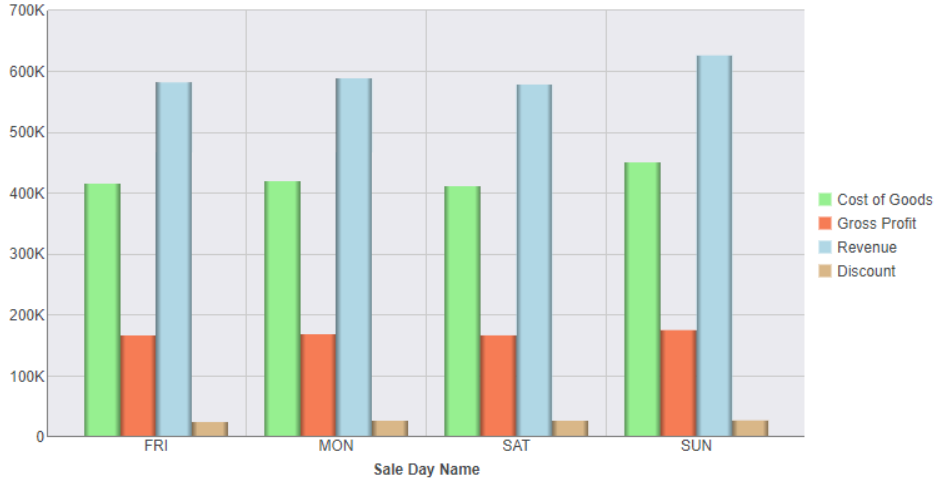
次のリクエストは、縦棒グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US
BY TIME_DAYNAME
WHERE TIME_DAYNAME EQ 'FRI' OR 'SAT' OR 'SUN' OR 'MON'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=X-AXIS,$
TYPE=DATA, COLUMN=N2, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N3, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N4, BUCKET=Y-AXIS,$
TYPE=DATA, COLUMN=N5, BUCKET=Y-AXIS,$
*GRAPH_JS
"swapDataAndLabels": false,
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"},
   {"series": 2, "color": "lightblue"},
   {"series": 3, "color": "burlywood"}
   ]
*END
ENDSTYLE
END
```

下図は、出力結果を示しています。

- 棒の第 1 セットは、第 1 グループ値 (曜日 FRI) のすべてのシリーズ値 (売上原価、粗利益、収益、値引) を表します。
- 棒の第 2 セットは、第 2 グループ値 (曜日 MON) のすべてのシリーズ値を表します。
- 棒の第 3 セットは、第 3 グループ値 (曜日 SAT) のすべてのシリーズ値を表します。

- 棒の第 4 セットは、第 4 グループ値 (曜日 SUN) のすべてのシリーズ値を表します。



swapDataAndLabels プロパティを true に変更すると、下図のグラフが生成されます。

- 棒の第 1 セットは、各曜日 (グループ値) の売上原価 (COGS_US) (シリーズ 0) を表します。
- 棒の第 2 セットは、各曜日 (グループ値) の粗利益 (GROSS_PROFIT_US) (シリーズ 1) を表します。
- 棒の第 3 セットは、各曜日 (グループ値) の収益 (REVENUE_US) (シリーズ 2) を表します。
- 棒の第 4 セットは、各曜日 (グループ値) の値引 (DISCOUNT_US) (シリーズ 3) を表します。

ラベルの位置も入れ替えられ、ラベルが新しい位置に表示されています。

6 グラフタイトルのプロパティ

ここでは、グラフのタイトル、サブタイトル、脚注のフォーマットを制御する方法について説明します。

トピックス

- グラフタイトルのプロパティの概要
- グラフ脚注のフォーマット設定
- グラフサブタイトルのフォーマット設定
- グラフタイトルのフォーマット設定

グラフタイトルのプロパティの概要

ここでは、グラフのタイトル、サブタイトル、脚注の表示およびフォーマットを制御するプロパティについて説明します。次のコードは、プロパティ名およびデフォルト値を示しています。

```
"title": {
   "text": "Chart Title",
   "visible": false,
   "align": "center",
   "font": "bold 10pt Sans-Serif",
   "color": "black",
   "backgroundColor": undefined,
      "border": {
                 "width": 0,
                 "color": undefined,
                 "dash": "",
                 "cornerRadius": {"x": 0}
                }

},
```

```
"subtitle": {
   "text": "Chart Subtitle",
   "visible": false,
   "align": "center",
   "font": "10pt Sans-Serif",
   "color": "black",
   "backgroundColor": undefined,
      "border": {
                "width": 0,
                "color": undefined,
                "dash": "",
                "cornerRadius": {"x": 0}
               }

},

"footnote": {
   "text": "Chart Footnote",
   "visible": false,
   "align": "center",
   "font": "10pt Sans-Serif",
   "color": "black",
   "backgroundColor": undefined,
      "border": {
                "width": 0,
                "color": undefined,
                "dash": "",
                "cornerRadius": {"x": 0}
               }

 },
```

グラフの各軸にもタイトルオブジェクトがあります。詳細は、315 ページの 「 軸のプロパティ 」 を参照してください。

グラフ脚注のフォーマット設定

footnote プロパティは、グラフ脚注のテキスト、表示、フォーマットを制御します。

構文　グラフ脚注のフォーマット設定

```
"footnote": {
   "text": "string",
   "visible": boolean,
   "align": "string",
   "font": "string",
   "color": "string"
   "backgroundColor": "string",
   "border": {
               "width": number,
               "color": "string",
               "dash": "string",
               "cornerRadius": {"x": value, "y": value}
              }
}
```

説明

`"text": "string"`

脚注のテキストを定義する文字列です。デフォルト値は "Chart Footnote" です。

`"visible": boolean`

グラフ脚注の表示を制御します。有効な値には、次のものがあります。

- true - グラフ脚注を表示します。
- false - グラフ脚注を表示しません。これがデフォルト値です。

`"align": "string"`

脚注の位置揃えを定義する文字列です。有効な値には、次のものがあります。

- "center" - 描画領域内で脚注を中央揃えにします。これがデフォルト値です。
- "chartFrame" - グラフフレームを基準に脚注を中央揃えにします。
- "left" - 描画領域内で脚注を左揃えにします。
- "right" - 描画領域内で脚注を右揃えにします。

`"font": "string"`

グラフ脚注のサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列でグラフ脚注の色を定義します。デフォルト値は "black" です。

色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"backgroundColor": "string"`
グラフタイトルオブジェクトの背景塗りつぶし色を定義する色指定の文字列、またはグラデーションです。デフォルト値は undefined です。

`"border"`
グラフタイトルオブジェクト周囲の境界のプロパティを定義します。

`"width": number`
境界の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

`"color": "string"`
グラフタイトルオブジェクト周囲の境界色を定義する色指定の文字列です。デフォルト値は undefined です。

`"dash": "string"`
グラフタイトルオブジェクト周囲の境界線の破線スタイルを定義する文字列です。破線長さと破線間隔をピクセル単位で定義する数値を列記します。デフォルト値は破線なし (" ") です。

`"cornerRadius": {"x": value, "y": value}`
タイトルオブジェクト周囲のボックスの角形状プロパティを定義します。

cornerRadius プロパティには、1 つまたは 2 つの値を指定することができます。値の 1 つが 0 (ゼロ) の場合、角形状は四角になります。これがデフォルト角形状です。曲線の角形状は、円形または楕円形にすることができます。x および y 値は、円形の半径、または楕円形の長半径と短半径を表します。

x および y の有効値は次のとおりです。

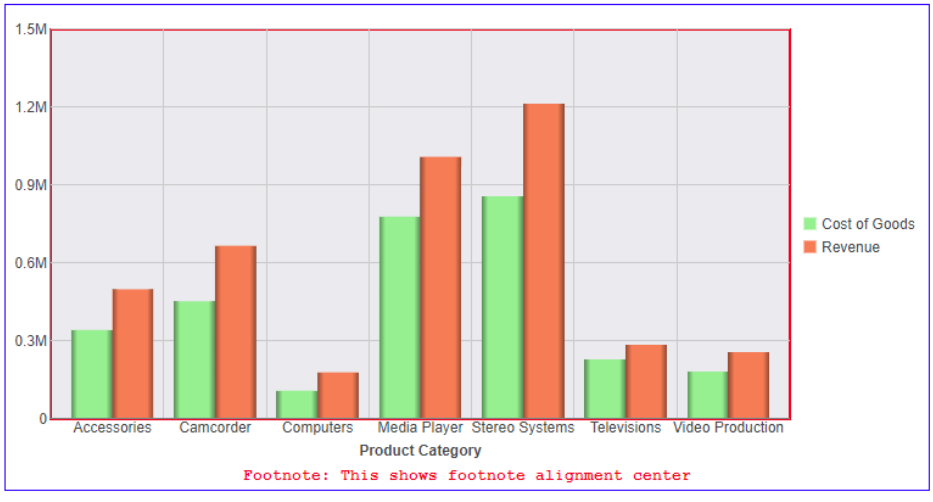
- 数値 - 角形状の半径をピクセル単位で指定します。
- パーセント文字列 (例、"20%") - タイトルオブジェクト周囲のボックスの高さ (y) または幅 (x) を表すパーセントを指定します。
- CSS 長さ文字列 (例、"5em"、"10px") - 角形状の半径をピクセル単位で指定します。CSS 長さ文字列についての詳細は、https://developer.mozilla.org/en-US/docs/Web/CSS/length を参照してください。
- "x" および "y" プロパティで定義されるオブジェクト (ここで "x" および "y" には上記のいずれかを指定可能) - タイトルオブジェクトボックスの幅 (x) および高さ (y) に基づいて角形状の半径を指定します。

例　グラフ脚注のフォーマット設定

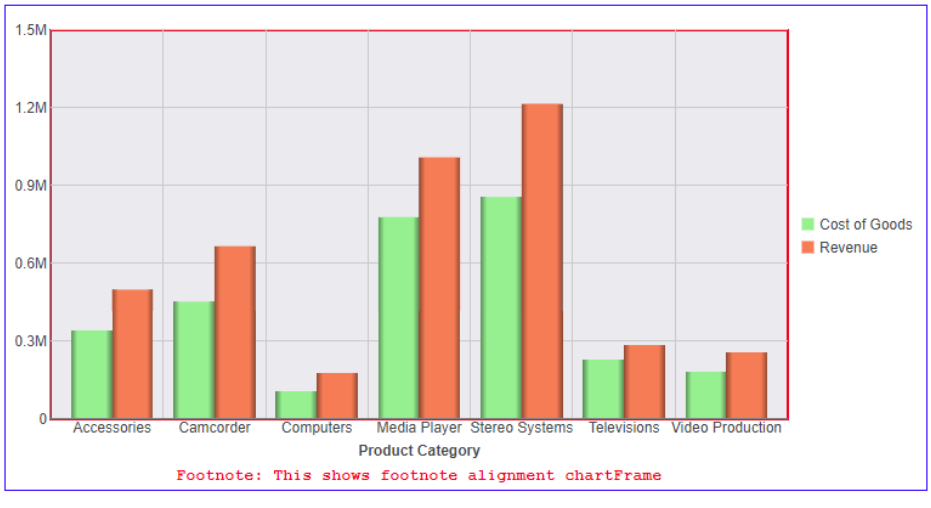
次のリクエストは、縦棒グラフを生成し、グラフ脚注を表示します。このリクエストでは、&FOOTALIGN と呼ばれるダイアログマネージャ変数が使用され、その値が 'center' に設定されています。脚注の位置揃え、およびテキストの一部は、この &FOOTALIGN 変数によって制御されます。

```
-SET &FOOTALIGN = 'center';
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"border": {"color": "blue"},
"chartFrame": {"border": {
   "width": 2, "color": "red"}},
"legend": {"position": "right"},
"footnote": {
   "text": "Footnote: This shows footnote alignment &FOOTALIGN",
   "font": "Bold 10pt Courier", "visible": true,
   "align": "&FOOTALIGN","color": "red"},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"},   ]
*END
ENDSTYLE
END
```

位置揃えが 'center' に設定されているため、出力結果では、脚注が描画領域の中央に揃えられています。



&FOOTALIGN の値を 'chartFrame' に変更すると、脚注がグラフフレームの中央に揃えられます。



グラフサブタイトルのフォーマット設定

subtitle プロパティは、グラフサブタイトルのテキスト、表示、フォーマットを制御します。

構文　グラフサブタイトルのフォーマット設定

```
"subtitle": {
   "text": "string",
   "visible": boolean,
   "align": "string",
   "font": "string",
   "color": "string"
"backgroundColor": "string",
   "border": {
      "width": number,
      "color": "string",
      "dash": "string",
      "cornerRadius": {"x": value, "y": value}
               }

   }
```

説明

`"text": "string"`

サブタイトルのテキストを定義する文字列です。デフォルト値は "Subtitle" です。

`"visible": boolean`

グラフサブタイトルの表示を制御します。有効な値には、次のものがあります。

- true - グラフサブタイトルを表示します。
- false - グラフサブタイトルを表示しません。これがデフォルト値です。

`"align": "string"`

サブタイトルの位置揃えを定義する文字列です。有効な値には、次のものがあります。

- "center" - 描画領域内でサブタイトルを中央揃えにします。これがデフォルト値です。
- "chartFrame" - グラフフレームを基準にサブタイトルを中央揃えにします。
- "left" - 描画領域内でサブタイトルを左揃えにします。
- "right" - 描画領域内でサブタイトルを右揃えにします。

`"font": "string"`

グラフサブタイトルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列でグラフサブタイトルの色を定義します。デフォルト値は "black" です。

色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"backgroundColor": "string"`
グラフタイトルオブジェクトの背景塗りつぶし色を定義する色指定の文字列、またはグラデーションです。デフォルト値は undefined です。

`"border"`
グラフタイトルオブジェクト周囲の境界のプロパティを定義します。

`"width": number`
境界の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

`"color": "string"`
グラフタイトルオブジェクト周囲の境界色を定義する色指定の文字列です。デフォルト値は undefined です。

`"dash": "string"`
グラフタイトルオブジェクト周囲の境界線の破線スタイルを定義する文字列です。破線長さと破線間隔をピクセル単位で定義する数値を列記します。デフォルト値は破線なし (" ") です。

`"cornerRadius": {"x": value, "y": value}`
タイトルオブジェクト周囲のボックスの角形状プロパティを定義します。

cornerRadius プロパティには、1 つまたは 2 つの値を指定することができます。値の 1 つが 0 (ゼロ) の場合、角形状は四角になります。これがデフォルト角形状です。曲線の角形状は、円形または楕円形にすることができます。x および y 値は、円形の半径、または楕円形の長半径と短半径を表します。

x および y の有効値は次のとおりです。

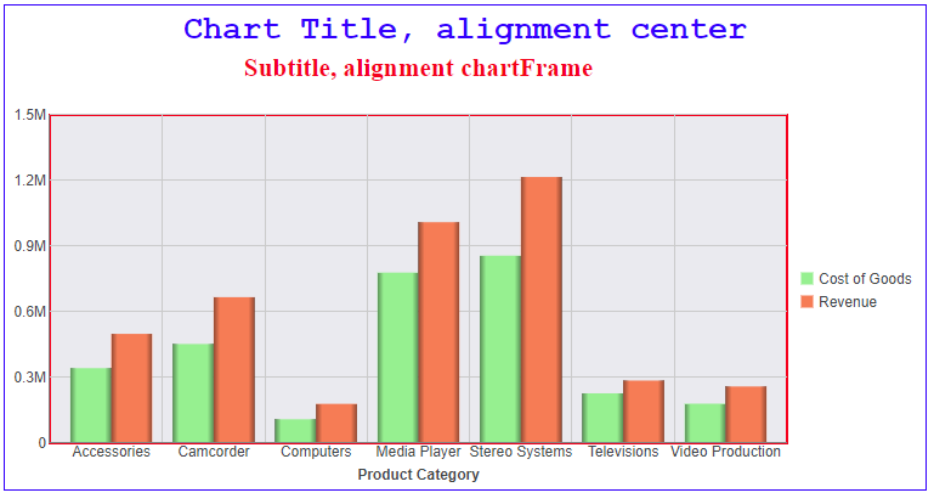
- 数値 - 角形状の半径をピクセル単位で指定します。
- パーセント文字列 (例、"20%") - タイトルオブジェクト周囲のボックスの高さ (y) または幅 (x) を表すパーセントを指定します。
- CSS 長さ文字列 (例、"5em"、"10px") - 角形状の半径をピクセル単位で指定します。CSS 長さ文字列についての詳細は、https://developer.mozilla.org/en-US/docs/Web/CSS/length を参照してください。
- "x" および "y" プロパティで定義されるオブジェクト (ここで "x" および "y" には上記のいずれかを指定可能) - タイトルオブジェクトボックスの幅 (x) および高さ (y) に基づいて角形状の半径を指定します。

例　グラフサブタイトルのフォーマット設定

次のリクエストは、縦棒グラフを生成し、グラフのサブタイトルを表示します。このリクエストでは、&SUBTALIGN と呼ばれるダイアログマネージャ変数が使用され、その値が 'chartFrame' に設定されています。サブタイトルの位置揃え、およびテキストの一部は、この &SUBTALIGN 変数によって制御されます。

```
-SET &SUBALIGN = 'chartFrame';
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"border": {"color": "blue"},
"chartFrame": {"border": {
   "width": 2, "color": "red"}},
"legend": {"position": "right"},
"title": {
   "text": "Chart Title, alignment center",
   "visible": true,
   "font": "Bold 20pt Courier",
   "align": "center", "color": "blue"},
"subtitle": {
   "text": "Subtitle, alignment &SUBALIGN",
   "font": "Bold 16pt Comic Sans MS", "visible": true,
   "align": "&SUBALIGN", "color": "red"},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}]
*END
ENDSTYLE
END
```

出力結果では、タイトルが描画領域の中央に揃えられ ("align": "center")、サブタイトルがグラフフレームの中央に揃えられています ("align": "chartFrame")。



グラフタイトルのフォーマット設定

title プロパティは、グラフタイトルのテキスト、表示、フォーマットを制御します。

注意：スパークライングラフでは、グラフタイトルはサポートされません。また、ブレットグラフでは、グラフタイトルの位置揃えを変更することはできません。

構文　　グラフタイトルのフォーマット設定

```
"title": {
   "text": "string",
   "visible": boolean,
   "align": "string",
   "font": "string",
   "color": "string"
"backgroundColor": "string",
 "border": {
    "width": number,
    "color": "string",
    "dash": "string",
    "cornerRadius": {"x": value, "y": value}
          }
  }
```

説明

`"text": "string"`

タイトルのテキストを定義する文字列です。デフォルト値は "Title" です。

`"visible": boolean`

グラフタイトルの表示を制御します。有効な値には、次のものがあります。

- true - グラフタイトルを表示します。
- false - グラフタイトルを表示しません。これがデフォルト値です。

`"align": "string"`

タイトルの位置揃えを定義する文字列です。有効な値には、次のものがあります。

- "center" - 描画領域内でタイトルを中央揃えにします。これがデフォルト値です。
- "chartFrame" - グラフフレームを基準にタイトルを中央揃えにします。
- "left" - 描画領域内でタイトルを左揃えにします。
- "right" - 描画領域内でタイトルを右揃えにします。

`"font": "string"`

グラフタイトルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列でグラフタイトルの色を定義します。デフォルト値は "black" です。

色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"backgroundColor": "string"`

グラフタイトルオブジェクトの背景塗りつぶし色を定義する色指定の文字列、またはグラデーションです。デフォルト値は undefined です。

`"border"`

グラフタイトルオブジェクト周囲の境界のプロパティを定義します。

`"width": number`

境界の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

`"color": "string"`
グラフタイトルオブジェクト周囲の境界色を定義する色指定の文字列です。デフォルト値は undefined です。

`"dash": "string"`
グラフタイトルオブジェクト周囲の境界線の破線スタイルを定義する文字列です。破線長さと破線間隔をピクセル単位で定義する数値を列記します。デフォルト値は破線なし (" ") です。

`"cornerRadius": {"x": value, "y": value}`
タイトルオブジェクト周囲のボックスの角形状プロパティを定義します。

cornerRadius プロパティには、1 つまたは 2 つの値を指定することができます。値の 1 つが 0 (ゼロ) の場合、角形状は四角になります。これがデフォルト角形状です。曲線の角形状は、円形または楕円形にすることができます。x および y 値は、円形の半径、または楕円形の長半径と短半径を表します。

x および y の有効値は次のとおりです。

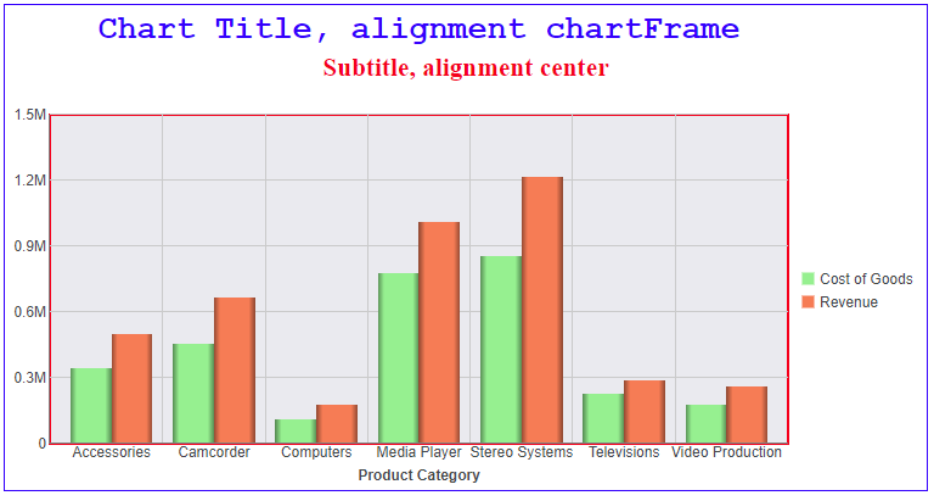
- 数値 - 角形状の半径をピクセル単位で指定します。
- パーセント文字列 (例、"20%") - タイトルオブジェクト周囲のボックスの高さ (y) または幅 (x) を表すパーセントを指定します。
- CSS 長さ文字列 (例、"5em"、"10px") - 角形状の半径をピクセル単位で指定します。CSS 長さ文字列についての詳細は、https://developer.mozilla.org/en-US/docs/Web/CSS/length を参照してください。
- "x" および "y" プロパティで定義されるオブジェクト (ここで "x" および "y" には上記のいずれかを指定可能) - タイトルオブジェクトボックスの幅 (x) および高さ (y) に基づいて角形状の半径を指定します。

例　グラフタイトルのフォーマット設定

次のリクエストは、縦棒グラフを生成し、グラフのタイトルとサブタイトルを表示します。タイトルの位置揃えは "chartFrame"、サブタイトルの位置揃えは "center" です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
border:{color:'blue'},
chartFrame: {border: {width:2, color:'red'}},
legend: {position:'right'},
 "border": {"color": "blue"},
"chartFrame": {"border": {
   "width": 2, "color": "red"}},
"legend": {"position": "right"},
"title": {
   "text": "Chart Title, alignment chartFrame",
   "visible": true, "font": "Bold 20pt Courier",
   "align": "chartFrame", "color": "blue"},
"subtitle": {
   "text": "Subtitle, alignment center",
   "font": "Bold 16pt Comic Sans MS", "visible": true,
   "align": "center", "color": "red"},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果では、タイトルがグラフフレームの中央に揃えられ ("align": "chartFrame")、サブタイトルが描画領域の中央に揃えられています ("align": "center")。

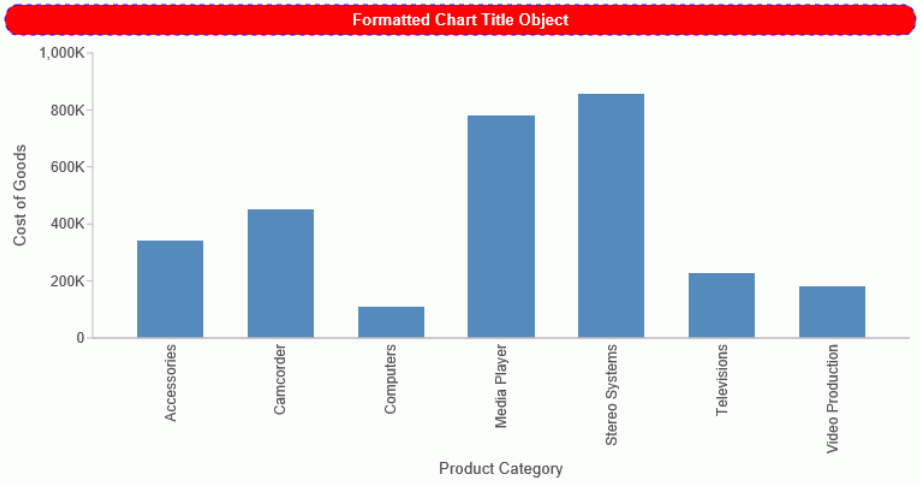


例　グラフタイトルの背景塗りつぶしと境界プロパティのフォーマット設定

次のリクエストは、タイトルボックスを生成し、背景塗りつぶしを赤色、境界を青色の破線、角形状を円形に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
*GRAPH_JS
"title": {
      "visible": true,
      "text": "Formatted Chart Title Object",
      "color": "white",
      "backgroundColor": "red",
      "border":
              {
               "width": 1,
               "color": "blue",
               "dash": "4 4",
               "cornerRadius": {"x": 10, "y":"25%"}
               }
         }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

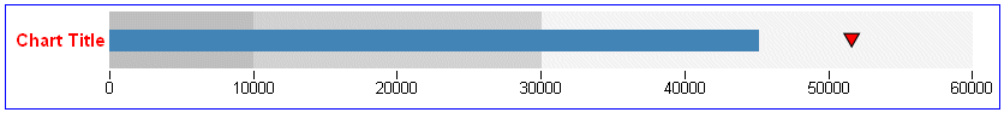


例　ブレットグラフのグラフタイトルのフォーマット設定

次のリクエストは、ブレットグラフを生成し、グラフタイトルを表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY TIME_DAYOFWEEK
WHERE TIME_DAYNAME EQ 'SAT' OR 'SUN'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VAXIS 80
ON GRAPH SET LOOKGRAPH CUSTOM
ON GRAPH SET STYLE *
*GRAPH_JS
"chartType": "bullet",
"border": {"width": 1, "color": "blue"},
"title": {"visible": true, "text": "Chart Title", "color": "red"},
"yaxis": {"colorBands": [
    {"start": 0, "stop": 10000, "color": "silver"},
    {"start": 10000, "stop": 30000, "color": "lightgrey"},
    {"start": 30000, "stop": 60000, "color": "whitesmoke"}]},
"series": [
    {"series": 0, "group": 0, "color": "steelblue"},
    {"series": 0, "group": 1, "color": "red", "marker":{"shape":"triangle"}}
    ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



7

凡例のプロパティ

ここでは、凡例領域のフォーマットを制御する方法について説明します。

トピックス

- 凡例のプロパティの概要
- 凡例領域の背景色の定義
- 凡例ラベルのフォーマット設定
- 円グラフ凡例ラベルでのグラフマクロの使用
- 凡例領域の枠線の生成
- 凡例マーカーの位置の制御
- 凡例マーカーのサイズの制御
- 折れ線グラフの凡例マーカーとシリーズマーカーの一致
- 凡例領域の最大パーセントの設定
- 凡例の方向の制御
- 凡例の位置の制御
- 凡例項目の順序の入れ替え
- スクロール凡例の生成
- ドッキング凡例の生成
- 凡例タイトルのフォーマット設定
- 凡例領域の表示の制御
- サイズ凡例のフォーマット設定

凡例のプロパティの概要

凡例のプロパティは、凡例領域の表示およびフォーマットを制御します。

次のコードセグメントは、デフォルト値を示しています。

```
"legend": {
    "visible": true,
    "position": "right",
  // {
  //   "side": undefined,
  //   "anchor": "chartFrame",
  //   "excludeFromLayout": undefined
  //   "left": undefined,
  //   "right": undefined
  //   "top": undefined,
  //   "bottom": undefined,
  //   "align": undefined
  // },
 "maxSize": '50%',
 "orientation": "auto",
 "reverseOrder": false,
 "markerSize": 8,
 "markerPosition": "left",
 "matchSeriesMarkers": false,
 "maxEntries": "undefined",
 "title": {
          "visible": false,
          "text": "Legend Title",
          "font": "10pt Sans-Serif",
          "color": "black",
    "tooltip": undefined,
    "wrap": 1},
 "labels": {
    "content": undefined,
    "font": "7.5pt Sans-Serif",
    "color": "black",
    "tooltip": undefined,
    "wrap": 1},
 "lineStyle": {
          "width": 0,
          "color": "black",
          "dash": ""},
 "backgroundcolor": "transparent",

    "scroll": {
        "enabled": false,
        "size": 15,
        "color": "rgb(240, 240, 240)",
        "handle": {
            "color": "grey",
            "hoverColor": "rgb(88, 88, 88)",
            "border": {
                "width": 0,
                "color": "transparent",
                "dash": ""
        }}},
```

```
    "dock": {
        "enabled": false,
        "animate": true,
        "minimized": false,
        "resizeFrame": true,
        "showMarkers": true,
        "expandDirection": "undefined",
        "button": {
            "size": "auto",
            "color": "grey",
            "hoverColor": "black",
            "border": {
                "width": 0,
                "color": "transparent",
                "dash": ""
        }}},

"sizeLegend":
 {
    "layout": "circle",
    "lineStyle":
     {
      "width": 0,
      "color": "black",
      "dash": ""
      }
  }

}
```

凡例領域の背景色の定義

backgroundcolor プロパティは、凡例領域の背景色を定義します。

構文　凡例領域の背景色の定義

```
"legend": {
   "backgroundcolor": "color"}
```

説明

`"backgroundcolor": "color"`

次のいずれかの値を指定します。

- 色名または数値指定で色を定義する文字列、あるいはグラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "transparent" です。
- グラデーションを定義する JSON オブジェクト。

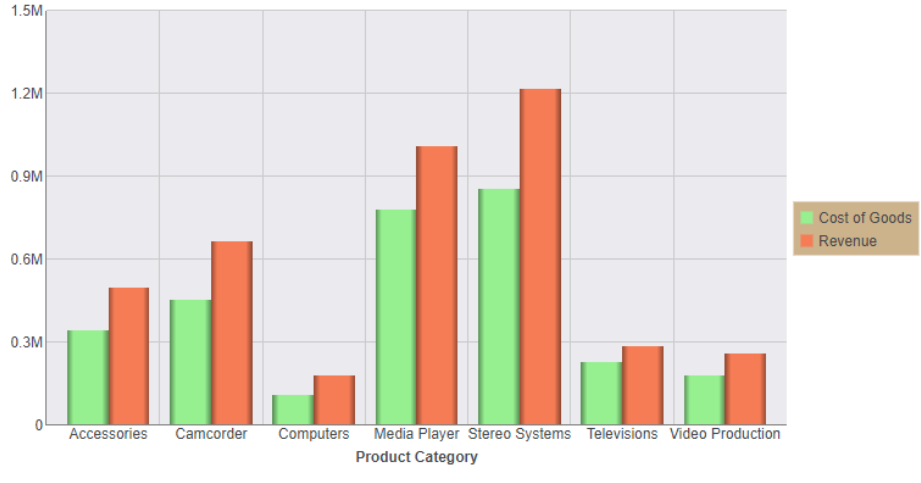
色およびグラデーションの定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　　凡例領域の背景色の定義

次のリクエストは、凡例領域の背景色を黄褐色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"backgroundcolor": "tan"},
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"}
    ]
*END
ENDSTYLE
END
```

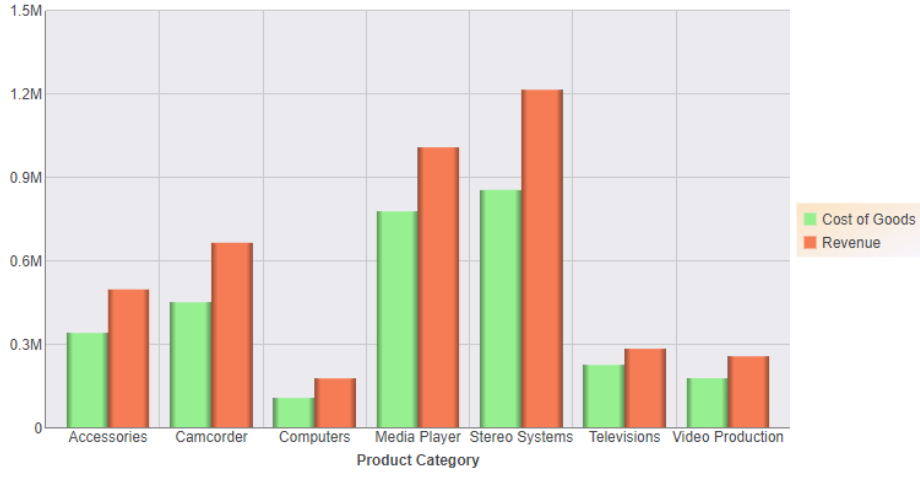
出力結果は次のとおりです。



次のリクエストは、凡例領域にビスクからゴーストホワイトに変化する線状グラデーションを適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {
   "position": "right",
   "backgroundcolor": {
      "type": "linear",
      "start": {"x": "0%", "y": "0%"},
      "end": {"x": "100%", "y": "100%"},
      "stops": [[0, "bisque"], [1, "ghostwhite"]],
      }},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



凡例ラベルのフォーマット設定

labels プロパティは、凡例ラベルのフォーマットを制御します。

構文　凡例ラベルのフォーマット設定

```
"legend":
 {
   "labels":
    {
      "font": "string",
      "color": "string"
      "tooltip": "string",
      "wrap": number
    }
 }
```

説明

`"font": "string"`

凡例ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "7.5pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列で凡例ラベルの色を定義します。デフォルト値は "black" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"tooltip": "string"`

マウスポインタをラベル上に置いたときに表示する文字列、または文字列を返す関数を指定します。

`"wrap": number`

長いラベルの折り返し行数を定義します。整数を指定します。必要に応じて、長いラベルをこの行数に折り返すか、"auto" を指定して、長いラベルを必要な行数に折り返します。デフォルト値は 1 です。

例　　凡例ラベルのフォーマット設定

次のリクエストは、凡例ラベルを赤色、太字、12pt Times Roman フォントに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"position": "bottom",
  "labels": {
   "font": "Bold 12pt Times Roman",
   "color": "red"
   }},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

円グラフ凡例ラベルでのグラフマクロの使用

グラフエンジンマクロを使用して、円グラフの凡例ラベルをカスタマイズすることができます。たとえば、データラベルに表示されるシリーズのパーセント値および値の情報を、円グラフの凡例ラベルに表示することができます。

構文　円グラフ凡例ラベルでのグラフマクロの使用

テキストおよびマクロでカスタマイズするには、content プロパティを凡例ラベルに追加します。

```
"legend": {
        "labels": {
            "content": "string"
        }
    }
```

説明

`"content": "string"`

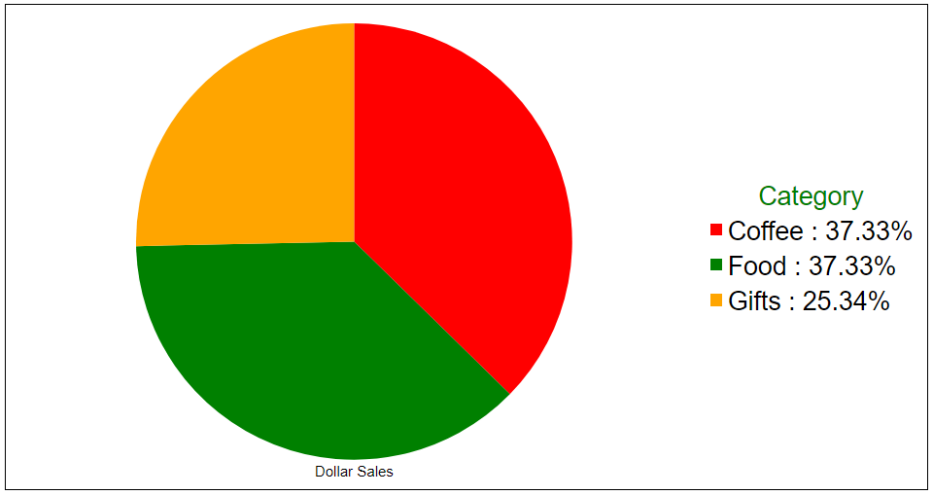
ラベルのカスタマイズを定義する、オプションのテキストおよびマクロ参照です。グラフエンジンマクロについての詳細は、109 ページの 「 グラフテンプレートエンジンを使用したツールヒント、タイトル、データラベルのカスタマイズ 」 を参照してください。

例　円グラフ凡例へのシリーズラベルおよびシリーズパーセント値の追加

次のリクエストは、円グラフの凡例ラベルに {{series_label}} マクロおよび {{series_percent}} マクロを挿入します。

```
GRAPH FILE ibisamp/ggsales
SUM DOLLARS
BY CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=color, $
TYPE=DATA, COLUMN=N2, BUCKET=measure, $
*GRAPH_JS
    "legend": {
        "title": {"font": "16pt Sans-Serif",
                  "color":"green"},
        "labels": {
            "font": "16pt Sans-Serif",
            "content": "{{series_label}} : {{series_percent}}"

        }
    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



凡例領域の枠線の生成

lineStyle プロパティは、凡例領域の周囲に描画される枠線の幅および色を制御します。

構文　凡例領域の枠線の生成

```
"legend": {
            "lineStyle": {
      "width": number,
      "color": "color",
      "dash": "string"'
   }
}
```

説明

`"width": number,`

線の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です (線なし)。

`"color": "color"`

線の色を定義します。次のいずれかを指定します。

- 色名または数値指定の文字列で色を定義する文字列、あるいはグラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "black" です。
- グラデーションを定義する JSON オブジェクト。

"dash": "string"

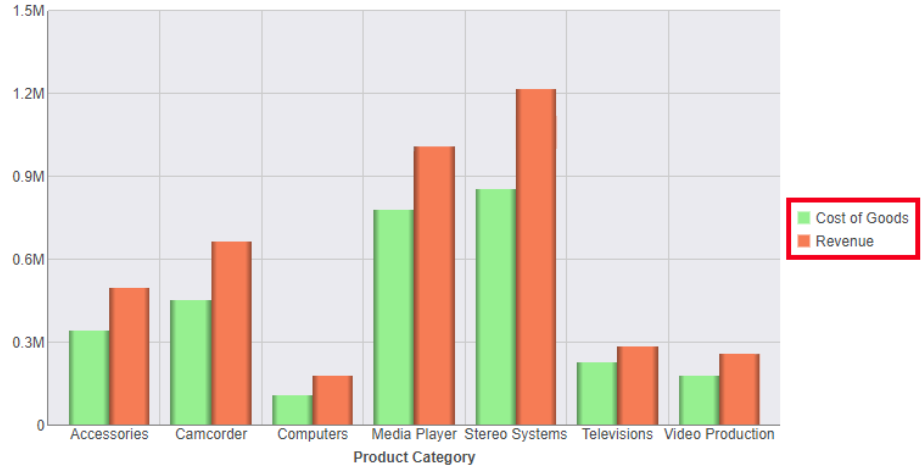
線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。

例　凡例領域の枠線の生成

次のリクエストは、凡例領域の周囲に 4 ピクセル幅の赤色の枠線を生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {
"lineStyle": {
     "width": 4,
     "color": "red"},
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"}
    ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



凡例マーカーの位置の制御

markerPosition プロパティは、凡例ラベルを基準に凡例マーカーの位置を制御します。

構文　凡例マーカーの位置の制御

```
"legend": {"markerPosition": "string"}
```

説明

`"markerPosition": "string"`

凡例ラベルを基準に凡例マーカーの位置を定義する文字列です。有効な値には、次のものがあります。

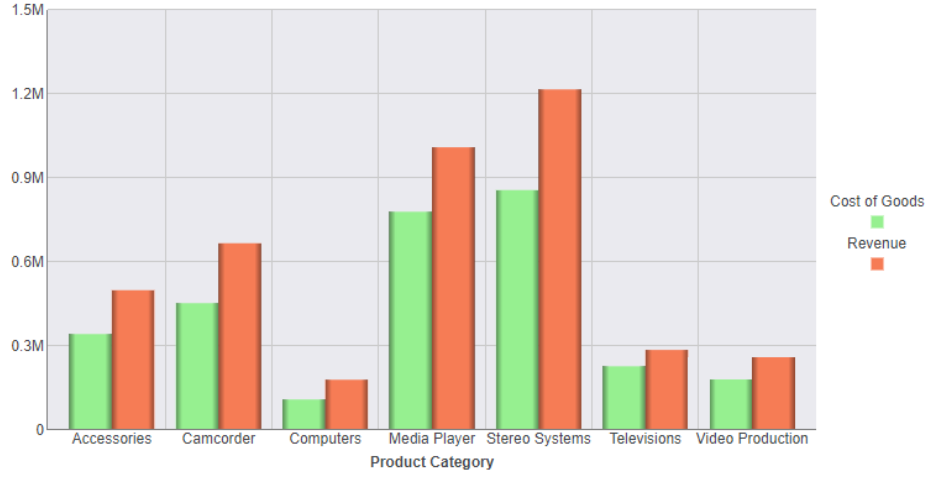
- "bottom"
- "left" - これがデフォルト値です。
- "right"
- "top"

例　　凡例マーカーの位置の制御

次のリクエストは、凡例マーカーを凡例ラベルの下側に配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"markerPosition": "bottom"},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

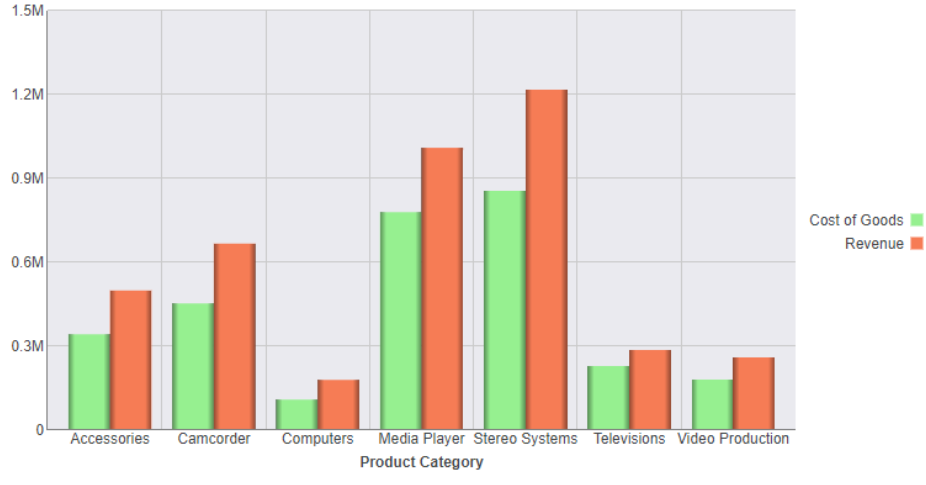
出力結果は次のとおりです。



次のリクエストでは、凡例マーカーを凡例ラベルの右側に配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {
   "markerPosition": "right"
   },
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

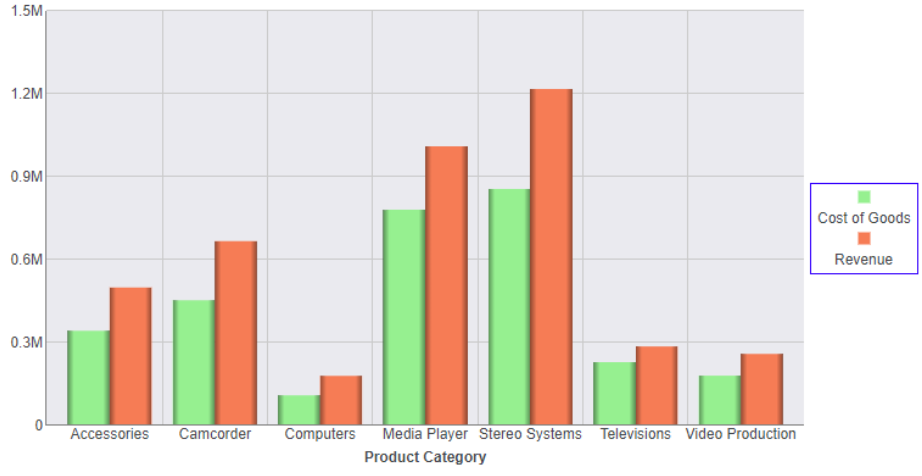
出力結果は次のとおりです。



次の凡例プロパティでは、マーカーの位置を "top" に変更します。また、凡例の周囲に枠線を描画します。

```
"legend": {
        "markerPosition": "top",
    "lineStyle": {
        "color": "blue"}}
```

上記の凡例プロパティから、次のグラフが生成されます。



凡例マーカーのサイズの制御

markerSize プロパティは、凡例マーカーのサイズを制御します。

構文　　凡例マーカーのサイズの制御

```
"legend": {"markerSize": number}
```

説明

`"markerSize": number`

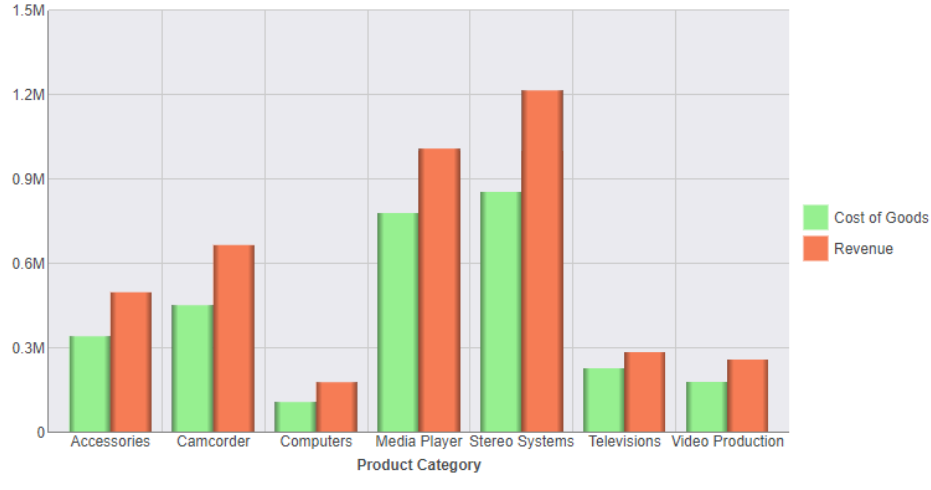
凡例マーカーのサイズをピクセル単位で定義します。デフォルト値は 8 です。

例　　凡例マーカーのサイズの制御

次のリクエストは、凡例マーカーのサイズを 20 に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"markerSize": 20},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



折れ線グラフの凡例マーカーとシリーズマーカーの一致

legend:matchSeriesMakers プロパティを使用して、線シリーズの凡例マーカーに折れ線グラフまたは複合グラフの線の外観を正確に反映させることができます。このプロパティを true に設定した場合、線の凡例エントリには線が含まれ、この線にマーカーが表示される場合のみマーカーも含まれます。マーカーが追加された折れ線グラフの場合、凡例には同一のマーカーが表示されます。マーカーが追加されていない折れ線グラフの場合、凡例にはマーカーが表示されません。

構文　凡例マーカーとシリーズマーカーの一致

```
"legend":{"matchSeriesMarkers":boolean}
```

説明

`"matchSeriesMarkers":boolean`

線シリーズの凡例マーカーの外観を制御します。有効な値には、次のものがあります。

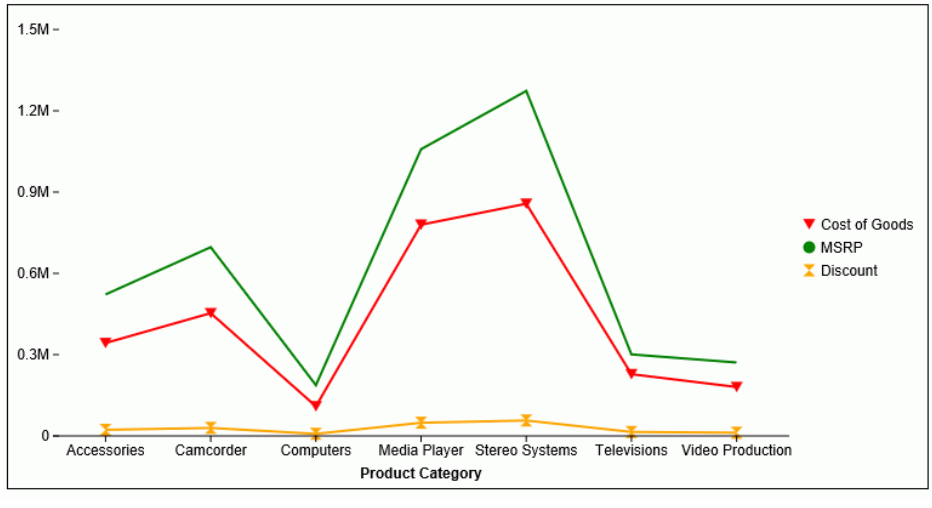
- true - 線を含む凡例マーカーを生成します。グラフの線にマーカーが追加されている場合、凡例の線には同一のマーカーが表示されます。グラフの線にマーカーが追加されていない場合、凡例の線にはマーカーが表示されません。
- false - 線を含まない (マーカーのみの) 凡例を生成します。グラフの線にマーカーが追加されている場合、凡例には同一のマーカーが表示されます。グラフの線にマーカーが追加されていない場合、凡例にはデフォルトのマーカーが表示されます。これがデフォルト値です。

例　折れ線グラフの凡例マーカーとシリーズマーカーの一致

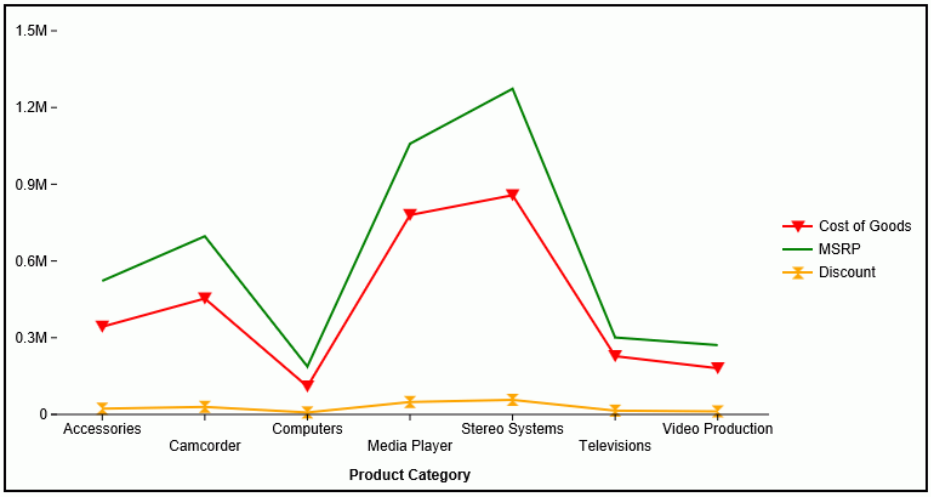
次のリクエストでは、凡例プロパティ matchSeriesMarkers が false (デフォルト) に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US  DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis,$
TYPE=DATA, COLUMN=MSRP_US , BUCKET=y-axis,$
TYPE=DATA, COLUMN=DISCOUNT_US, BUCKET=y-axis,$
*GRAPH_JS
"legend": {"matchSeriesMarkers": false},
"axisList": {
"y1":{"majorGrid":{"visible":false}},
"x1":{"majorGrid":{"visible":false}}
},
"series":[
          {"series":0, "marker":{"shape":"triangle", "size":10}},
          {"series":1,  "marker":{"visible": false}},
          {"series":2, "marker":{"shape":"hourglass", "size":10}}
          ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。このグラフのシリーズ 1 (MSRP_US) にはマーカーが追加されていないため、シリーズ 1 の凡例マーカーはデフォルトマーカーになります。シリーズ 0 およびシリーズ 2 の凡例マーカーには、このグラフの線シリーズと同一のマーカーが表示されます。



下図は、matchSeriesMarkers を true に変更して出力した結果を示しています。このグラフのシリーズ 1 (MSRP_US) にはマーカーが追加されていないため、シリーズ 1 の凡例マーカーは線のみでマーカーは表示されません。シリーズ 0 およびシリーズ 2 の凡例マーカーには、このグラフの線シリーズと同一のマーカーが線を含めて表示されます。



凡例領域の最大パーセントの設定

"legend": "maxSize" プロパティを使用して、凡例に使用できる描画領域の最大パーセントを設定することができます。

構文　　凡例領域の最大パーセントの設定

```
"legend":
{
 "maxSize": "string"

}
```

説明

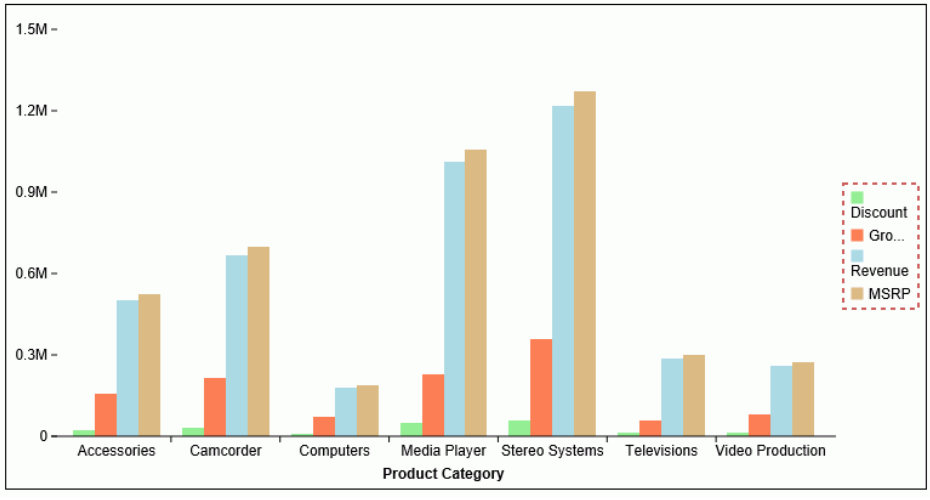
`"maxSize": "string"`
パーセント記号を含む文字列で、凡例領域のパーセントを示します。デフォルト値は "50%" です。

例　凡例領域の最大パーセントの設定

次のリクエストは、maxSize プロパティを 10% に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column = discount_us, bucket = y-axis,$
type=data, column = gross_profit_us, bucket = y-axis,$
type=data, column = revenue_us, bucket = y-axis,$
type=data, column = msrp_us, bucket = y-axis,$
type=data, column = product_category, bucket = x-axis,$
*GRAPH_JS
"xaxis":{"majorGrid": {"visible": false},
"minorGrid": {"visible": false}},
"yaxis":{"majorGrid": {"visible": false},
"minorGrid": {"visible": false}},
"legend": {
"maxSize": "10%",
"lineStyle": {
"width": 2,
"color": "indianred",
"dash": "4 4"}},
"series": [
{"series": 0, "color": "lightgreen"},
{"series": 1, "color": "coral"},
{"series": 2, "color": "lightblue"},
{"series": 3, "color": "burlywood"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。凡例は、割り当てられた領域に収まっていません。



凡例の方向の制御

"legend": "orientation" プロパティを使用して、凡例項目を縦方向に表示するか横方向に表示するかを制御することができます。

構文　　凡例の方向の制御

```
"legend":
{
"orientation": "string"
}
```

説明

`"string"`

次のいずれかの値です。

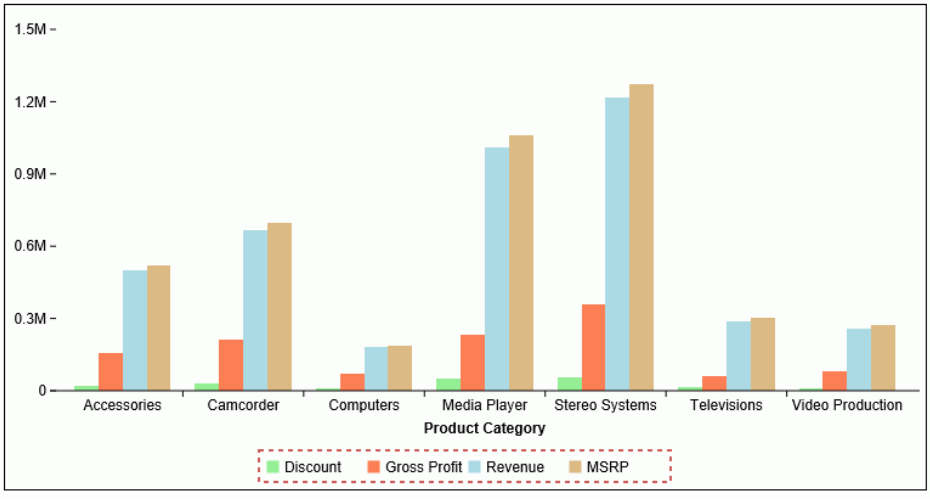
- **"auto"** - グラフエンジンにより方向が定義されます。これがデフォルト値です。
- **"vertical"** - 凡例項目を縦方向に表示します。
- **"horizontal"** - 凡例項目を横方向に表示します。

例　凡例の方向の制御

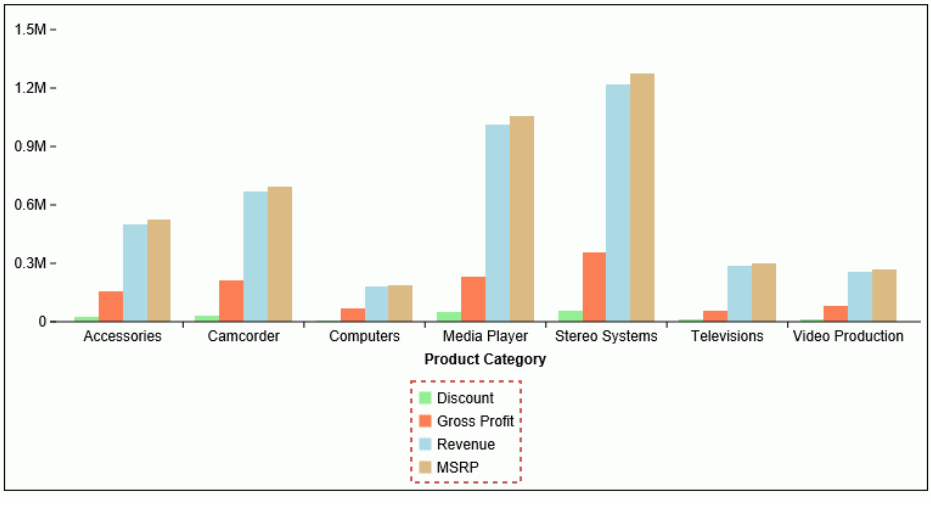
次のリクエストは、グラフの下側に凡例を配置し、凡例の方向についてはデフォルト値を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column = discount_us, bucket = y-axis,$
type=data, column = gross_profit_us, bucket = y-axis,$
type=data, column = revenue_us, bucket = y-axis,$
type=data, column = msrp_us, bucket = y-axis,$
type=data, column = product_category, bucket = x-axis,$

*GRAPH_JS
"xaxis":{"majorGrid": {"visible": false},
         "minorGrid": {"visible": false}},
"yaxis":{"majorGrid": {"visible": false},
         "minorGrid": {"visible": false}},
"legend": {
    "position": "bottom",
    "orientation": "auto",
"lineStyle": {
    "width": 2,
    "color": "indianred",
    "dash": "4 4"}},
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"},
    {"series": 2, "color": "lightblue"},
    {"series": 3, "color": "burlywood"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。グラフエンジンにより、凡例項目が横方向に表示されています。



orientation プロパティを "orientation": "vertical" に変更した場合の出力結果は、次のとおりです。



凡例の位置の制御

position プロパティは、凡例の位置を制御します。

構文　　凡例の位置の制御

```
"legend": {"position": position}
```

説明

`"position": position`

凡例の位置を定義する文字列または JSON オブジェクトです。

文字列の場合、有効は値には次のものがあります。

- "bottom"
- "left"
- "right" - これがデフォルト値です。
- "top"
- "auto" - グラフエンジンにより、凡例の位置が制御されます。

厳密な配置には、JSON オブジェクトを使用してください。次のプロパティが利用可能です。

```
{
   "side": "string",
   "anchor": "string",
  "align": "string",
   "excludeFromLayout": boolean
   "left": number,
   "right": number
   "top": number,
   "bottom": number
 }
```

説明

`"side": "string"`
: "left"、"right"、"bottom"、"top"、または undefined のいずれかです。

`"align": "string"`
: "side": "left" または "side": "right" の場合、"top"、"middle"、"bottom" のいずれかを指定できます。"side": "top" または "side": "bottom" の場合、"left"、"middle"、"right" のいずれかを指定できます。

`"anchor": "string"`
: 凡例の配置参照として使用されるオブジェクトです。"chartBackground"、"chartFrame"、"chartAxisFrame" のいずれかを指定できます。デフォルト値は "chartFrame" です。

"excludeFromLayout": boolean
フレーム内の領域を凡例用に予約するかどうかを指定します。true に設定された場合、凡例領域はグラフ領域内に設定されず、グラフ上を浮動します。凡例の side プロパティが定義されている場合、デフォルト値は false です。それ以外の場合は、true です。

"left": number
凡例が左側に設定されている場合、アンカーオブジェクトと凡例の左端との間隔をピクセル数で定義します。凡例が右側に設定されている場合は無視されます。

"right": number
凡例が右側に設定されている場合、アンカーオブジェクトと凡例の右端との間隔をピクセル数で定義します。凡例が左側に設定されている場合は無視されます。

"top": number
凡例が上側に設定されている場合、アンカーオブジェクトと凡例の上端との間隔をピクセル数で定義します。凡例が下側に設定されている場合は無視されます。

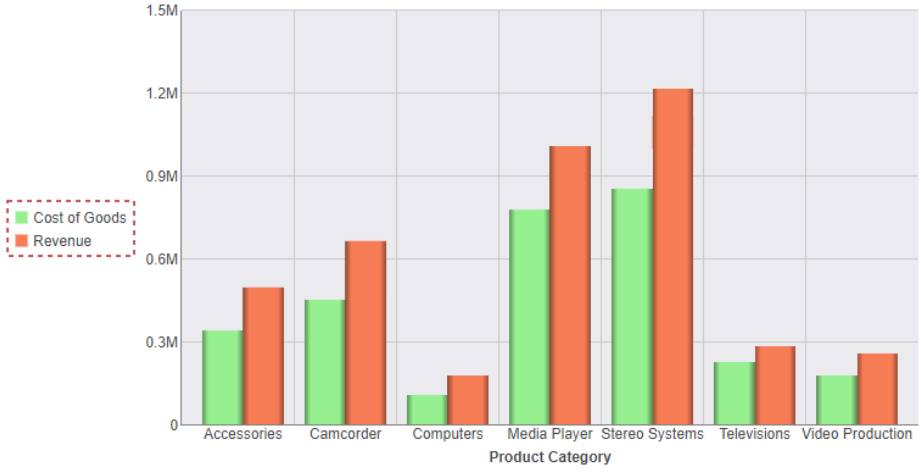
"bottom": number
凡例が下側に設定されている場合、アンカーオブジェクトと凡例の下端との間隔をピクセル数で定義します。凡例が上側に設定されている場合は無視されます。

例　凡例の位置の制御

次のリクエストは、凡例をグラフの左側に移動し、凡例の周囲に枠線を描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {
   "position": "left",
   "lineStyle": {
      "width": 2,
      "color": "indianred",
      "dash": "4 4"}},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

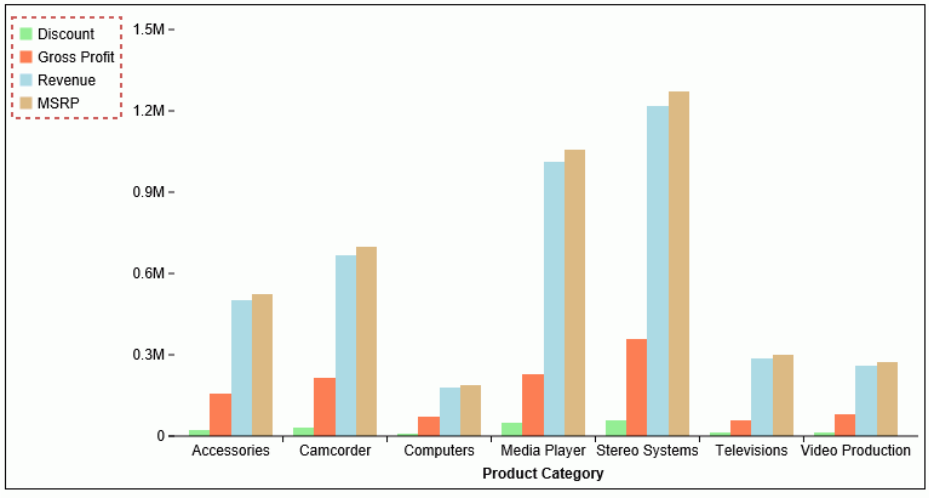


例　JSON オブジェクトを使用した凡例の位置の制御

次のリクエストは、凡例をグラフ背景の左上に、左端から 5 ピクセル、上端から 10 ピクセルの位置に配置します。凡例領域は予約されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column = discount_us, bucket = y-axis,$
type=data, column = gross_profit_us, bucket = y-axis,$
type=data, column = revenue_us, bucket = y-axis,$
type=data, column = msrp_us, bucket = y-axis,$
type=data, column = product_category, bucket = x-axis,$
*GRAPH_JS
"xaxis":{"majorGrid": {"visible": false},
        "minorGrid": {"visible": false}},
"yaxis":{"majorGrid": {"visible": false},
         "minorGrid": {"visible": false}},
"legend": {
"position":
 {
   "side": "left",
   "align": "top",
   "anchor": "chartBackground",
   "excludeFromLayout": false,
   "left": 5,
   "top": 10,
 },
"lineStyle": {
"width": 2,
"color": "indianred",
"dash": "4 4"}
},
"series": [
{"series": 0, "color": "lightgreen"},
{"series": 1, "color": "coral"},
{"series": 2, "color": "lightblue"},
{"series": 3, "color": "burlywood"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



凡例項目の順序の入れ替え

"legend" :"reverseOrder" プロパティを使用して、凡例項目のデフォルト順序を入れ替えることができます。

構文　凡例項目の順序の入れ替え

```
"legend":
{
 "reverseOrder": boolean
}
```

説明

`"reverseOrder": boolean`

凡例項目の順序を制御します。次のいずれかの値です。

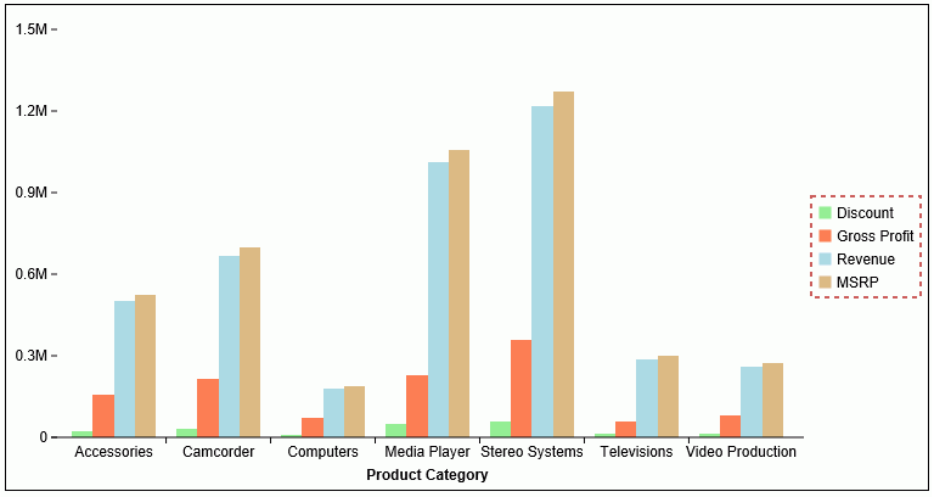
- **"auto"** - グラフエンジンにより、凡例項目の順序が制御されます。
- **true** - 縦方向の凡例項目は下から上へ、横方向の凡例項目は右から左へ描画されます。
- **false** - 縦方向の凡例項目は上から下へ、横方向の凡例項目は左から右へ描画されます。これがデフォルト値です。

例　凡例順序の入れ替え

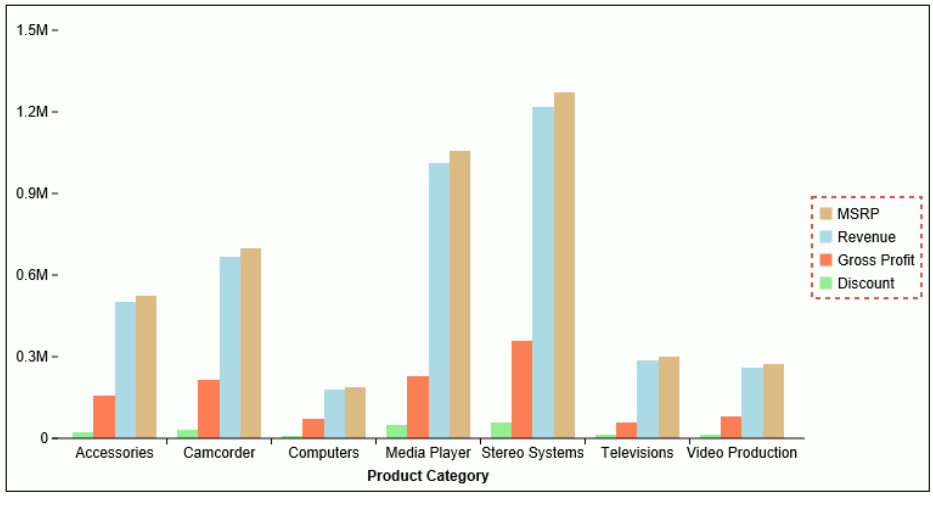
次のリクエストは、棒グラフを生成し、デフォルト設定の凡例順序 ("reverseOrder": false) を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column = discount_us, bucket = y-axis,$
type=data, column = gross_profit_us, bucket = y-axis,$
type=data, column = revenue_us, bucket = y-axis,$
type=data, column = msrp_us, bucket = y-axis,$
type=data, column = product_category, bucket = x-axis,$

*GRAPH_JS
"xaxis":{"majorGrid": {"visible": false},
         "minorGrid": {"visible": false}},
"yaxis":{"majorGrid": {"visible": false},
         "minorGrid": {"visible": false}},
"legend": {
    "reverseOrder": false,
    "lineStyle": {
        "width": 2,
        "color": "indianred",
        "dash": "4 4"}},
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"},
    {"series": 2, "color": "lightblue"},
    {"series": 3, "color": "burlywood"}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



下図は、"reverseOrder" を true に変更して出力した結果を示しています。凡例項目の順序が逆になります。



スクロール凡例の生成

シリーズ数が多すぎると、凡例領域にすべてのシリーズが収まらない場合があります。その場合は、スクロール凡例を有効にし、スクロールバーおよびスクロールハンドルのプロパティを設定することができます。スクロールバーは、凡例全体を表示するためにスクロールバーが必要となる場合のみ生成されます。

構文　スクロール凡例の生成

```
"legend":
  {
  "scroll": {
     "enabled": boolean,
     "size": number,
     "color": "string",
     "handle": {
        "color": "string",
        "hoverColor": "string",
        "border": {
            "width": number,
            "color": "string",
            "dash": "string"
                   }
                 }
           }
  }
```

説明

`"enabled": boolean`

必要に応じて凡例をスクロール可能にするかどうかを制御します。有効な値には、次のものがあります。

- true - 凡例のスクロールを有効にします。
- false - 凡例のスクロールを無効にします。これがデフォルト値です。

`"size": number`

スクロールバーの幅をピクセル単位で定義します。デフォルト値は 15 です。

`"color": "string"`

スクロールバーの塗りつぶし色を、色指定またはグラデーションの文字列で定義します。デフォルト色は "rgb(240, 240, 240)" です。

"handle":
スクロールバーハンドルのプロパティを定義します。

"color": "string"

スクロールバーのハンドル色を、色指定の文字列で定義します。デフォルト色は "grey" です。

"hoverColor": "string"

スクロールバーハンドルにマウスポインタを置いた場合またはクリックした場合のハンドル色を、色指定の文字列で定義します。デフォルト色は "rgb(88, 88, 88)" です。

"border":
スクロールバーハンドル境界線のプロパティを定義します。

"width": number

ハンドル境界の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

"color": "string"

ハンドル境界の色を、色指定の文字列で定義します。デフォルト値は "transparent" です。

"dash": "string"

境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

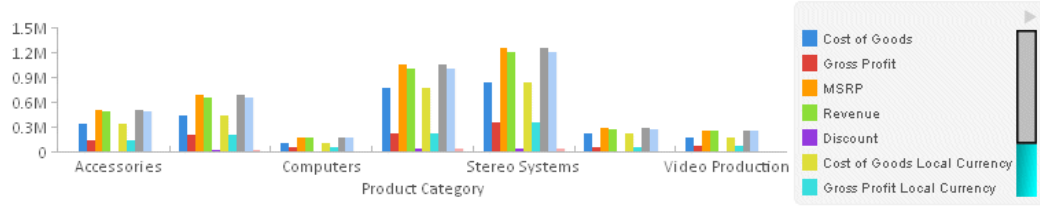
破線を定義する文字列では、複数の破線スタイルを指定することができます。たとえば、「"2 4 4 2"」と指定すると、境界線の破線スタイルが交互に入れ替わります。

例　　スクロール凡例の生成

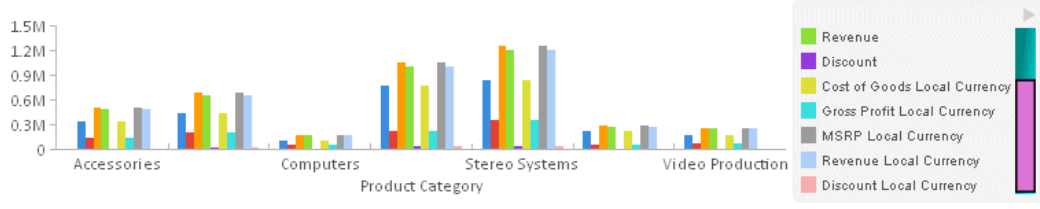
次のリクエストは、スクロール凡例を生成します。 スクロールバーの塗りつぶし色は青緑からシアンに移行する線状グラデーションに設定され、ハンドルの色は銀色、境界線は黒色に設定されています。スクロールバーの上にマウスポインタを置くか、クリックした時点で、塗りつぶし色が赤紫色に変わります。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US MSRP_US REVENUE_US DISCOUNT_US
COGS_LOCAL GROSS_PROFIT_LOCAL MSRP_LOCAL REVENUE_LOCAL DISCOUNT_LOCAL
BY PRODUCT_CATEGORY
ON GRAPH SET VAXIS 150
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENID_Default.sty,$
*GRAPH_JS
"xaxis": {"labels": {"rotation": 0}},
"legend": {"scroll": {
    "enabled": true,
    "size": 15,
    "color": "linear-gradient(0,0,100%,100%, 20% teal, 95% cyan)",
    "handle": {
        "color": "silver",
        "hoverColor": "orchid",
        "border": {
            "width": 3,
            "color": "black",
            "dash": ""
            }
        }}}
*END
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N5, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N6, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N7, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N8, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N9, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N10, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N11, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
ENDSTYLE
END
```

下図は、クリックする前のスクロールバーを示しています。



下図は、ハンドルを凡例領域の最下部へドラッグした後のスクロールバーを示しています。



ドッキング凡例の生成

ドッキング凡例は、アクティブ展開ボタンのみが表示される、最小化された (折りたたまれた) 凡例です。凡例のマーカーを表示させることもできます。

構文　　ドッキング凡例の生成

```
"legend": {
                    "dock"{
            "enabled": boolean,
            "animate": boolean,
            "minimized": boolean,
            "resizeFrame": boolean,
            "showMarkers": boolean,
            "expandDirection": "string",
            "button": {
                                                    "size": number,
              "color": "string",
              "hoverColor": "string",
              "border": {
                    "width": number,
                    "color": "string",
                    "dash": "string"
                      }
                    }
                }
        }
```

説明

`"enabled": boolean`

ドッキング凡例を生成するかどうかを制御します。有効な値には、次のものがあります。

- true - ドッキング凡例を有効にします。
- false - ドッキング凡例を無効にします。これがデフォルト値です。

`"animate": boolean`

スライドアクションを使用して凡例の展開と折りたたみをアニメーション化するかどうかを制御します。有効な値には、次のものがあります。

- true - 凡例の展開と折りたたみをアニメーション化します。これがデフォルト値です。
- false - アニメーション化を無効にします。

`"minimized": boolean`

初期状態で凡例を折りたたみ表示にするか、展開表示にするかを制御します。有効な値には、次のものがあります。

- true - 初期状態でドッキング凡例を折りたたみ表示にします。
- false - 初期状態でドッキング凡例を展開表示にします。これがデフォルト値です。

`"resizeFrame": boolean`

凡例の展開表示と折りたたみ表示に応じてフレームを拡大縮小するかどうかを制御します。有効な値には、次のものがあります。

- true - 凡例の展開表示と折りたたみ表示に応じてフレームを拡大縮小します。これがデフォルト値です。
- false - 凡例の展開表示と折りたたみ表示に応じてフレームを拡大縮小しません。

`"showMarkers": boolean`

ドッキング凡例を折りたたみ表示にした場合に凡例マーカーを表示するかどうかを制御します。有効な値には、次のものがあります。

- true - 凡例を折りたたみ表示にした場合に凡例マーカーを表示します。これがデフォルト値です。
- false - 凡例を折りたたみ表示にした場合に凡例マーカーを表示しません。

"expandDirection": "string"

凡例の位置が free に設定されている場合に、凡例の展開方向を指定します ("left"、"right"、"top"、"bottom")。凡例の位置が free に設定されていない場合、この値は undefined です。凡例の展開方向はグラフエンジンによって決定されます。

"button":

展開ボタンのプロパティを定義します。

"size": number

ボタンのサイズをピクセル単位で定義するか (オブジェクトの中心から形状の最縁端までの距離の 2 倍で定義される)、"auto" を指定して凡例マーカーと同一サイズでボタンを描画します。デフォルト値は "auto" です。

"color": "string"

ボタンの色を定義する色指定の文字列です。デフォルト値は "grey" です。

"hoverColor": "string"

ボタン上にマウスポインタを置くか、ボタンをクリックした際のボタンの色を定義する色指定の文字列です。デフォルト値は "black" です。

"border":

展開ボタン境界線のプロパティを定義します。

"width": number

ボタンの境界の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

"color": "string"

ボタン境界の色を定義する色指定の文字列です。デフォルト値は "transparent" です。

"dash": "string"

境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

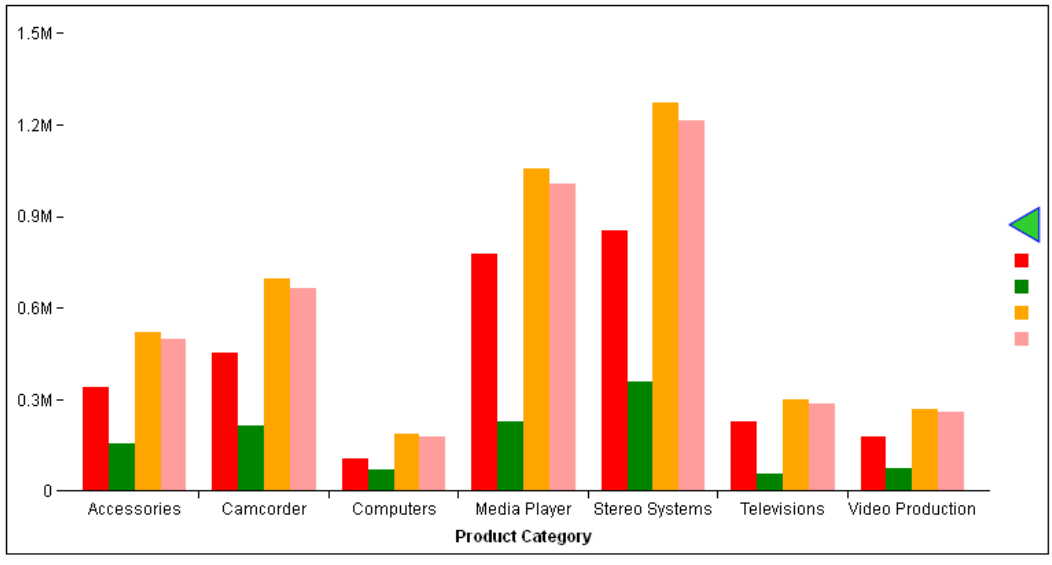
破線を定義する文字列では、複数の破線スタイルを指定することができます。たとえば、「"2 4 4 2"」と指定すると、境界線の破線スタイルが交互に入れ替わります。

例　ドッキング凡例の生成

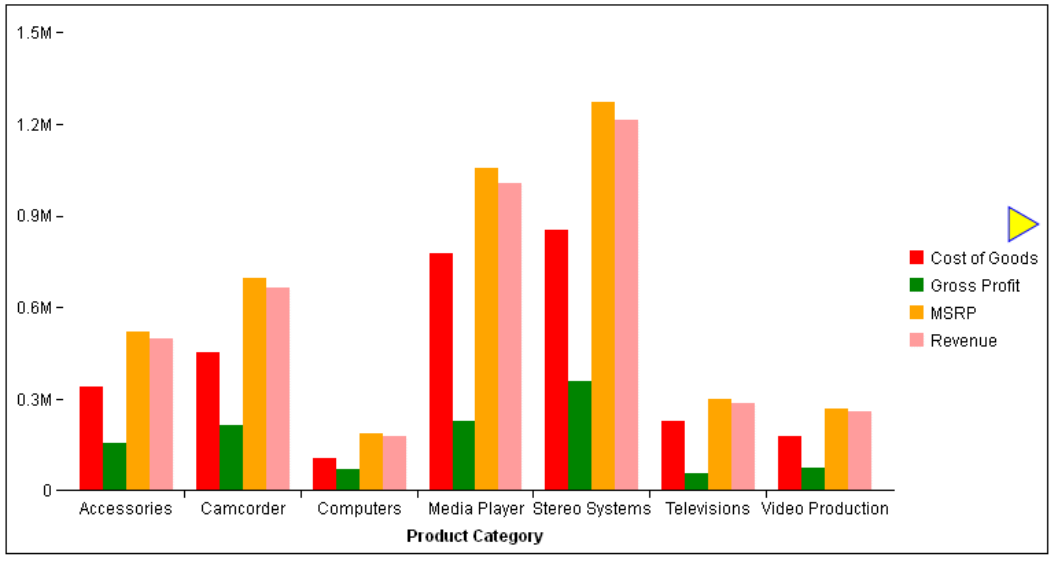
次のリクエストは、初期表示で折りたたまれた凡例を生成し、凡例マーカーを表示します。展開ボタンは、直径 25 ピクセルの黄緑色、青色の境界に設定され、展開ボタン上にマウスポインタを置くか、クリックした際に黄色に変わります。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
MSRP_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
*GRAPH_JS
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"legend": {"dock": {
    "enabled": true,
    "animate": true,
    "minimized": true,
    "resizeFrame": true,
    "showMarkers": true,
    "expandDirection": "undefined",
    "button": {
        "size": 25,
        "color": "limegreen",
        "hoverColor": "yellow",
        "border": {
            "width": 1,
            "color": "blue",
            "dash": ""
            }}
    }}
*END

TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=MSRP_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

下図は、初期表示で折りたたまれた凡例を示しています。



下図では、展開された凡例、およびボタン上にマウスポインタを置いたときの色の変化が示されています。グラフフレームのサイズが変更されています。



凡例タイトルのフォーマット設定

title プロパティは、凡例タイトルの表示およびフォーマットを制御します。

構文　凡例タイトルのフォーマット設定

```
"legend":
  {
   "title":
   {
      "visible": boolean,
      "text": "string",
      "font": "string",
      "color": "string"
      "tooltip": "string",
      "wrap": number
   }
 }
```

説明

`"visible": boolean`
次のいずれかの値を指定します。

- true - 凡例タイトルを表示します。
- false - 凡例タイトルを表示しません。これがデフォルト値です。
- "auto" - 凡例タイトルの表示と配置をグラフエンジンが自動的に決定します。

`"text": "string"`

凡例タイトルのテキストを定義する文字列です。デフォルト値は "Legend Title" です。

`"font": "string"`

凡例タイトルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列で凡例タイトルの色を定義します。デフォルト値は "black" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"tooltip": "string"`
マウスポインタを凡例タイトル上に置いたときに表示する文字列、または文字列を返す関数を指定します。

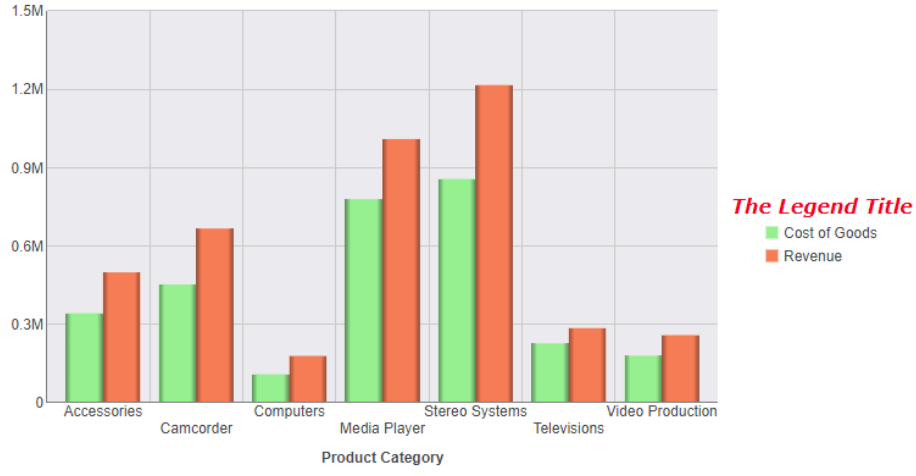
"wrap": number
長いタイトルの折り返し行数を定義します。整数を指定します。必要に応じて、長いタイトルをこの行数に折り返すか、"auto" を指定して、長いタイトルを必要な行数に折り返します。デフォルト値は 1 です。

例　凡例タイトルのフォーマット設定

次のリクエストは、凡例タイトルのフォーマットを赤色、太字、斜体、12pt Verdana フォントに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {
  "title": {
      "visible": true,
      "text": "The Legend Title",
      "font": "bold italic 12pt Verdana",
      "color": "red"
      }},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



凡例領域の表示の制御

visible プロパティは、凡例領域の表示を制御します。

構文　凡例領域の表示の制御

```
"legend": {"visible": boolean}
```

説明

`"visible": boolean`

次のいずれかの値を指定します。

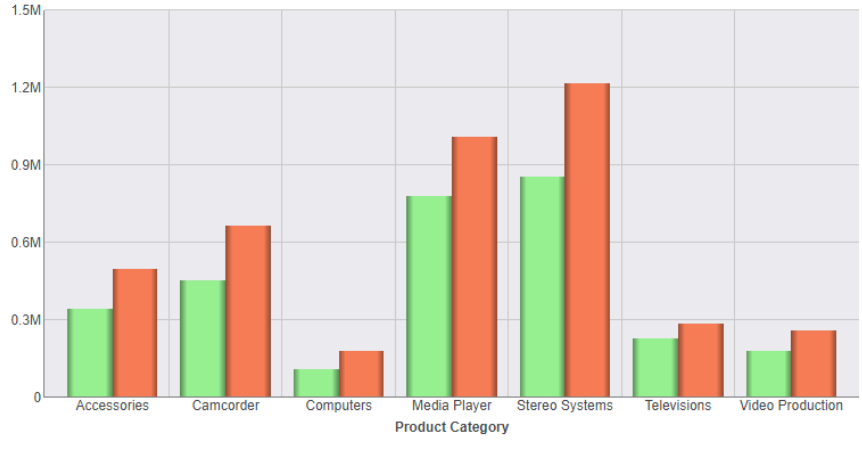
- true - 凡例領域を表示します。これがデフォルト値です。
- false - 凡例領域を表示しません。
- "auto" - グラフに複数のシリーズが存在する場合に凡例領域を表示します。

例　　凡例領域の表示の制御

次のリクエストは、グラフの凡例領域を非表示にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": false},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



サイズ凡例のフォーマット設定

サイズ凡例は、メジャーが size 属性カテゴリに割り当てられている場合に生成されます。サイズ凡例は、互いに重なり合う 2 つまたは 3 つの円として描画され、各円の大きさはグラフ内のマーカーのサイズと一致します。

構文　サイズ凡例のフォーマット設定

```
"legend":
{
  "sizeLegend":
  {
    "layout": "string",
    "lineStyle":
     {
      "width": number,
      "color": "string",
      "dash": "string"
      }
   }
}
```

説明

`"layout": "string"`
サイズ凡例の外観を定義します。有効な値は、"circle" または "quarterCircle" です。デフォルト値は "circle" です。

`"lineStyle"`
円または四分円のプロパティを定義します。

`"width": number`
円または四分円の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

`"color": "string"`
円または四分円の色を定義する文字列です。デフォルト値は "black" です。

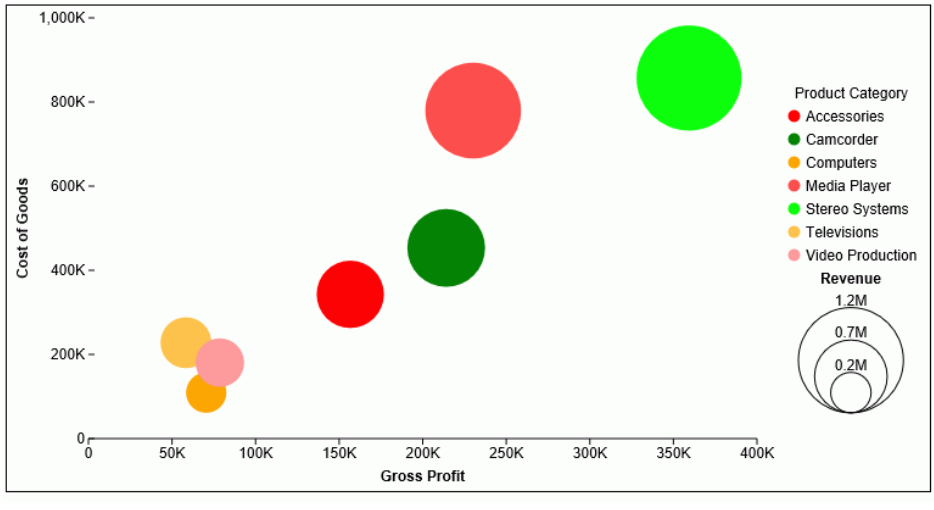
`"dash": "string"`
円または四分円の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。

例　サイズ凡例のフォーマット設定

次のリクエストは、メジャーが size 属性カテゴリに割り当てられたバブルグラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=x-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=size,
*GRAPH_JS
"xaxis": {"majorGrid": {"visible":false},
          "minorGrid": {"visible":false}},
"yaxis": {"majorGrid": {"visible":false},
          "minorGrid": {"visible":false}},
*END
ENDSTYLE
END
```

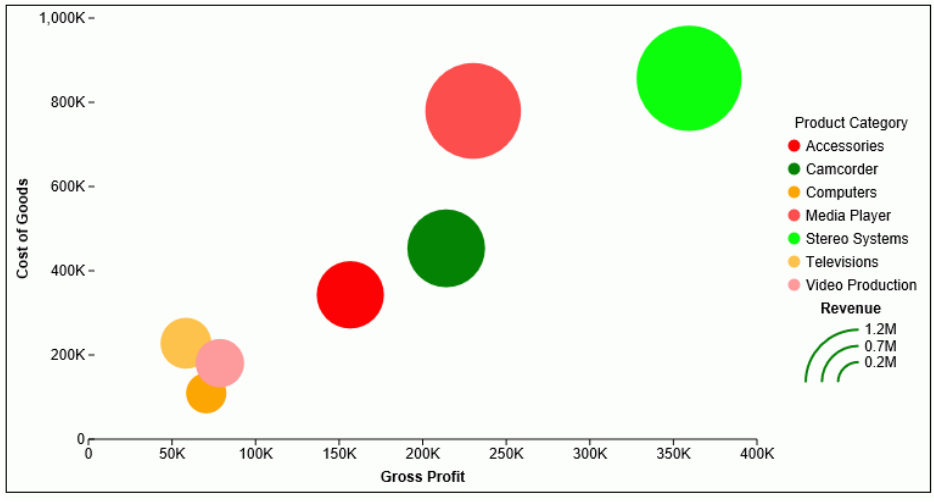
出力結果は次のとおりです。デフォルト設定のサイズ凡例が、3 つの黒い円として表示されています。



次のリクエストでは、サイズ凡例を緑色の四分円に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=x-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=size,
*GRAPH_JS
"legend":
    {
  "sizeLegend":
  {
    "layout": "quarterCircle",
    "lineStyle":
     {
      "width": 2,
      "color": "green",
      "dash": ""
      }
   }
},
"xaxis": {"majorGrid": {"visible":false},
          "minorGrid": {"visible":false}},
"yaxis": {"majorGrid": {"visible":false},
          "minorGrid": {"visible":false}},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



8

軸のプロパティ

ここでは、グラフの軸の外観を制御する方法について説明します。

トピックス

- 軸のプロパティの概要
- 軸リストの作成
- 軸の罫線間のセグメントを交互に塗りつぶし
- ライザ幅の制御
- 積み上げ棒グラフのライザのソート
- 数値軸ベースラインの外観の制御
- 対数目盛りの有効化
- 軸線のフォーマット設定
- 色バンドの定義
- 主罫線、目盛り、ラベルの個数の制御
- Y 軸主罫線と Y2 軸主罫線の同期
- 数値軸の方向の制御
- 軸ラベルのフォーマット設定
- 主罫線と目盛りのフォーマット設定
- 数値軸での最小値および最大値の定義
- 複数 Y 軸および分割 Y 軸グラフの最大値と最小値の同期
- 数値軸の最小値ラベルと最大値ラベルの置換
- 棒グラフの数値軸のゼロ値の表示
- 補助罫線と目盛りのフォーマット設定
- X 軸のラベルレイアウトの制御
- X 軸スクロールのフォーマット設定
- 横方向スクロールバーの初期位置の設定
- 軸モードの設定
- 数値軸ラベルのフォーマットの制御
- 軸のデフォルト表示位置の入れ替え
- 軸タイトルのフォーマット設定
- 時間軸のフォーマット設定

軸のプロパティの概要

軸のプロパティは、グラフの軸 (X 軸、Y 軸、Y2 軸、Z 軸) の外観を制御します。軸は、グラフの描画に使用されるデータに応じて、数値軸にすることも、序数軸 (文字列) にすることもできます。Y2 軸は、パレートグラフのほか、シリーズの yAxisAssignment プロパティで特定のシリーズが y2axis に割り当てられた棒グラフ、折れ線グラフ、面グラフにのみ表示されます。Z 軸は、3D グラフおよびヒートマップにのみ表示されます。

次のコードセグメントは、デフォルト設定を示しています。

```
"xaxis": {


  "mode": undefined,
  "min": undefined,
  "max": undefined,
  "mustIncludeZero": true,
  "intervalMode": undefined,
  "intervalValue": undefined,
  "bIsLog": false,
  "numberFormat": "auto",
  "swapChartSide": false,
  "invert": false,
  "altFrameColor": undefined,
  "colorBands": [],


"groupFit":
        {"rule": "auto",
         "value": undefined
        },


  "title": {
    "text": "the WebFOCUS column title",
    "visible": false,
    "font": "bold 9pt Sans-Serif",
    "color": "black"
  },
  "labels": {
    "visible": true,
    "font": "7.5pt Sans-Serif",
    "color": "black",
    "excludeMin": false,
    "excludeMax": false,
    "nestingConcatSymbol": "null",
    "nestingLineStyle": {
                                                "width": 1,
            "color": "black",
            "dash": ""
        }
    "rotation": undefined
  },
  "baseLineStyle": {
    "width": 1,
    "color": "black",
    "dash": ""
  },
  "bodyLineStyle": {
    "width": 1,
    "color": "transparent",
    "dash": ""
  },
```

```
"majorGrid": {
  "visible": true,
  "aboveRisers": true,
  "lineStyle": {
    "width": 1,
    "color": "rgba(255, 255, 255, 0.3)",
    "dash": ""
  },
  "ticks": {
    "length": 5,
    "visible": true,
    "style": "inner",
    "lineStyle": {
      "width": 1,
      "color": "black"
    }
  }
},
"minorGrid": {
  "visible": false,
  "count": undefined,
  "lineStyle": {
    "width": 1,
    "color": "black",
    "dash": ""
  },
  "ticks": {
    "length": 5,
    "visible": false,
    "style": "inner",
    "lineStyle": {
      "width": 1,
      "color": "black"
    }
  }
},

"labelLayout": {
  "stagger": "auto",
  "skip": "auto",
  "truncate": "auto"
              },
"scroll": {
  "initialPosition": undefined,
  "style": "simple",
     "miniChartProperties": {
     "height": n,
     "dataselection": "selectionRect",
     "intervalButtons": "auto"
                                                    },

                                 }
},
```

```
"yaxis": {
  "mode": undefined,
  "min": undefined,
  "max": undefined,
  "mustIncludeZero": true,
  "intervalMode": undefined,
  "intervalValue": undefined,
  "bIsLog": false,
  "numberFormat": "auto",
  "swapChartSide": false,
  "invert": false,
  "altFrameColor": undefined,
  "colorBands": [],
  "title": {
    "text": "Y Axis Title",
    "visible": false,
    "font": "bold 9pt Sans-Serif",
    "color": "black"
    "position": "orthogonal"
},
  "labels": {
    "visible": true,
    "font": "7.5pt Sans-Serif",
    "color:" "black",
    "excludeMin": false,
    "excludeMax": false,
    "replaceMin": undefined,
    "replaceMax": undefined,
    "rotation": undefined
  },
  "baseLineStyle": {
    "width": 1,
    "color": "black",
    "dash": ""
  },
  "bodyLineStyle": {
    "width": 1,
    "color": "transparent",
    "dash": ""
  },
```

```
 "majorGrid: {
    "visible: true,
    "aboveRisers: true,
    "lineStyle: {
      "width: 1,
      "color: "rgba(255, 255, 255, 0.3)",
      "dash: ""
    },
    "ticks: {
      "length: 5,
      "visible: true,
      "style: "inner",
      "lineStyle: {
        "width: 1,
        "color: "black"
      }
    }
  },
  "minorGrid": {
    "visible": false,
    "count": undefined,
    "lineStyle": {
      "width": 1,
      "color": "black",
      "dash": ""
    },
    "ticks": {
      "length": 5,
      "visible": false,
      "style": "inner",
      "lineStyle": {
        "width": 1,
        "color": "black"
      }
  },
  "colorScale":{ (deprecated, use the chart level colorScale object)
    "colors": ["#253494", "#2C7FB8", "#41B6C4", "#A1DAB4"]
  }
},
```

```
"y2axis": {
  "min": undefined,
  "max": undefined,
  "mustIncludeZero": true,
  "intervalMode": undefined,
  "intervalValue": undefined,
  "bIsLog": false,
  "numberFormat": "auto",
  "invert": false,
  "altFrameColor": undefined,
  "colorBands": [],
  "title": {
    "text": "Y2 Axis Title",
    "visible": false,
    "font": "bold 9pt Sans-Serif",
    "color": "black"
    "position": "orthogonal" or "flip"
 },
  "labels": {
    "visible": true,
    "font": "7.5pt Sans-Serif",
    "color:" "black",
    "excludeMin": false,
    "excludeMax": false,
    "replaceMin": undefined,
    "replaceMax": undefined,
  },
  "baseLineStyle": {
    "width": 1,
    "color": "black",
    "dash": ""
  },
  "bodyLineStyle": {
    "width": 1,
    "color": "transparent",
    "dash": ""
  },
```

```
 "majorGrid": {
    "lockToY1": false,
    "visible": true,
    "aboveRisers": true,
    "lineStyle": {
      "width": 1,
      "color": "rgba(255, 255, 255, 0.3)",
      "dash": ""
    },
    "ticks": {
      "length": 5,
      "visible": true,
      "style": "inner",
      "lineStyle": {
        "width": 1,
        "color": "black"
      }
    }
  },
  "minorGrid": {
    "visible": false,
    "count": undefined,
    "lineStyle": {
      "width": 1,
      "color": "black",
      "dash": ""
    },
    "ticks": {
      "length": 5,
      "visible": false,
      "style": "inner",
      "lineStyle": {
        "width": 1,
        "color": "black"
      }
    }
  }
},
```

```
zaxis: {
  "mode": undefined,
  "min": undefined,
  "max": undefined,
  "mustIncludeZero": true,
  "intervalMode": undefined,
  "intervalValue": undefined,
  "bIsLog": false,
  "numberFormat": "auto",
  "swapChartSide": false,
  "invert": false,
  "altFrameColor": undefined,
  "colorBands": [],
  "title": {
    "text": "Z Axis Title",
    "visible": false,
    "font": "bold 9pt Sans-Serif",
    "color": "black"
    "position": "orthogonal"
},
  "labels": {
    "visible": true,
    "font": "7.5pt Sans-Serif",
    "color:" "black",
    "excludeMin": false,
    "excludeMax": false,
    "replaceMin": undefined,
    "replaceMax": undefined,
    "rotation": undefined
  },
  "baseLineStyle": {
    "width": 1,
    "color": "black",
    "dash": ""
  },
  "bodyLineStyle": {
    "width": 1,
    "color": "transparent",
    "dash": ""
  },
```

```
 "majorGrid": {
    "visible": false,
    "aboveRisers": true,
    "lineStyle": {
      "width": 1,
      "color": "rgba(255, 255, 255, 0.3)",
      "dash": ""
    },
    "ticks": {
      "length": 5,
      "visible": true,
      "style": "outer",
      "lineStyle": {
        "width": 1,
        "color": "black"
      }
    }
  },
  "minorGrid": {
    "visible": false,
    "count": undefined,
    "lineStyle": {
      "width": 1,
      "color": "black",
      "dash": ""
    },
    "ticks": {
      "length": 5,
      "visible": false,
      "style": "inner",
      "lineStyle": {
        "width": 1,
        "color": "black"
      }
    }
  }
},
```

注意："yaxis": "colorScale" プロパティは廃止され、colorScale プロパティに置き換えられました。colorScale プロパティについての詳細は、819 ページの 「 色スケールの色モードの定義 」 を参照してください。

軸リストの作成

xaxis および yaxis プロパティを使用する代わりに、axisList を作成することで、任意の数の X 軸および Y 軸をサポートすることができます。xaxis および yaxis のプロパティはすべて、axisList 内の軸でサポートされます。

軸のプロパティの指定には、軸リストの使用が推奨されます。

構文　軸リストを使用した軸のプロパティの指定

```
axisList: {
                x1: {xaxis_properties},
                x2: {xaxis_properties},
                  ...
                xi: {xaxis_properties},
                y1: {yaxis_properties},
                y2: {yaxis_properties},
                  ...
                yj: {yaxis_properties}
}
```

説明

`x1, ... xi, y1 ... yj`
　任意の数の X 軸および Y 軸です。

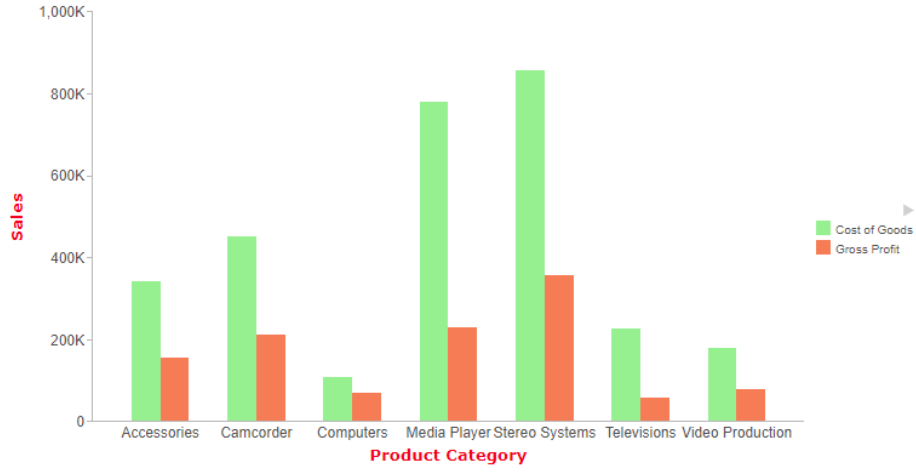
`xaxis_properties, yaxis_properties`
　サポートされる軸のプロパティです。

例　軸リストの使用

次のリクエストは、軸リストを使用して棒グラフの X 軸タイトルおよび Y 軸タイトルのフォーマットを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket = x-axis,$
type=data, column=n2, bucket = y-axis,$
type=data, column=n3, bucket = y-axis,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
*GRAPH_JS_FINAL
"axisList": {
"x1": {"title": {
"visible": true, "color": "red", "font": "bold 10pt Verdana"}},
"y1": {"title": {
"visible": true, "color": "red", "font": "bold 10pt Verdana", "text":
"Sales"}}
},
"series": [
{"series": 0, "color": "lightgreen"},
{"series": 1, "color": "coral"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



軸の罫線間のセグメントを交互に塗りつぶし

altFrameColor プロパティは、軸の罫線間のセグメントを特定の色で交互に塗りつぶします。

構文　軸の罫線間のセグメントを交互に塗りつぶし

```
"axisname":
{
  "altFrameColor": "color"},
```

説明

`axisname`

axisList の軸にすることも、xaxis、yaxis、y2axis、zaxis のいずれかにすることもできます。

`"altFrameColor": "color"`

次のいずれかの値を指定します。

- 色名または数値指定の文字列で定義する色、あるいは文字列で定義するグラデーション。
- グラデーションを定義する JSON オブジェクト。

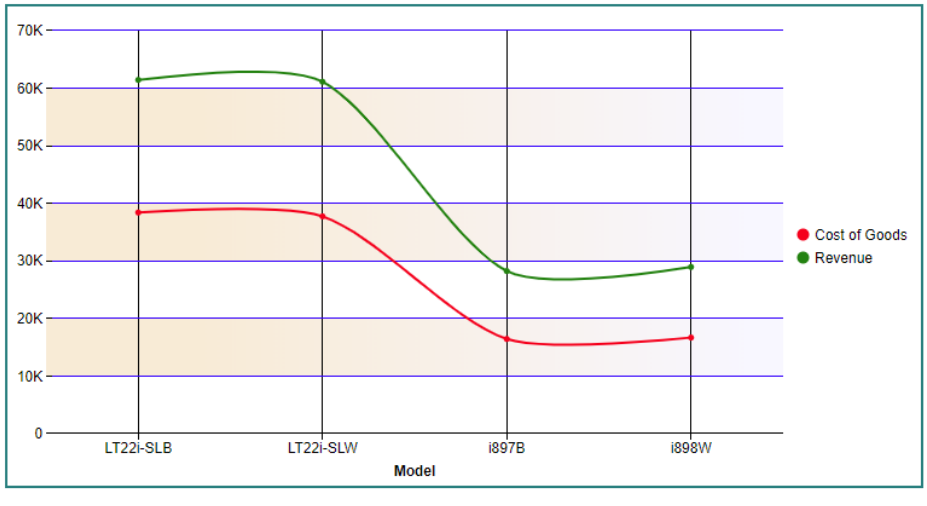
色およびグラデーションの定義についての詳細は、63 ページの「 色およびグラデーション 」 を参照してください。

例　軸の罫線間のセグメントを交互に塗りつぶし

次のリクエストは、WF_RETAIL_LITE データソースを使用して縦折れ線グラフを生成し、Y 軸の主罫線間のセクションを 2 つの色で交互に塗りつぶします。一方の色は透明色で、他方の色はアンティークホワイトからゴーストホワイトに変化するグラデーションです。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {
  "altFrameColor":
        "linear-gradient(0%,0%,100%,0%, 20% antiquewhite, 95% ghostwhite)",
    "majorGrid": {"visible": true,
        "lineStyle": {"width": 1, "color": "blue"}
    }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ライザ幅の制御

デフォルト設定では、グラフエンジンはコンテナのサイズに合わせてグラフを出力します。ライザの一部がコンテナ内に正しく収まらない場合は、スクロールバーが生成されます。AUTOFIT パラメータを RESIZE に設定した場合、コンテナのサイズが変更されると、ライザのサイズも変更されます。

"xaxis": "groupFit" プロパティを使用して、ライザ幅を制御することができます。ライザの最小幅、最大幅、または正確な幅を設定するか、スクロールバーが生成される場合は、フレーム内に表示する最小グループ数を指定することができます。

構文　ライザ幅の制御

```
"axisname": {
 "groupFit":
   {"rule": "string",
      "value": value      }
   }
```

説明

`"axisname"`
axisList の軸または xaxis です。

`"rule": "string"`
ライザ幅の決定方法を制御する次のルールのいずれかを指定します。

- **"auto"**　グラフフレームの幅に応じてライザ幅を決定します。これがデフォルト値です。
- **"minSize"**　value で指定したピクセル値以上にライザ幅を設定します。
- **"maxSize"**　value で指定したピクセル値以下にライザ幅を設定します。
- **"exactSize"**　value で指定した正確なピクセル値を常にライザ幅に設定します。
- **"maxCount"**　ライザの一部がコンテナ内に正しく収まらない場合にスクロールバーを生成すると同時に、value で指定した数のライザをグラフに表示します。
- **"labelHeight"**　横棒グラフのライザの幅を、X 軸のグループラベルのフォントサイズに指定します。

"value": value

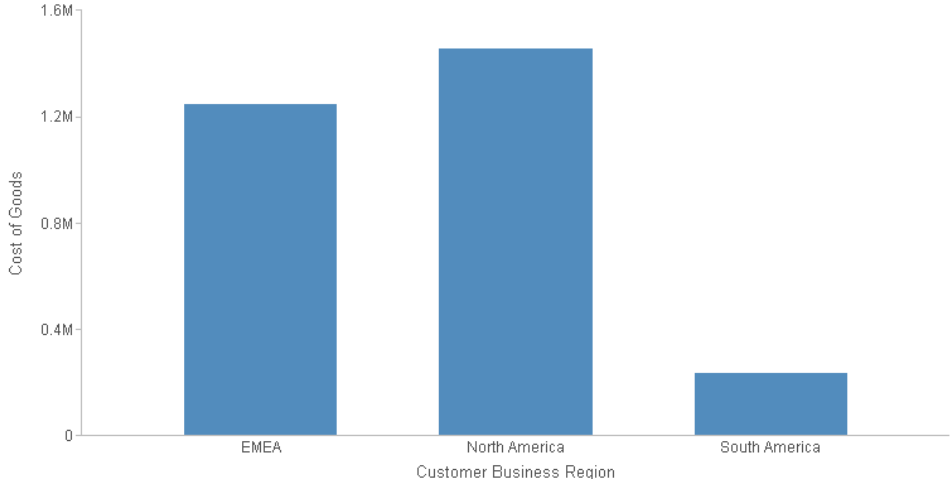
"minSize"、"maxSize"、"exactSize" のいずれかの場合、幅をピクセル数で指定します。"scrollCount" の場合、グループ数を指定します。"auto" の場合、この値は無視されます。"labelHeight" の場合、ラベルの高さにパディングを追加します。この場合、パーセント文字列またはピクセル数を指定します。デフォルト値は undefined です。

例　ライザ幅の制御

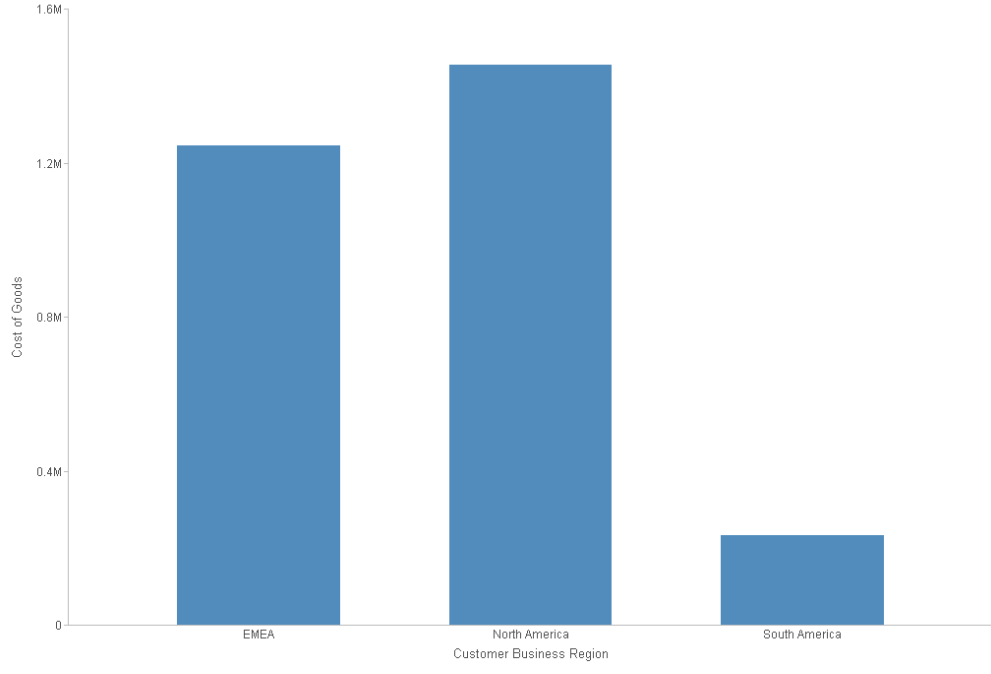
次のリクエストは、ライザ幅が自動的に変更されるグラフを生成します。AUTOFIT パラメータが RESIZE に設定されているため、フレームのサイズが変更されると、ライザの幅も変更されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT RESIZE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=side-by-side,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=x-axis, $
*END
ENDSTYLE
END
```

下図は、初期状態のグラフを示しています。



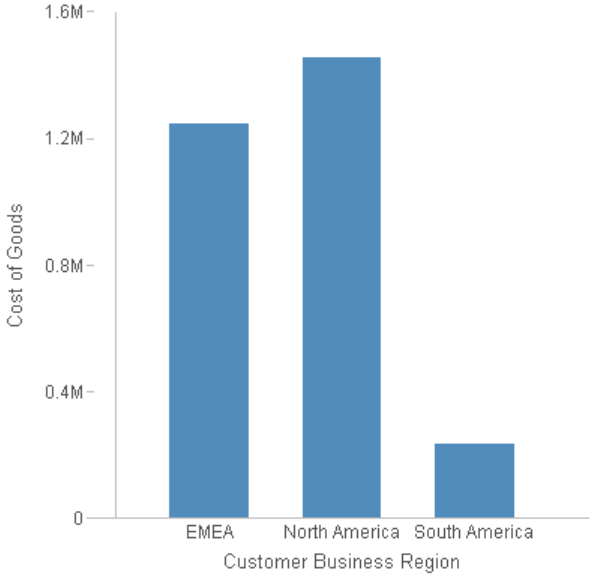
下図は、フレームが最大サイズに変更された場合の同一グラフを示しています。グラフフレームに合わせてライザ幅が拡大されます。



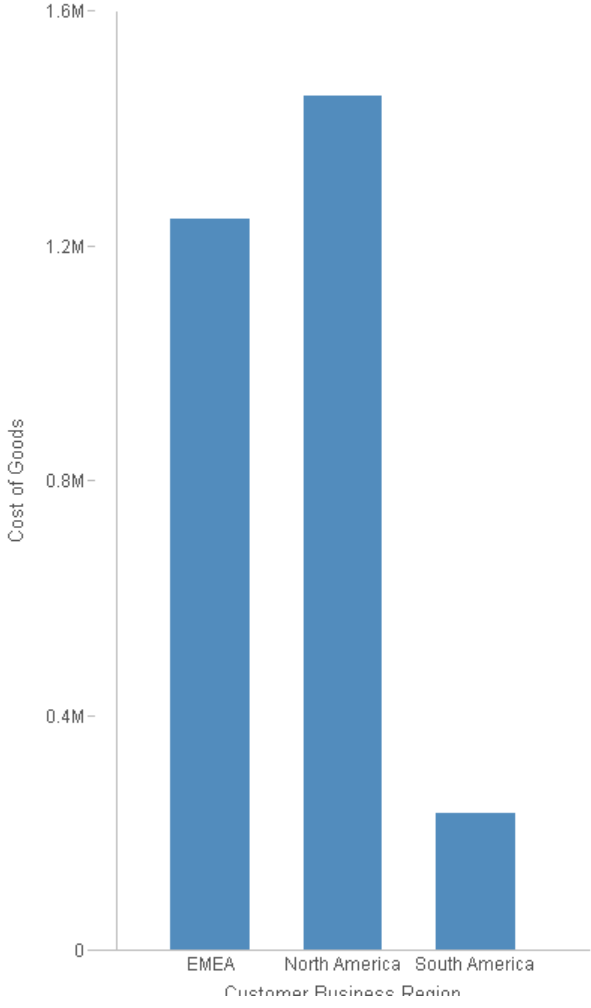
次のリクエストでは、groupFit プロパティを使用して、ライザを正確に 100 ピクセル幅に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT RESIZE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=side-by-side,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=x-axis, $
*GRAPH_JS
"xaxis": {"groupFit": {
    "rule": "exactSize",
    "value": 100
    }}
*END
ENDSTYLE
END
```

下図は、初期状態のグラフを示しています。



下図は、フレームが最大サイズに変更された場合のグラフを示しています。ライザ幅は変更されていません。

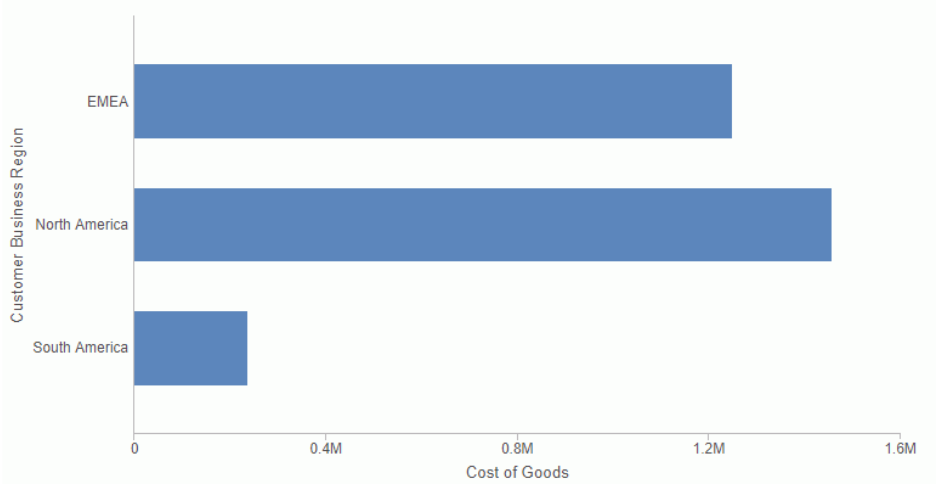


例　横棒グラフのライザ幅をグループラベルのフォントサイズに設定

次のリクエストは、横棒グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-ORIENTATION=HORIZONTAL,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
ENDSTYLE
END
```

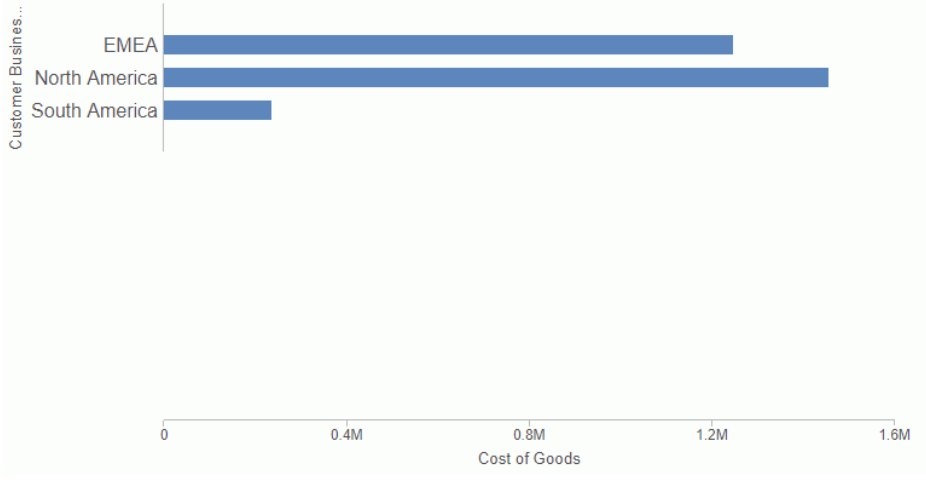
出力結果は次のとおりです。棒の幅はグループラベルよりかなり広くなっています。



次のリクエストでは、ラベルのフォントサイズを設定し、groupFit の labelHeight ルールを使用して、棒の幅をこのフォントサイズに一致させ、パディングファクタ 80% を追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-ORIENTATION=HORIZONTAL,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
*GRAPH_JS
"xaxis":
       {
       "labels": {"font": "12pt Sans-Serif"},
       "groupFit":
         {
          "rule": "labelHeight",
          "value": "80%"
          }
        }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



積み上げ棒グラフのライザのソート

積み上げ棒グラフでは、1 本のライザが、グループ内のすべてのシリーズを積み上げた合計値を表します。軸の sort プロパティを使用して、これらの積み上げライザを、数値軸の値に基づいて昇順または降順にソートすることができます。

構文　積み上げ棒グラフのライザのソート

```
"axisname": {"sort": {"order": "string", "by": "string"}}
```

説明

`"axisname"`
ソートされたライザを描画する軸です。axisList の軸にすることも、xaxis、yaxis、y2axis、zaxis のいずれかにすることもできます。

`"sort"`
ソートのプロパティを定義します。

`"order": "string"`
ライザのソート方法を指定します。有効な値には、次のものがあります。

- "ascending"
- "descending"

デフォルト設定では、ライザはソートされません。

`"by": "string"`
ソートする値の数値軸を指定します。有効な値には、次のものがあります。

- "x"
- "x*n*" - *n* は分割 X 軸のインデックス番号です。
- "y"
- "y*n*" - *n* は分割 Y 軸のインデックス番号です。
- "z"
- "matrix*n*" - *n* はソートする行のインデックス番号 (縦棒グラフの場合) または列のインデックス番号 (横棒グラフの場合) です。

  注意：マトリックス行または列のインデックス番号は 0 (ゼロ) から開始されます。つまり、"matrix0" が 1 行目または 1 列目を示します。

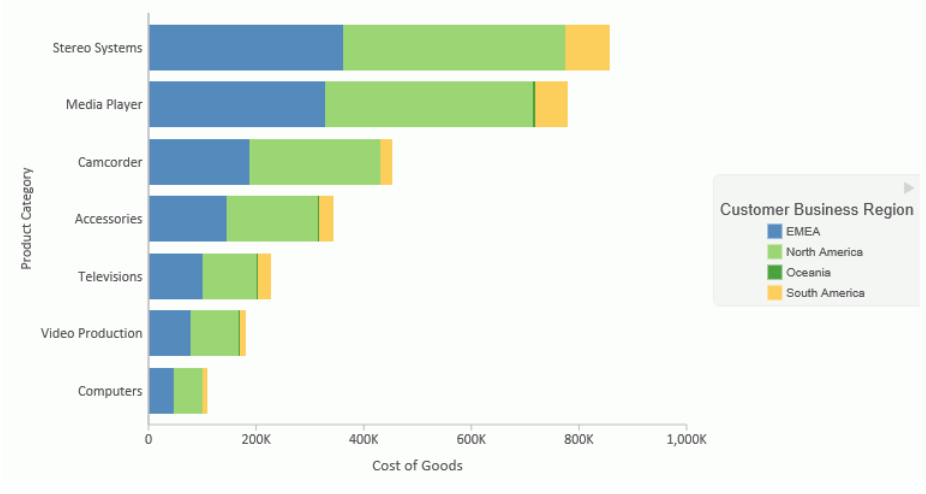
- "matrix*n*, y*i*" - マトリックス行または列内の分割 Y 軸でソートする場合に使用します。

例　積み上げ棒グラフのライザの降順ソート

次のリクエストは、積み上げ横棒グラフを生成し、X 軸の各ライザを Y 軸値の降順でソートします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=stacked, CHART-ORIENTATION=horizontal,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"axisList": {
"x1": {"sort": {"order": "descending", "by": "y"}}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

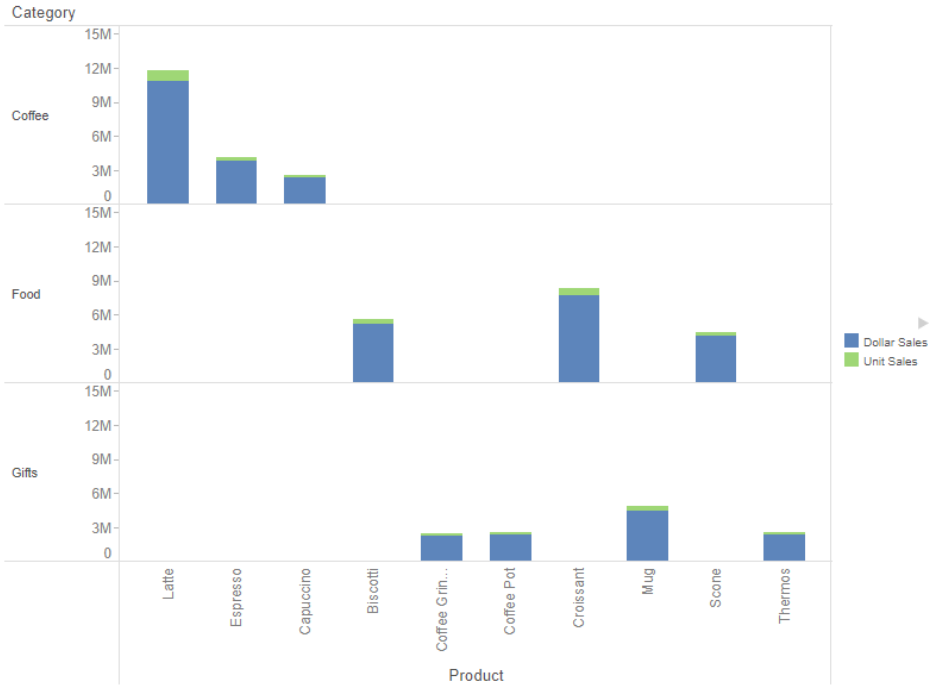


例　マトリックス行でのソート

次のリクエストは、マトリックスグラフを生成し、行番号 0 (ゼロ) で降順にソートします。

```
GRAPH FILE GGSALES
SUM DOLLARS UNITS
BY CATEGORY
BY PRODUCT
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VZERO OFF
ON GRAPH SET AUTOFIT ON
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/WFC/Global/Themes/Standard/Default/theme.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=stacked,$
TYPE=DATA, COLUMN=N1, BUCKET=row, $
TYPE=DATA, COLUMN=N2, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"axisList": {
  "x1": {
    "sort": {
            "order": "descending",
            "by": "matrix0"
            }
      }
          }

*END
ENDSTYLE
END
```

出力結果は次のとおりです。



数値軸ベースラインの外観の制御

数値軸では、baseLineStyle プロパティを使用して、軸ベースラインの外観を制御することができます。ベースラインは、罫線に平行な線で、通常は 0 (ゼロ) 値を通過します。

構文　数値軸ベースラインの外観の制御

```
"axisname":
{
  "baseLineStyle": {
                "width": number,
    "color": "string",
    "dash": "string"
  }
}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"width": number`

ベースラインの幅をピクセル単位で定義します。

`"color": "string"`

色名または数値指定でベースラインの色を定義する文字列です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"dash": "string"`

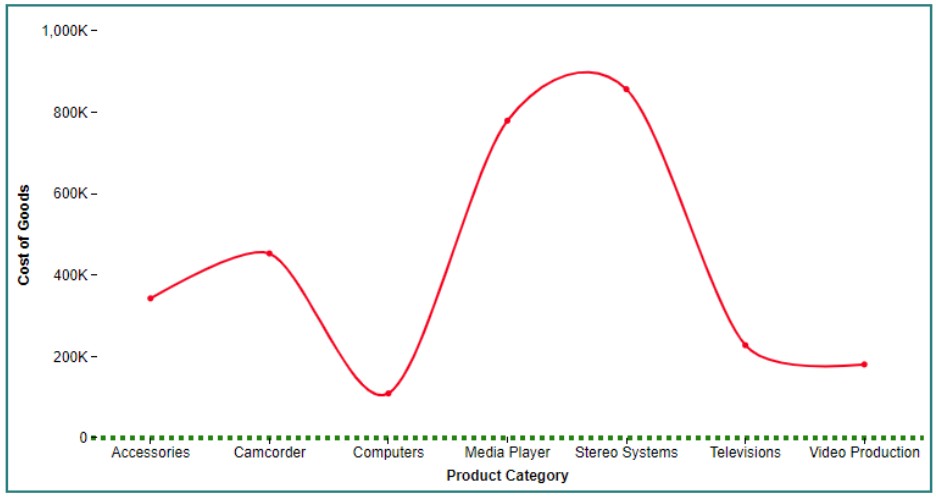
破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　数値軸ベースラインの外観の制御

次のリクエストは、WF_RETAIL_LITE データソースを使用して縦折れ線グラフを生成し、Y 軸のベースラインを緑色の破線に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"yaxis": {"majorGrid": {"visible": false},
   "baseLineStyle": {
      "width": 4, "color": "green", "dash": "4 4"},
"xaxis":  {"majorGrid": {"visible": false}   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



対数目盛りの有効化

数値軸では、bIsLog プロパティを使用して、対数目盛りを有効または無効にすることができます。

注意：サポートされる対数目盛りの底は 10 のみです。

構文　対数目盛りの有効化

```
"axisname": {
  "bIsLog": boolean  }
```

説明

`"axisname"`
対数目盛りを表示する数値軸を指定します。axisList の数値軸にすることも、xaxis、yaxis、zaxis のいずれかの数値軸にすることもできます。

`"bIsLog: boolean`
次のいずれかの値を指定します。

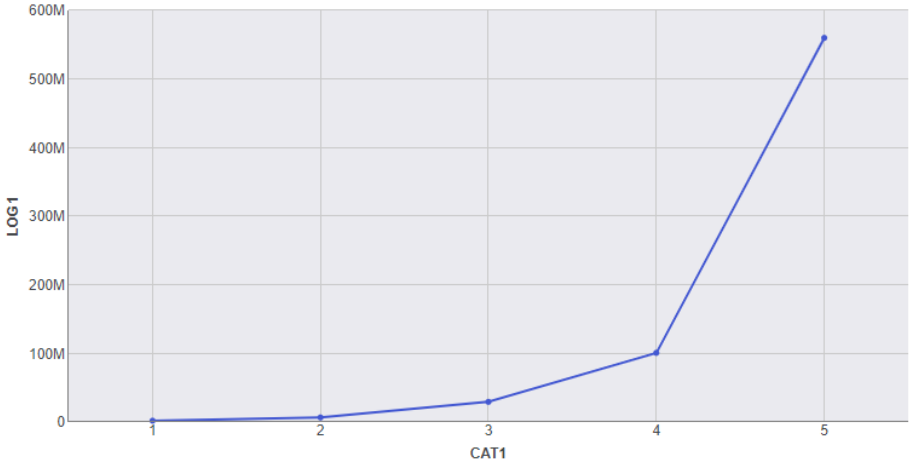
- true - 数値軸で対数目盛りを有効にします。
- false - 数値軸で対数目盛りを無効にします。これがデフォルト値です。

例　数値軸での対数目盛りの有効化

次のリクエストは、WF_RETAIL_LITE データソースを使用して、売上原価 (COGS_US) の範囲に応じて桁数の異なる新しい売上原価を生成し、これらの売上原価をカテゴリ別に分類します。次に縦折れ線グラフを生成し、数値 Y 軸に新しい売上原価を表示します。

```
DEFINE FILE WF_RETAIL_LITE
LOG1= IF COGS_US GT 500 AND COGS_US LT 1000 THEN  COGS_US  *10
      ELSE IF COGS_US GE 1000 AND COGS_US LT 2000 THEN COGS_US  * 100
      ELSE IF COGS_US GE 2000 AND COGS_US LT 3000 THEN COGS_US  * 1000
      ELSE IF COGS_US GE 3000  AND COGS_US LT 3300 THEN COGS_US * 10000
      ELSE COGS_US ;
CAT1/I2= IF COGS_US GT 500 AND COGS_US LT 1000 THEN  2
      ELSE IF COGS_US GE 1000 AND COGS_US LT 2000 THEN 3
      ELSE IF COGS_US GE 2000 AND COGS_US LT 3000 THEN 4
      ELSE IF COGS_US GE 3000  AND COGS_US LT 3300 THEN 5
      ELSE 1 ;
END

GRAPH FILE WF_RETAIL_LITE
SUM LOG1
BY CAT1
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
ENDSTYLE
END
```

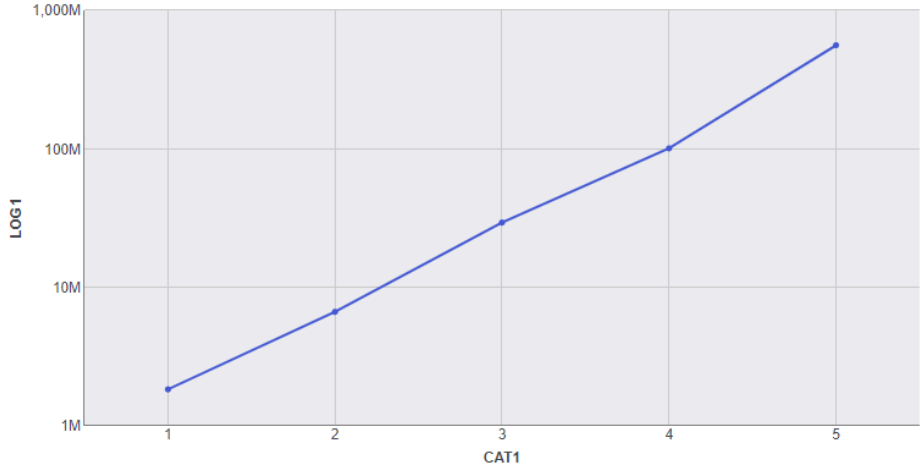
出力結果は次のとおりです。



次のリクエストでは、Y 軸で対数目盛りを有効にします。

```
DEFINE FILE WF_RETAIL_LITE
LOG1= IF COGS_US GT 500 AND COGS_US LT 1000 THEN  COGS_US  *10
      ELSE IF COGS_US GE 1000 AND COGS_US LT 2000 THEN COGS_US  * 100
      ELSE IF COGS_US GE 2000 AND COGS_US LT 3000 THEN COGS_US  * 1000
      ELSE IF COGS_US GE 3000  AND COGS_US LT 3300 THEN COGS_US * 10000
      ELSE COGS_US ;
CAT1/I2= IF COGS_US GT 500 AND COGS_US LT 1000 THEN  2
      ELSE IF COGS_US GE 1000 AND COGS_US LT 2000 THEN 3
      ELSE IF COGS_US GE 2000 AND COGS_US LT 3000 THEN 4
      ELSE IF COGS_US GE 3000  AND COGS_US LT 3300 THEN 5
      ELSE 1 ;
END

GRAPH FILE WF_RETAIL_LITE
SUM LOG1
BY CAT1
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"axisList": {
    "y1": {"bIsLog":true}
   }
*END
ENDSTYLE
END
```

生成されたグラフ出力では、Y 軸に対数目盛りが使用されているため、カテゴリ間の変化の割合が分析しやすくなります。



軸線のフォーマット設定

bodyLineStyle プロパティは、軸線の外観を制御します。軸線は、罫線に直角に交差する線です。軸線は、すべての目盛りを通過し、グラフフレーム の手前 (前面) に描画されます (グラフフレームが表示されている場合)。

構文　　軸線のフォーマット設定

```
"axisname": {
  "bodyLineStyle": {
                "width": number,
    "color": "string",
    "dash": "string"
  }
}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"width": number`

軸線の幅をピクセル単位で定義します。デフォルト値は `1` です。

"color": "string"

色名または数値指定の文字列で軸線の色を定義します。デフォルト値は "transparent" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"dash": "string"

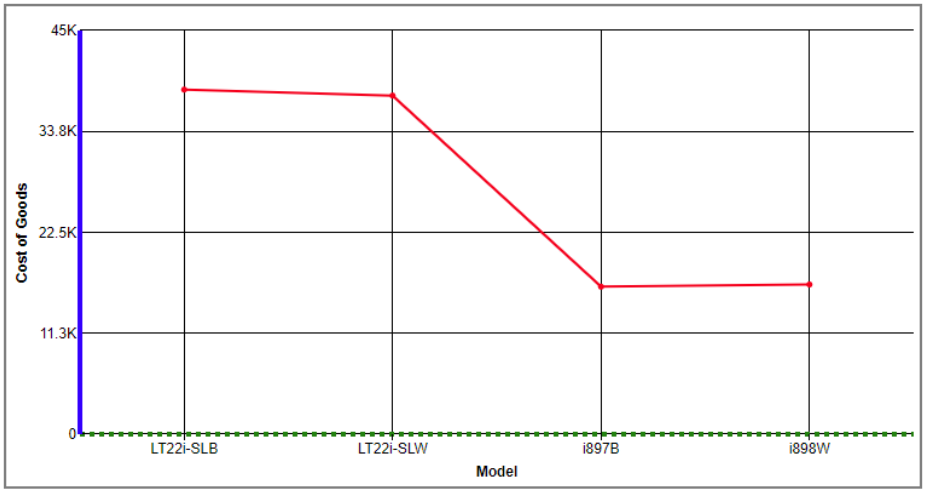
破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　軸線のフォーマット設定

次のリクエストは、縦折れ線グラフを生成し、Y 軸の軸線とベースラインのフォーマットを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "grey"},
"axisList": {
"y1": {
   "majorGrid": {"visible": true},
   "intervalMode": "count",
   "intervalValue": 5,
   "baseLineStyle": {
      "width": 4, "color": "green", "dash": "4 4"},
   "bodyLineStyle": {"width": 4, "color": "blue"}}
   }
*END
ENDSTYLE
END
```

下図の出力結果では、軸線が罫線に直交する青色の線で表示されています。ベースラインは、罫線の下側に緑色の破線で表示されています。



色バンドの定義

colorBand プロパティは、グラフ領域全体に色のブロックを作成します。

構文　　色バンドの定義

```
"axisname":
{
  "colorBands": [
  {
    "start": startval,
    "stop": stopval,
    "color": "color"  }
  ]
},
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"start": startval`

色バンドを開始する軸上の位置を数値または文字列で定義します。

数値軸の場合

- 数値は、軸上に表示される値にする必要があります。
- 文字列は、軸の長さを表すパーセントにする必要があります。文字列は、パーセント記号 (%) を含めて二重引用符 (") で囲みます (例、"50%")。

X 軸の場合

- 数値は、軸の長さをパーセントで表す 0 (ゼロ) から 1 までの値にする必要があります。たとえば、0.5 と指定すると、色バンドがグラフの中央から開始されます。
- 文字列は、軸上に表示されるグループラベルにする必要があります。文字列は二重引用符 (") で囲みます。

`"stop": stopval`

色バンドを終了する軸上の位置を数値または文字列で定義します。

数値軸の場合

- 数値は、軸上に表示される値にする必要があります。
- 文字列は、軸の長さを表すパーセントにする必要があります。文字列は、パーセント記号 (%) を含めて二重引用符 (") で囲みます (例、"50%")。

X 軸の場合

- 数値は、軸の長さをパーセントで表す 0 (ゼロ) から 1 までの値にする必要があります。たとえば、0.5 と指定すると、色バンドがグラフの中央で終了します。
- 文字列は、軸上に表示されるグループラベルにする必要があります。文字列は二重引用符 (") で囲みます。

`"color": "color"`

バンドの色を定義します。次のいずれかの値を指定します。

- 色名または数値指定の文字列で定義する色、あるいは文字列で定義するグラデーション。
- グラデーションを定義する JSON オブジェクト。

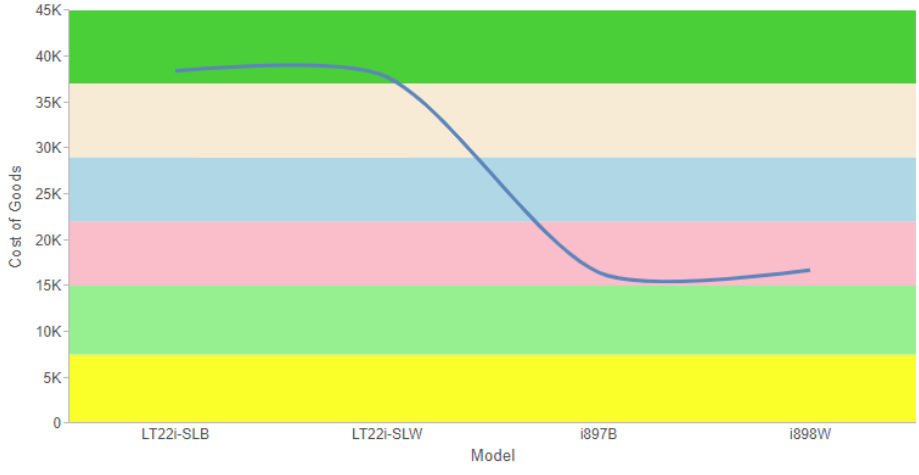
色およびグラデーションの定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　折れ線グラフでの色バンドの定義

次のリクエストでは、Y 軸に色バンドを定義します。Y 軸は数値軸のため、色バンドを定義する数値には、Y 軸に表示される値を指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS_FINAL
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1":
{
 "colorBands": [
  {"start": 0, "stop": 7500, "color": "yellow"},
  {"start": 7500, "stop": 15000, "color": "lightgreen"},
  {"start": 15000, "stop": 22000, "color": "pink"},
  {"start": 22000, "stop": 29000, "color": "lightblue"},
  {"start": 29000, "stop": 37000, "color": "antiquewhite"},
  {"start": 37000, "stop": 45000, "color": "limegreen"}
   ]
}
}
*END
ENDSTYLE
END
```

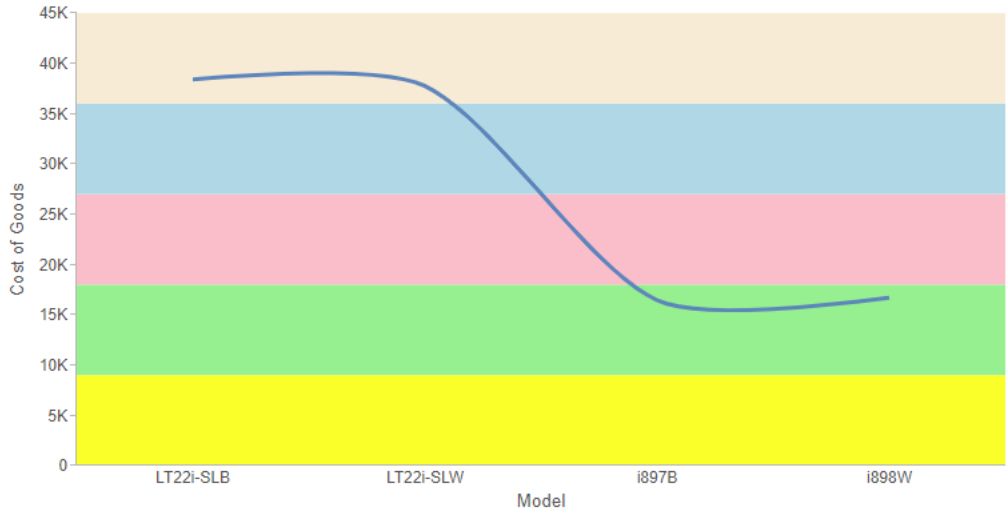
出力結果は次のとおりです。



次の colorBands オブジェクトでは、色バンドの start および stop プロパティをパーセントで指定します。

```
"y1": {
    "colorBands": [
        {"start": 0, "stop": "20%", "color": "yellow"},
        {"start": "20%", "stop": "40%", "color": "lightgreen"},
        {"start": "40%", "stop": "60%", "color": "pink"},
        {"start": "60%", "stop": "80%", "color": "lightblue"},
        {"start": "80%", "stop": "100%", "color": "antiquewhite"}
        ]}}
```

出力結果は次のとおりです。

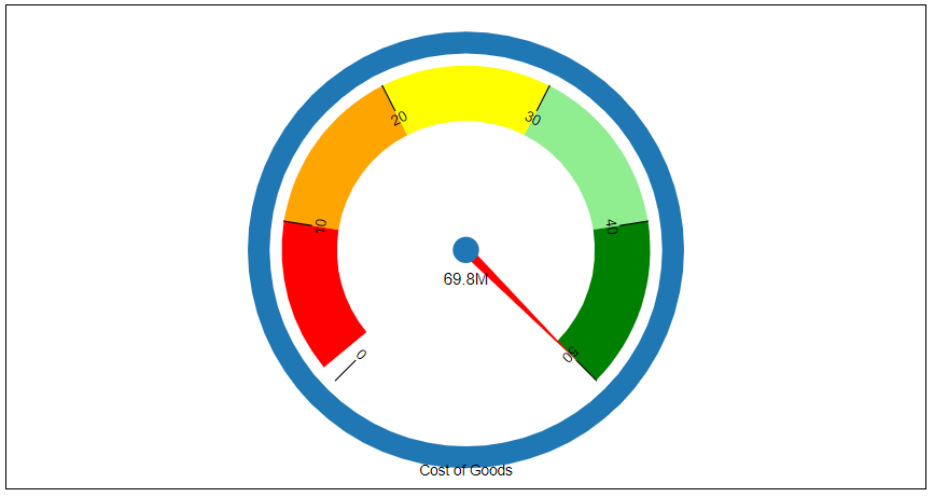


例　メータグラフでの色バンドの定義

次のリクエストは、メータグラフを生成し、メータ軸に色バンドを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=COGS_US, bucket=measure,$
*GRAPH_JS
"yaxis": {
   "min": 0,
   "max": 50,
   "colorBands": [
      {"start": 1, "stop": 10, "color": "red"},
      {"start": 10, "stop": 20, "color": "orange"},
      {"start": 20, "stop": 30, "color": "yellow"},
      {"start": 30, "stop": 40, "color": "lightgreen"},
      {"start": 40, "stop": 50, "color": "green"}
      ]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



主罫線、目盛り、ラベルの個数の制御

数値軸では、intervalMode および intervalValue プロパティを使用して、主罫線、目盛り、ラベルの個数を制御することができます。

構文　主罫線、目盛り、ラベルの個数の制御

```
"axisname": {
  "intervalMode": "string",
  "intervalValue": number}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"intervalMode": "string"`

間隔モードを定義する文字列です。有効な値には、次のものがあります。

- "count" - intervalValue で指定する値が、描画する主罫線の本数であることを示します。
- "interval" - intervalValue で指定する値が、描画する主罫線の間隔であることを示します。
- "skip" - intervalValue で指定する値が、スキップする主罫線の本数であることを示します。

デフォルト値は "undefined" です。この設定では、主罫線の本数および間隔が自動的に計算されます。

```
"intervalValue": number
```

intervalMode で指定したモードに応じて値を設定します。intervalMode が "count" の場合、描画する主罫線の本数を設定します。intervalMode が "interval" の場合、描画する主罫線の間隔を設定します。intervalMode が "skip" の場合、スキップする主罫線の本数を設定します。デフォルト値は "undefined" です。この設定では、主罫線の本数および間隔が自動的に計算されます。

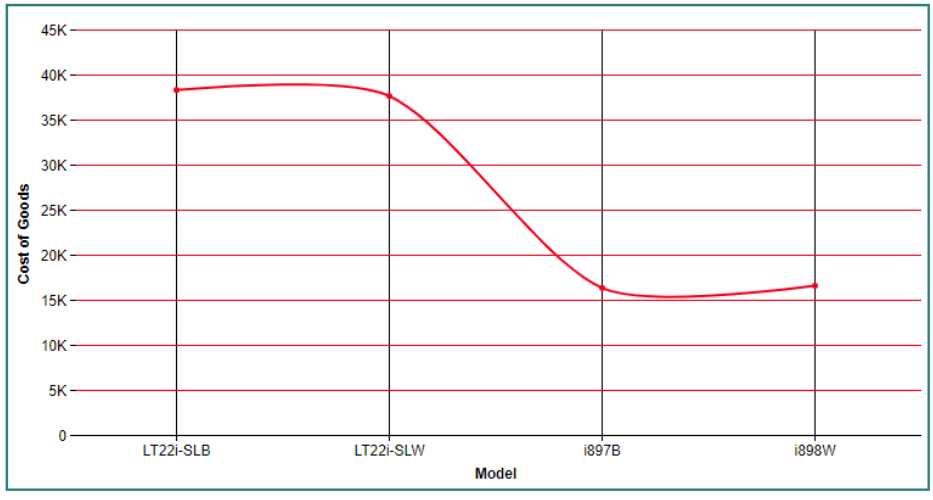
注意：intervalMode および intervalValue の設定に関係なく、最初と最後の主罫線は常に描画されます。

例　intervalMode および intervalValue の設定

次のリクエストは、縦折れ線グラフを生成し、主罫線を自動的に計算します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList":
 {
 "y1": {
    "intervalMode": "undefined",
   "intervalValue": "undefined",
   "majorGrid": {
      "visible": true,
      "aboveRisers": true,
      "lineStyle": {"color": "red"}
      }}}
*END
ENDSTYLE
END
```

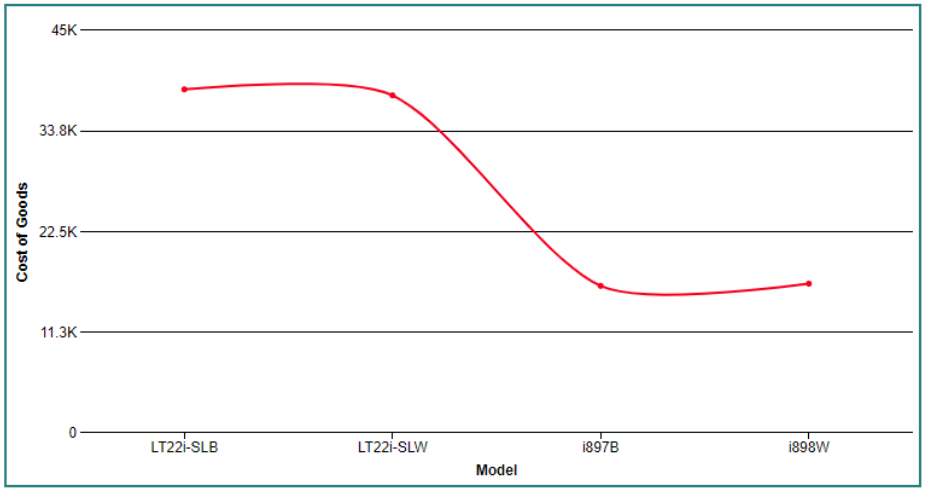
出力結果は次のとおりです。



次のリクエストは、5 本の主罫線を描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {
   "majorGrid": {"visible": true},
   "intervalMode": "count",
   "intervalValue": 5}
*END
ENDSTYLE
END
```

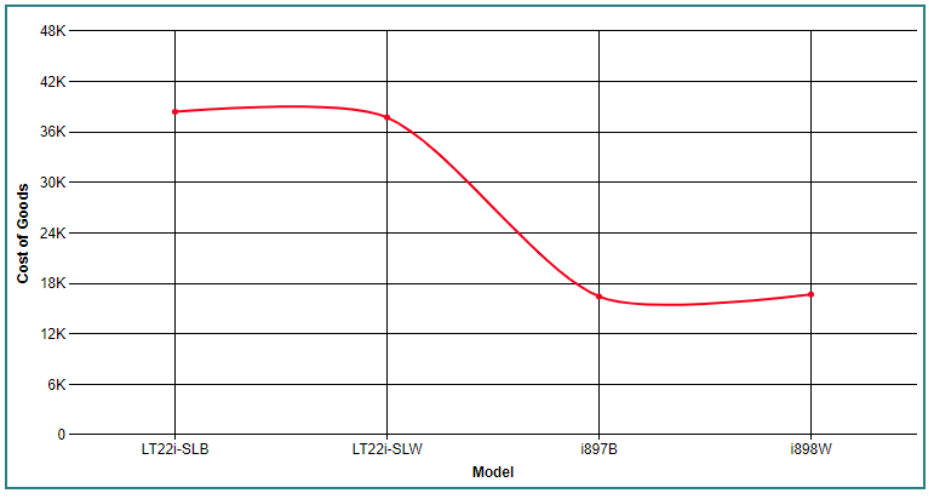
出力結果は次のとおりです。



次のリクエストでは、主罫線を描画する間隔を 6000 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {
   "majorGrid": {"visible": true},
   "intervalMode": "interval",
   "intervalValue": 6000}}
*END
ENDSTYLE
END
```

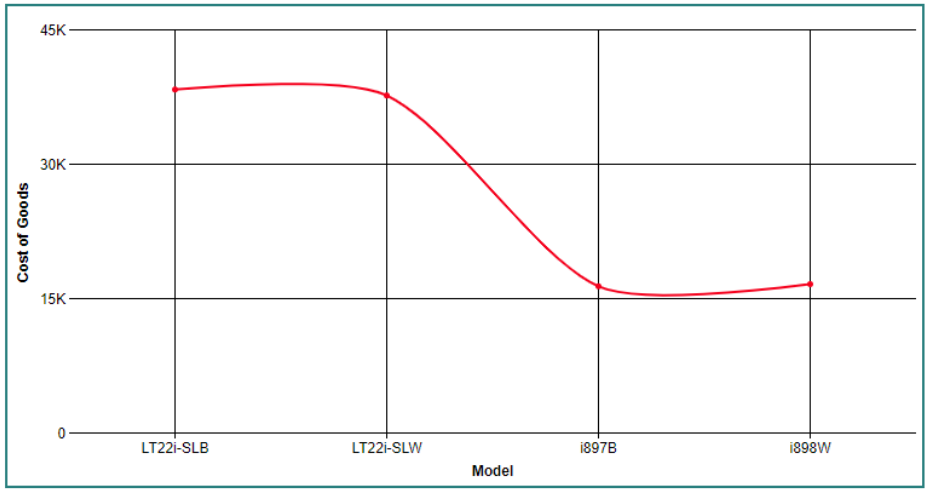
出力結果は次のとおりです。



次のリクエストでは、自動計算で表示される主罫線を交互に省略します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {
   "majorGrid": {"visible": true},
   "intervalMode": "skip",
   "intervalValue": 1
   }}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Y 軸主罫線と Y2 軸主罫線の同期

Y 軸と Y2 軸が存在する場合、デフォルト設定では、各軸で独自の主罫線が生成されます。各軸の目盛りラベル数は、それぞれのデータ値の範囲に応じて異なります。Y2 軸に Y 軸と同数の目盛りラベルを使用するには、Y2 軸の "majorGrid": "lockToY1" プロパティを true に設定します。これにより、Y2 軸と Y 軸で主罫線が共有されます。

構文　　Y 軸主罫線と Y2 軸主罫線の同期

```
"axisname": {
        "majorGrid":
        {
            "lockToY1":boolean
  }
     }
```

説明

`"axisname"`

axisList の y2 または y2axis を指定できます。

`"lockToY1": boolean`

Y2 軸に Y1 軸と同数の目盛りラベルを描画するかどうかを制御します。有効にした場合、Y2 軸と Y1 軸で主罫線が共有されます。有効な値には、次のものがあります。

- false - Y2 軸に Y1 軸と同数の目盛りラベルを使用しません。これがデフォルト値です。

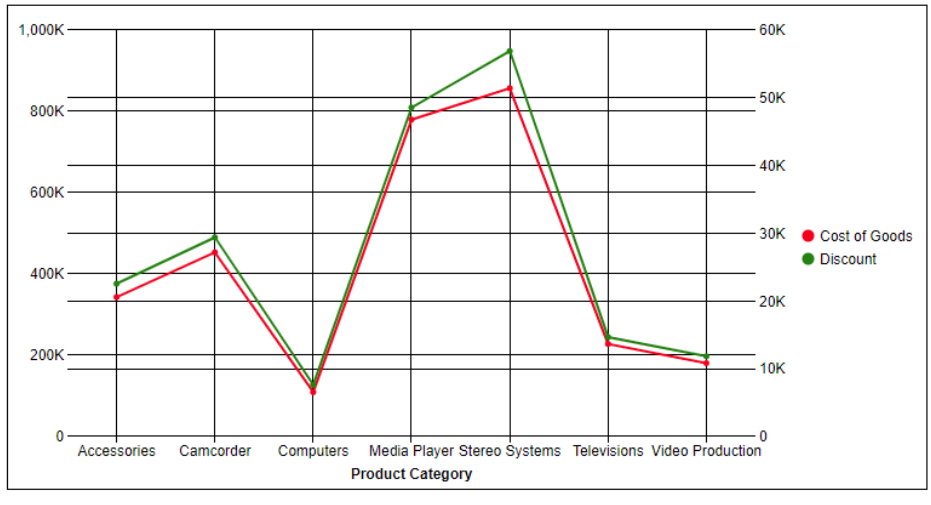
- true - Y2 軸の目盛りラベル数と Y1 軸の目盛りラベル数を同一にします。これにより、Y2 軸と Y1 軸で主罫線が共有されます。

例　Y 軸と Y2 軸での主罫線の共有

次のリクエストは、縦折れ線グラフを生成し、Y1 軸と Y2 軸を表示します。このリクエストでは、各軸に描画する目盛りラベル数は指定されていません。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis(2),$
ENDSTYLE
END
```

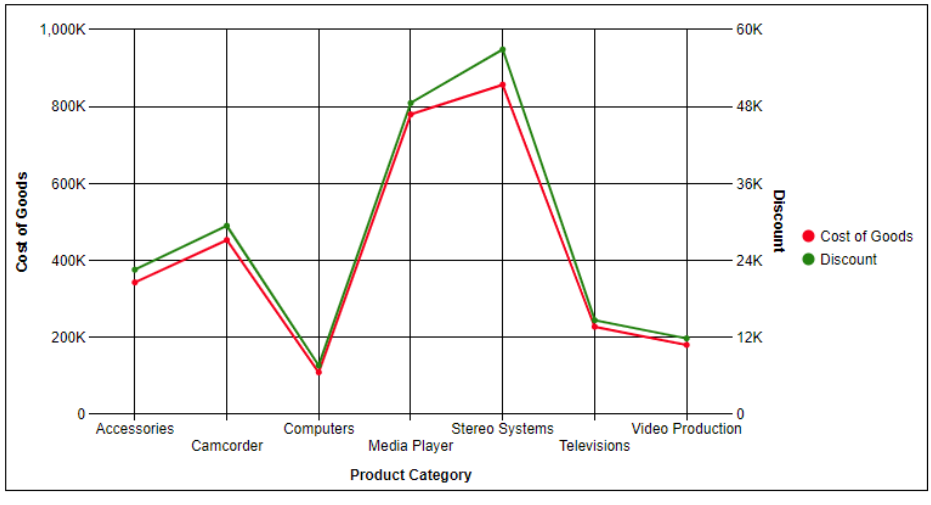
出力結果は次のとおりです。Y 軸と Y2 軸では使用する罫線の数が異なります。



次のリクエストでは、Y2 軸の y2-axis majorGrid オブジェクトで、lockToY1 プロパティが true に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis(2),$
*GRAPH_JS
"axisList":
{
"y2": {"majorGrid": {"visible": true, "lockToY1": true}}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。Y2 軸が調整され、Y 軸と同数の目盛りラベルが使用されています。Y 軸の元の主罫線が、両方の軸で共有されています。



数値軸の方向の制御

invert プロパティは、数値軸の方向 (昇順または降順) を制御します。

構文　数値軸の方向の制御

```
"axisname": {
  "invert": boolean  }
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、y2axis のいずれかにすることもできます。

`"invert": boolean`

軸の方向を降順にするかどうかを指定します。次のいずれかの値を指定します。

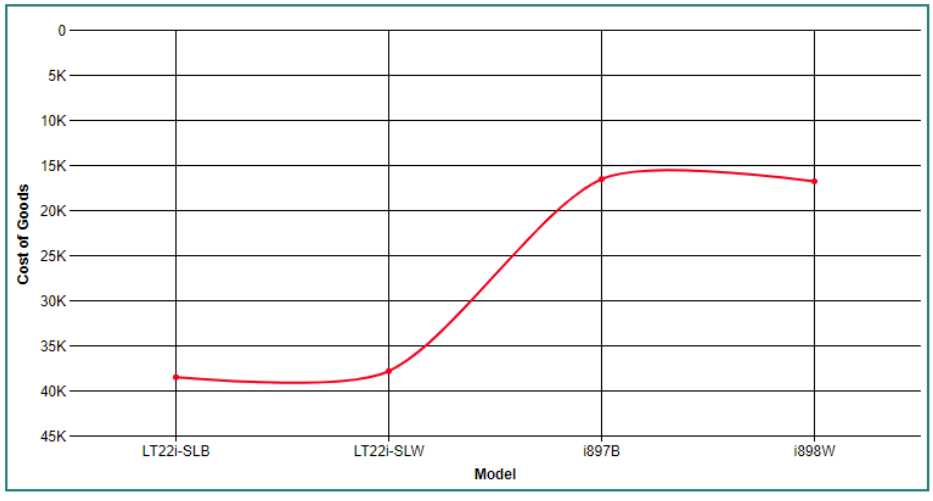
- true - 軸を降順で描画します。
- false - 軸を昇順で描画します。これがデフォルト値です。

例　数値軸の方向の制御

次のリクエストは、Y 軸を降順で描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {
   "invert": true}
   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



軸ラベルのフォーマット設定

labels プロパティは、軸ラベルの表示およびフォーマットを制御します。

注意：HTML5 グラフでは、左右両軸に同時に軸ラベルを表示することはできません (右か左のどちらか一方にしか表示できません)。

構文　　軸ラベルのフォーマット設定

```
"axisname": {
  "labels": {
    "visible": boolean,
    "font": "string",
    "color": "string",
    "excludeMin": boolean,
    "excludeMax": boolean,
    "nestingConcatSymbol": "string",
    "nestingLineStyle": {
            "width": number,
            "color": "string",
            "dash": "string"
             },
    "rotation": number  }
}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、y2axis、zaxis のいずれかにすることもできます。

`"visible": boolean`

軸ラベルの表示を制御します。有効な値には、次のものがあります。

- true - 軸ラベルを表示します。これがデフォルト値です。
- false - 軸ラベルを表示しません。

`"font": "string"`

軸ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "7.5pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列で軸ラベルの色を定義します。デフォルト値は "black" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"excludeMin": boolean`

最小値ラベルの表示を制御します。有効な値には、次のものがあります。

- true - 最小値ラベルを非表示にします。
- false - 最小値ラベルを表示します。これがデフォルト値です。
- "auto" - 最小値ラベルを描画するかどうかをグラフエンジンが制御します。最小値ラベルが別のラベルに接近している場合や別のラベルに重なり合う場合、グラフエンジンは最小値ラベルを描画しません。

`"excludeMax": boolean`

最大値ラベルの表示を制御します。有効な値には、次のものがあります。

- true - 最大値ラベルを非表示にします。
- false - 最大値ラベルを表示します。これがデフォルト値です。
- "auto" - 最大値ラベルを描画するかどうかをグラフエンジンが制御します。最大値ラベルが別のラベルに接近している場合や別のラベルに重なり合う場合、グラフエンジンは最大値ラベルを描画しません。

`"nestingConcatSymbol": "string"`

複数の階層ソートフィールドが X 軸に割り当てられた場合に、ネストされた X 軸ラベルの生成を制御します。

X 軸に複数のソートフィールドを割り当てることで、2 つの異なる効果 (レイアウト) を作成することができます。X 軸は、ソートの階層が X 軸上に表示されるネストされたレイアウトでも、各ラベルに完全修飾され、連結記号で区切られたソートパスが含まれる連結レイアウトでも描画することができます。

このプロパティが設定されていない場合、あるいは null (デフォルト値) または undefined に設定されている場合は、ネストされた X 軸ラベルが生成されます。このプロパティを特定の文字列に設定すると (例、":")、X 軸ラベルが連結され、指定された文字列がソートフィールド値の間に配置されます。

`"nestingLineStyle"`

ネストされたグループを連結する線のスタイルを制御します。

`"width": number`

線幅をピクセル単位で指定します。デフォルト値は 1 です。

`"color": "string"`

線の色を定義する色指定の文字列です。デフォルト値は "black" です。

`"dash": "string"`

破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

`"rotation": number`

軸ラベルの回転角度を定義します。有効な値は、0 (回転なし)、45、90、135、180、270、undefined です。デフォルト値は undefined です。undefined 以外の値を指定すると、軸ラベルの自動レイアウトが無効になります。

注意

- 回転角度 45 度のラベルを選択すると、X 軸スクロールも有効にした場合にラベルが一部切り取られた状態で表示されることがあります。回転角度 90 度のラベルを使用することで、この問題を回避することができます。
- Y 軸ラベルを回転することはできません。

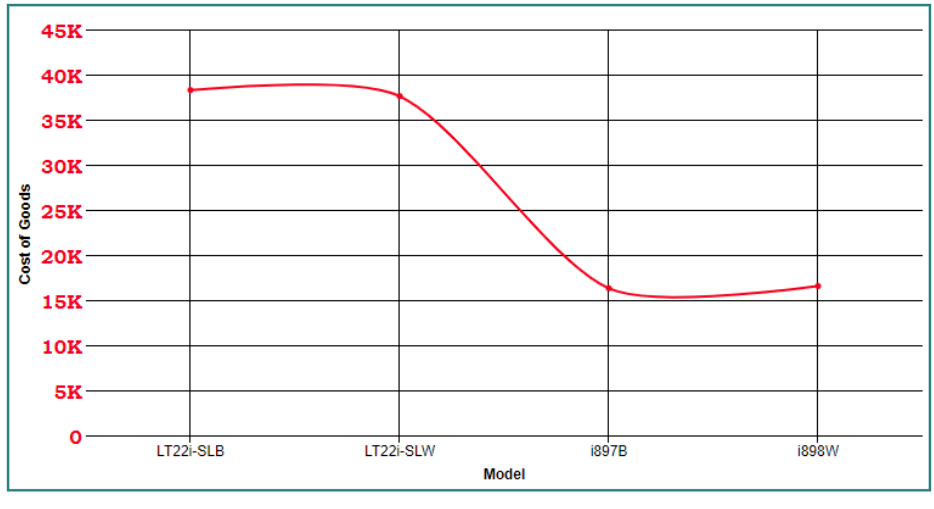
注意 HTML5 グラフでは、X 軸ラベルの自動折り返しはサポートされません。

例　軸ラベルのフォーマット設定

次のリクエストは、縦折れ線グラフを生成し、Y 軸ラベルを太字、赤色、12pt、Bookman Old Style フォントに設定します。

```
GGRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {"labels": {
   "color": "red", "font": "bold 12pt 'Bookman Old Style'"}}}
*END
ENDSTYLE
END
```

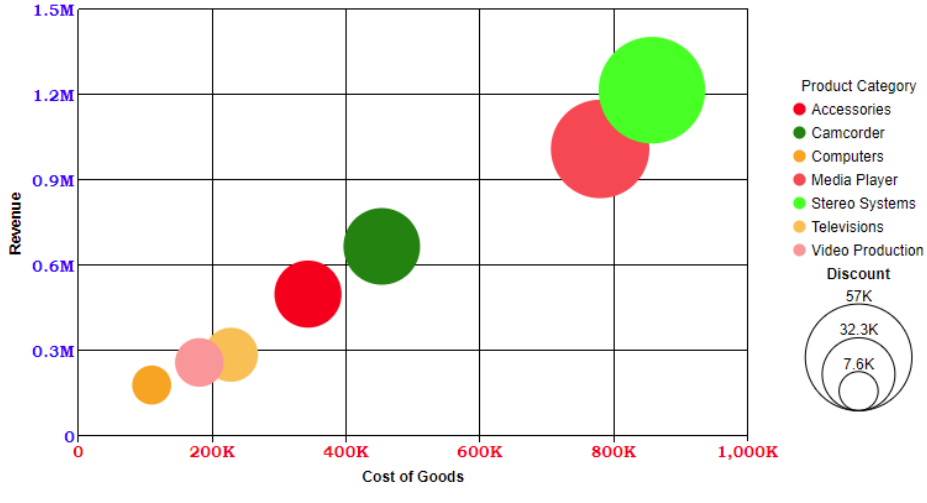
出力結果は次のとおりです。



次のリクエストは、バブルグラフを生成し、軸ラベルのフォーマットを "bold 10pt 'Bookman Old Style'" に設定します。Y 軸ラベルは青色、X 軸ラベルは赤色で表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"border": {"width": 0},
"series": [{"series": "all", "marker": {"shape": "circle"}}],
"axisList": {
"x1": {
   "labels": {"font": "bold 10pt 'Bookman Old Style'", "color": "red"}},
"y1": {
   "labels": {"font": "bold 10pt 'Bookman Old Style'", "color": "blue"}}  }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

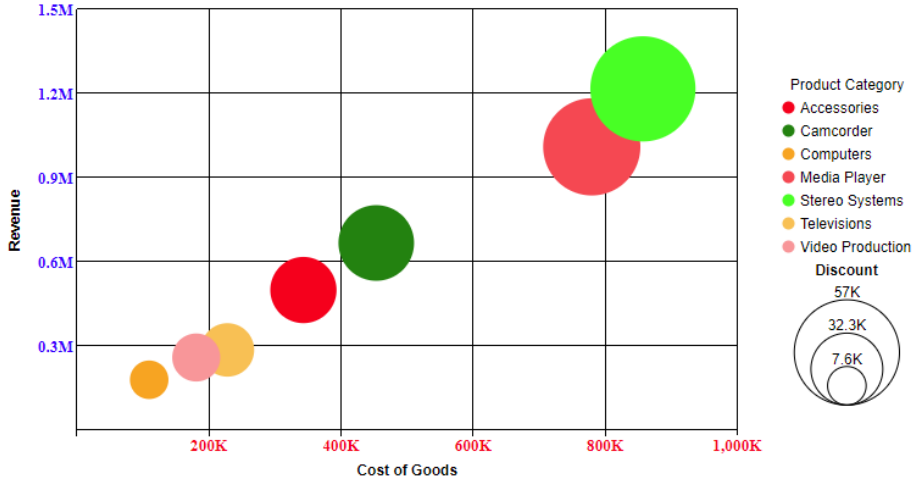


例　軸の最小値ラベルの非表示

次のリクエストは、軸の最小値ラベル (0,0) を非表示にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"border": {"width": 0},
"series": [{"series": "all", "marker": {"shape": "circle"}}],
"axisList": {
"x1": {"labels": {
   "font": "bold 10pt Bookman Old Style",
   "color": "red", "excludeMin": true}},
"y1": {"labels": {
   "font": "bold 10pt Bookman Old Style",
   "color": "blue", "excludeMin": true}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



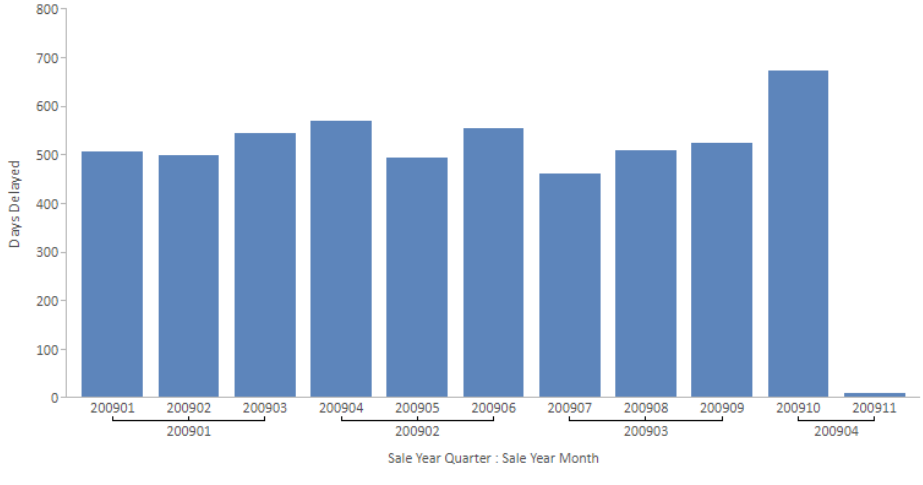
例　X 軸ラベルのネスト制御

X 軸に複数のソートフィールドを割り当てることで、2 つの異なる効果 (レイアウト) を作成することができます。X 軸は、ソートの階層が X 軸上に表示されるネストされたレイアウトでも、各ラベルに完全修飾され、連結記号で区切られたソートパスが含まれる連結レイアウトでも描画することができます。

次のリクエストは、ネストされた X 軸ラベルを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY TIME_YEARQTR
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=DAYSDELAYED, BUCKET=y-axis, $
TYPE=DATA, COLUMN=TIME_YEARQTR, BUCKET=x-axis, $
TYPE=DATA, COLUMN=TIME_YEARMTH, BUCKET=x-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"axisList": {
"x1": {"labels": {"nestingConcatSymbol": null}}}
*END
ENDSTYLE
END
```

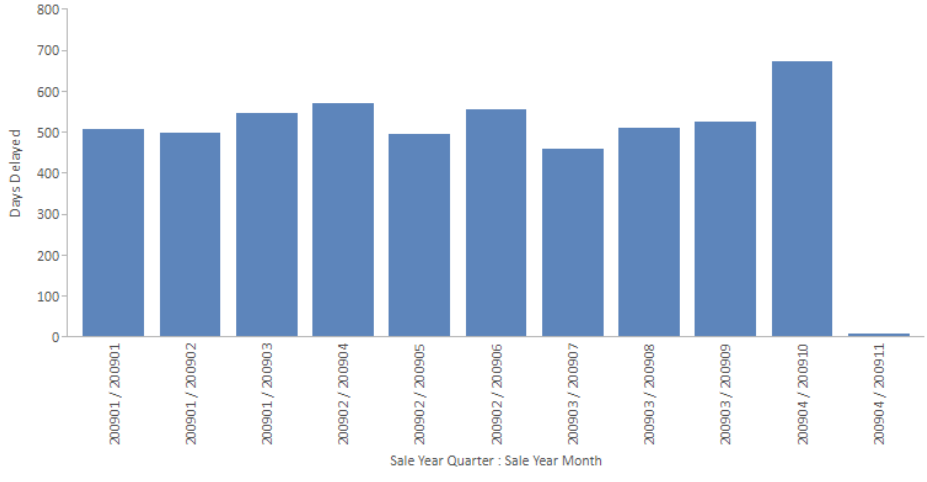
出力結果は次のとおりです。



次のリクエストでは、連結文字列として両端にブランクが追加されたスラッシュ (" / ") を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY TIME_YEARQTR
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=DAYSDELAYED, BUCKET=y-axis, $
TYPE=DATA, COLUMN=TIME_YEARQTR, BUCKET=x-axis, $
TYPE=DATA, COLUMN=TIME_YEARMTH, BUCKET=x-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"axisList": {
"x1": {"labels": {"nestingConcatSymbol": " / "}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

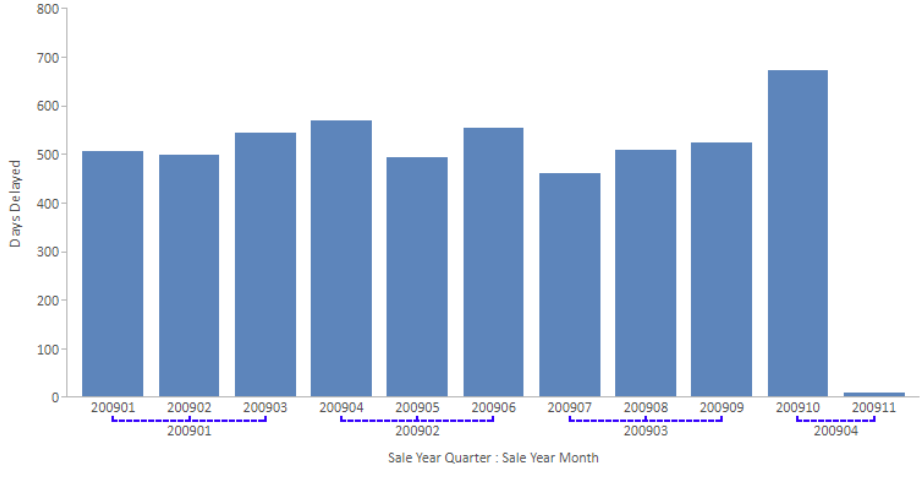


例　ネストされたラベルの接続線スタイル

次のリクエストは、ネストされたラベルの接続線の幅を 2 ピクセル、青色、破線に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY TIME_YEARQTR
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=DAYSDELAYED, BUCKET=y-axis, $
TYPE=DATA, COLUMN=TIME_YEARQTR, BUCKET=x-axis, $
TYPE=DATA, COLUMN=TIME_YEARMTH, BUCKET=x-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
*GRAPH_JS
"axisList": {
 "x1": {"labels": {"nestingConcatSymbol": null,
 "nestingLineStyle": {
         "width": 2,
         "color": "blue",
         "dash": "3 3"
        }}}
        }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



主罫線と目盛りのフォーマット設定

majorGrid プロパティは、主罫線と目盛りの表示およびフォーマットを制御します。

構文　主罫線と目盛りのフォーマット設定

```
"axisname": {
"majorGrid": {
  "visible": boolean,
  "aboveRisers": boolean,
  "lineStyle": {
    "width": number,
    "color": "string",
    "dash": "string"
   },
  "ticks": {
    "length": number,
    "visible": boolean,
    "style": "string",
    "lineStyle": {
      "width": number,
      "color": "string"}
    }
  }
}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"majorGrid":`

主罫線のプロパティを定義します。

`"visible": boolean`

主罫線の表示を制御します。デフォルト値は、生成するグラフタイプに応じて異なります。有効な値には、次のものがあります。

- true - 主罫線を描画します。
- false - 主罫線を描画しません。

`"aboveRisers": boolean`

罫線をライザの手前に描画するか、背後に描画するかを制御します。有効な値には、次のものがあります。

- true - 主罫線をライザの手前に描画します。

- ❏ false - 主罫線をライザの背後に描画します。これがデフォルト値です。

"lineStyle":
主罫線のスタイルを定義します。

"width": number

主罫線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

色名または数値指定の文字列で主罫線の色を定義します。デフォルト値は "rgba(255, 255, 255, 0.3)" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"dash": "string"

破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"ticks":
主罫線の目盛りを定義します。

"length": number

目盛りの長さをピクセル単位で定義します。デフォルト値は 5 です。

"visible": boolean
目盛りの表示を制御します。有効な値は、次のとおりです。

- ❏ true - 目盛りを描画します。これがデフォルト値です。
- ❏ false - 目盛りを描画しません。

"style": "string"
目盛りスタイルを定義する文字列です。デフォルト値は、生成するグラフタイプに応じて異なります。有効な値には、次のものがあります。

- ❏ "inner" - 目盛りを主罫線の内側に配置します。これがデフォルト値です。
- ❏ "outer" - 目盛りを主罫線の外側に配置します。
- ❏ "span" - 目盛りの半分を主罫線の内側に、半分を主罫線の外側に表示します。

"lineStyle":
主罫線目盛りの線スタイルを定義します。

"width": number

目盛りの幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

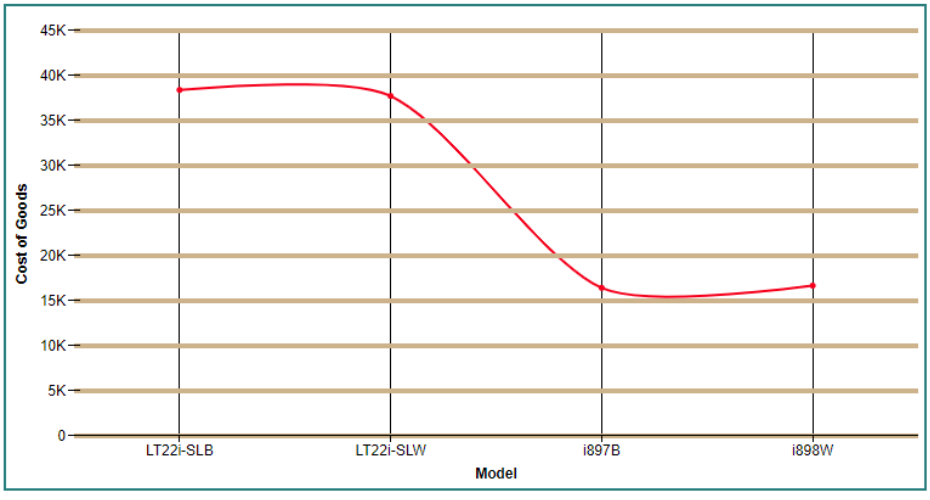
色名または数値指定の文字列で目盛りの色を定義します。デフォルト値は "black" です。

例　主罫線と目盛りのフォーマット設定

次のリクエストは、主罫線をライザの手前に表示し、4 ピクセル幅、黄褐色に設定します。目盛りスタイルが "span" に設定されているため、目盛りは主罫線の両側に表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {
   "title": {"visible": true},
   "majorGrid": {
      "visible": true,
      "aboveRisers": true,
      "lineStyle": {"width": 4, "color": "tan"},
      "ticks": {"style": "span"}
      }}}
*END
ENDSTYLE
END
```

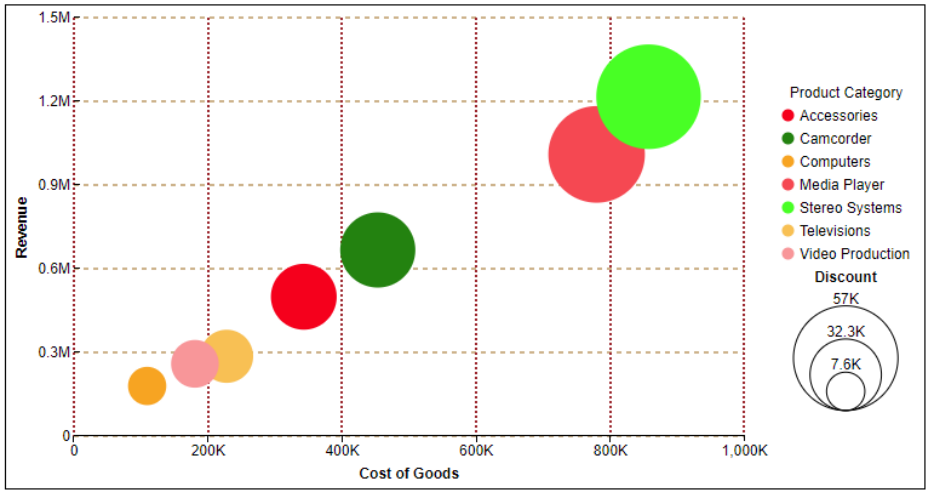
出力結果は次のとおりです。



次のリクエストは、バブルグラフを生成し、X 軸および Y 軸の主罫線のフォーマットを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"series": [{"series": "all", "marker": {"shape": "circle"}}],
"axisList": {
"x1": {
   "title": {"visible": true},
   "majorGrid": {"lineStyle": {
      "width": 2, "color": "brown", "dash": "2 2"}}},
"y1": {
   "title": {"visible": true},
   "majorGrid": {"lineStyle": {
      "width": 2, "color": "tan", "dash": "4 4"}}}}
*END
ENDSTYLE
END
```

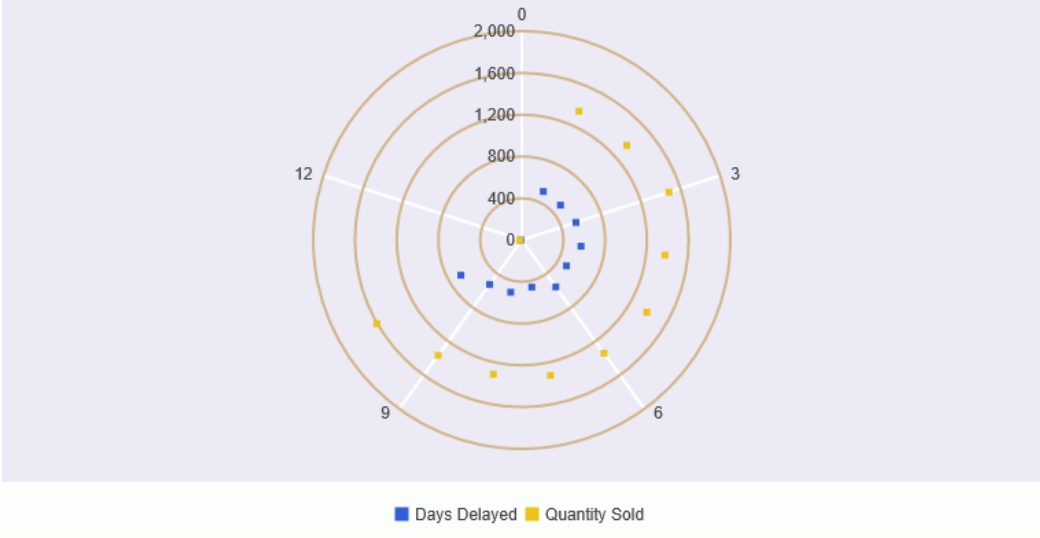
出力結果は次のとおりです。



次のリクエストは、極グラフを生成し、Y 軸の主罫線を黄褐色、X 軸の主罫線を白色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
ACROSS TIME_MTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH POLAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"polarProperties": {
    "straightGridLines": false, "extrudeAxisLabels": false},
"axisList":
"x1": {
    "title": {"visible": true},
    "majorGrid": {
        "aboveRisers": false,
        "lineStyle": {"width": 2, "color": "white"}}},
"y1": {
    "title": {"visible": true},
    "majorGrid": {
        "aboveRisers": false,
        "lineStyle": {"width": 2, "color": "tan"}}}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

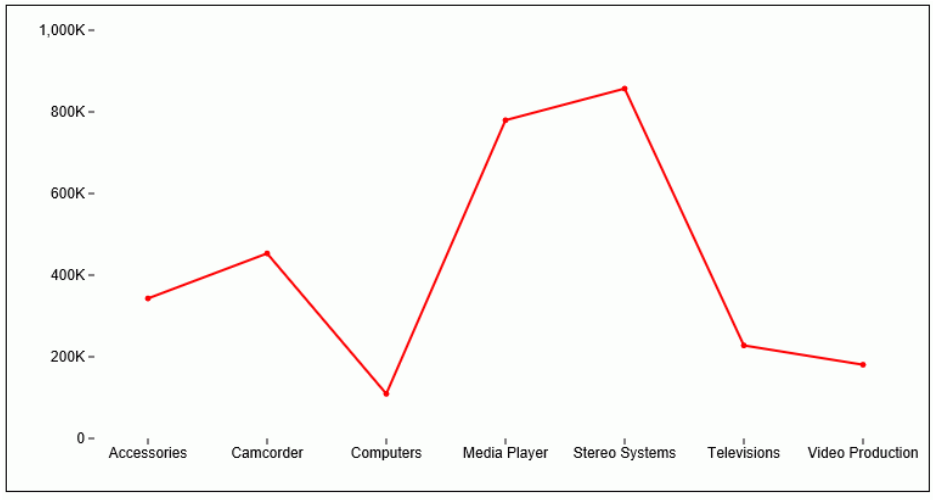


例　軸線の削除

"majorGrid":{"visible": false} プロパティを使用して、指定した軸の軸線およびすべての罫線を削除することができます。数値軸の場合、"baseLineStyle": {"width": 0} プロパティを使用してベースラインも削除する必要があります。以下はその例です。

```
GRAPH FILE wf_retail_lite
SUM COGS_US AS ' '
BY PRODUCT_CATEGORY AS ' '
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis,$
*GRAPH_JS
"yaxis":{"baseLineStyle": {"width": 0},
         "majorGrid": {"visible":false}},
"xaxis":{"majorGrid": {"visible":false}}
*END
ENDSTYLE
 END
```

出力結果は次のとおりです。



数値軸での最小値および最大値の定義

数値軸では、min および max プロパティを使用して、軸に描画する最小値および最大値を定義することができます。

構文　数値軸での最小値および最大値の定義

```
"axisname": {
  "min": number,
  "max": number}
```

説明

`axisname`

axisList の軸にすることも、xaxis、yaxis のいずれかにすることもできます。

`"min": number`

数値軸に描画する最小値です。デフォルト値は undefined です。この場合、データセット内の値に基づいて最小値が自動的に描画されます。

`"max": number`

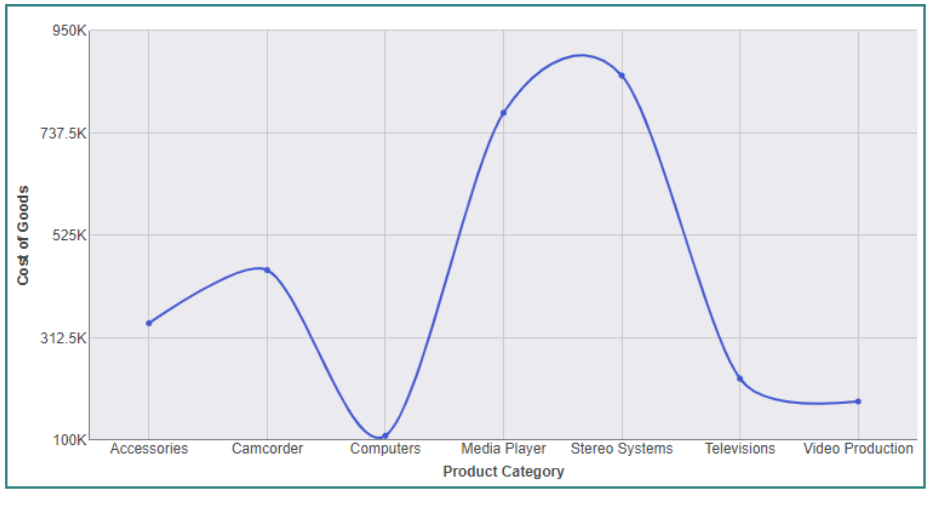
数値軸に描画する最大値です。デフォルト値は undefined です。この場合、データセット内の値に基づいて最大値が自動的に描画されます。

例　数値軸に表示する最小値および最大値の指定

次のリクエストは、縦折れ線グラフを生成し、Y 軸に表示する最小値を 100,000、最大値を 950,000 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {"min": 100000, "max": 950000}}
*END
ENDSTYLE
END
```

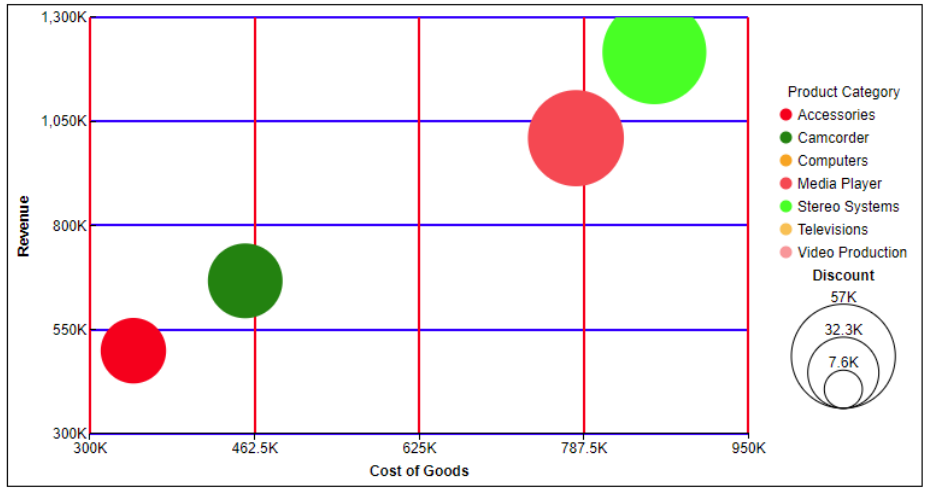
出力結果は次のとおりです。



次のリクエストは、バブルグラフを生成し、各軸に表示する最大値および最小値を設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"series": [{"series": "all", "marker": {"shape": "circle"}}],

"xaxis": {
    "min": 300000, "max": 950000,
    "majorGrid": {"lineStyle": {
      "width": 2, "color": "red", "dash": ""}}},
"yaxis": {
    "min": 300000, "max": 1300000,
    "majorGrid": {"lineStyle": {
      "width": 2, "color": "blue", "dash": ""}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



複数 Y 軸および分割 Y 軸グラフの最大値と最小値の同期

軸の最小値および最大値に "autoGlobal" プロパティを使用することで、複数 Y 軸および分割 Y 軸グラフの最小値と最大値を同期することができます。最小値と最大値がともに複数の Y 軸で同期されている場合、各軸の目盛りも同期されます。

構文　　複数 Y 軸および分割 Y 軸グラフの最大値と最小値の同期

```
"yaxis":
{
    "min": "autoGlobal",
    "max": "autoGlobal"
},
"y2axis":
{
    "min": "autoGlobal",
    "max": "autoGlobal"
}
```

または

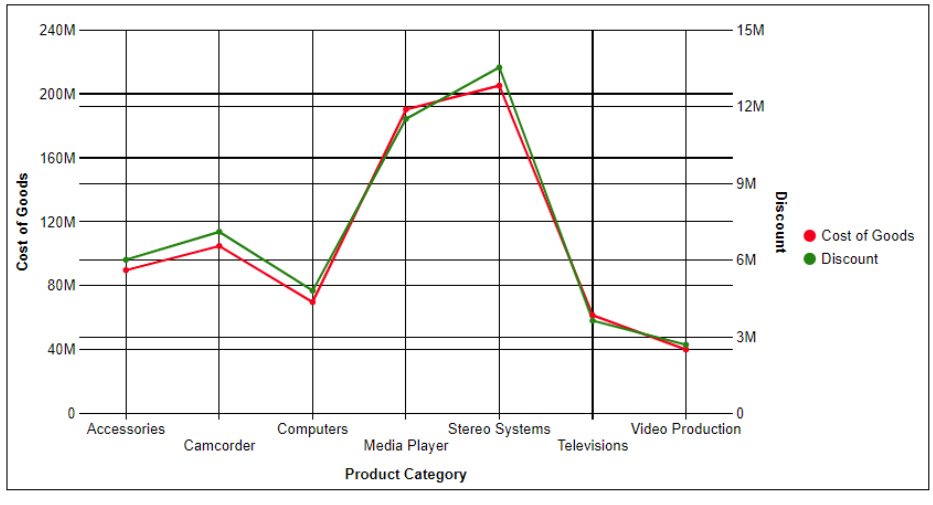
```
"axisList":
{
  "y1":
  {
    "min": "autoGlobal",
    "max": "autoGlobal"
  },
  "y2":
  {
    "min": "autoGlobal",
    "max": "autoGlobal"
  },
  ...
}
```

例　複数Y軸グラフでの目盛りの同期

次のリクエストは、軸の最小値と最大値を同期せずに、複数 Y 軸の折れ線グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis(2),$
ENDSTYLE
END
```

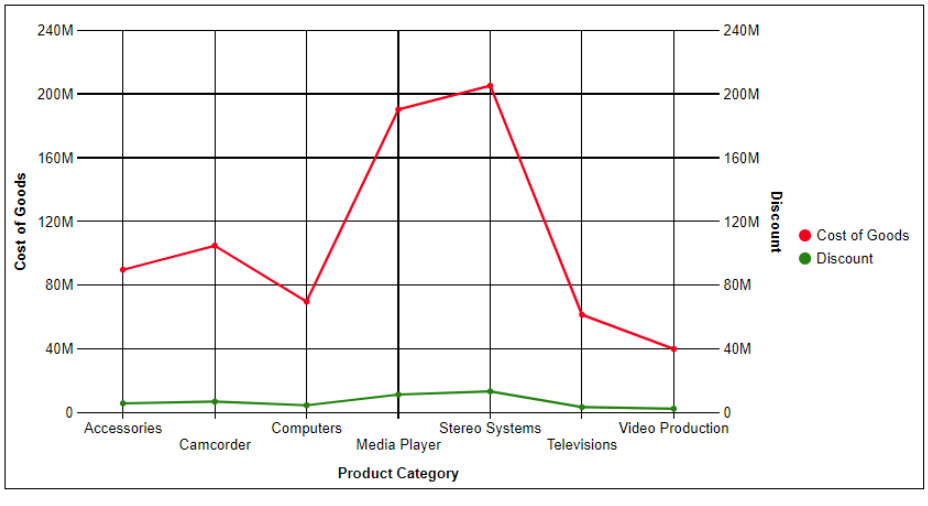
出力結果は次のとおりです。



次のリクエストでは、両方の Y 軸の最大値を同期します。最小値は、autoGlobal プロパティを設定しなくても等しいため、このグラフに autoGlobal プロパティを設定する必要はありません。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis(2),$
*GRAPH_JS
"axisList":
{
"y1": {
    "max": "autoGlobal"
    },
 "y2": {
     "max": "autoGlobal"
     }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



数値軸の最小値ラベルと最大値ラベルの置換

軸のプロパティ replaceMin および replaceMax を使用して、数値軸の最小値ポイントと最大値ポイントのラベルを置き換えることができます。

構文　数値軸の最小値ラベルと最大値ラベルの置換

```
"axisname":
 {

    "labels": {
               "replaceMin": "string",
               "replaceMax": "string"
           }
 }
```

説明

`axisname`
axisList の数値軸の名前、または xaxis、yaxis のいずれかです。

`"replaceMin": "string"`
任意の文字です。デフォルト値は undefined です。自動計算された最小値ラベルを使用します。

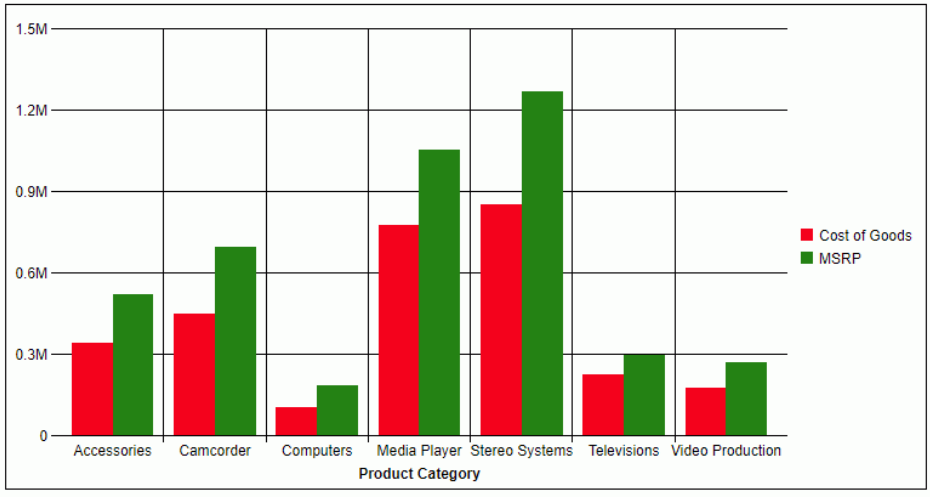
`"replaceMax": "string"`
任意の文字です。デフォルト値は undefined です。自動計算された最大値ラベルを使用します。

例　Y 軸の最小値ラベルと最大値ラベルの置換

次のリクエストは、棒グラフを生成し、Y 軸にデフォルト設定のラベル値を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
ENDSTYLE
END
```

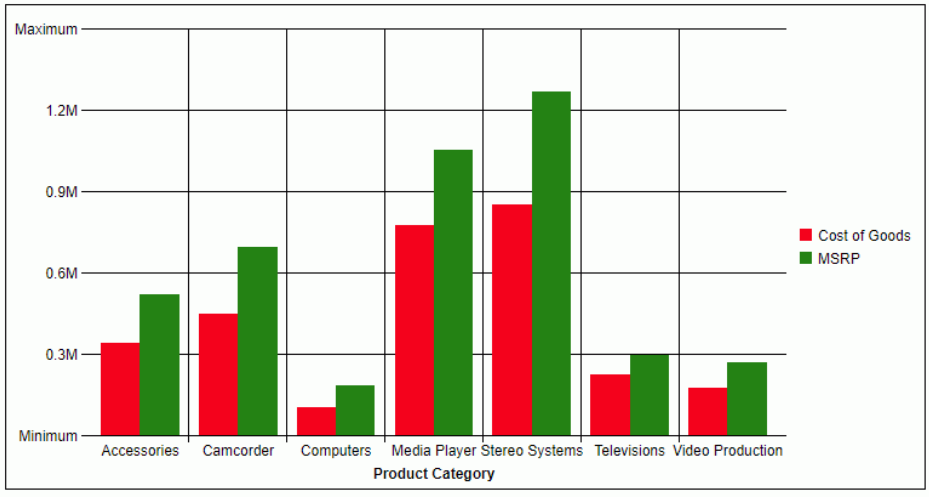
出力結果は次のとおりです。



次のリクエストでは、Y 軸の最小値ラベルを「Minimum」という単語に置き換え、Y 軸の最大値ラベルを「Maximum」という単語に置き換えます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"axisList": {
"y1":
  {"labels":
    {
     "replaceMin": "Minimum",
     "replaceMax": "Maximum"
     }
  }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



棒グラフの数値軸のゼロ値の表示

棒グラフまたは積み上げ棒グラフでは、0 (ゼロ) に近い値がデータに含まれない場合、数値軸にゼロ値は表示されません。このために、出力が正確に解釈しにくい場合があります。軸プロパティの mustIncludeZero を使用することで、軸上にゼロ値を表示することができます。

構文　棒グラフの数値軸のゼロ値の表示

```
"axisname": {"mustIncludeZero": boolean}
```

説明

`"axisname"`
axisList の数値軸の名前、または xaxis、yaxis のいずれかです。

`"mustIncludeZero": boolean`
次のいずれかの値です。

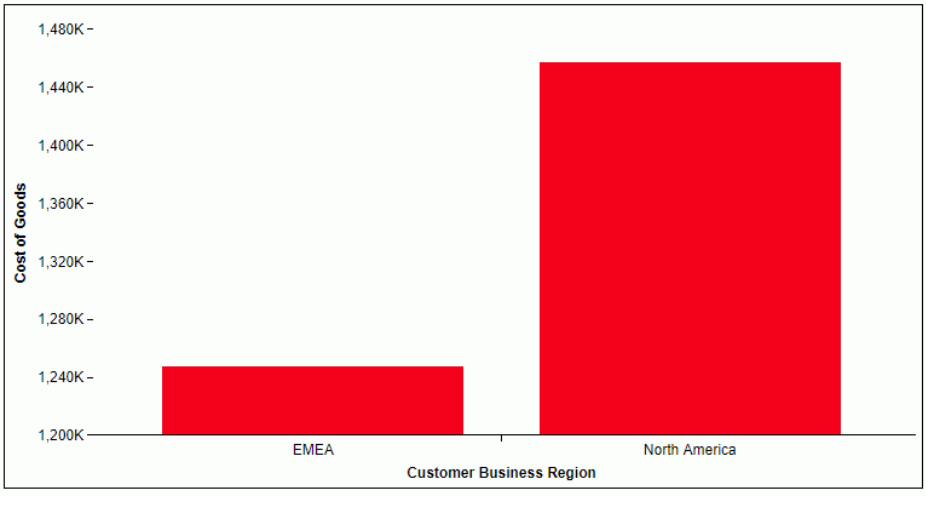
- true - 数値軸にゼロ値を表示します。これがデフォルト値です。
- false - 数値軸にゼロ値を表示しません。

例　数値軸へのゼロ値の追加

次のリクエストは、棒グラフを生成し、Y 軸の mustIncludeZero プロパティを false に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania' OR 'South America'
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=y-axis,$
type=data, column= business_region, bucket=x-axis,$
*GRAPH_JS
"axisList": {
"x1": {
    "majorGrid": {"visible": false}
},
"y1": {
    "majorGrid": {"visible": false},
    "mustIncludeZero": false
}
}
END
```

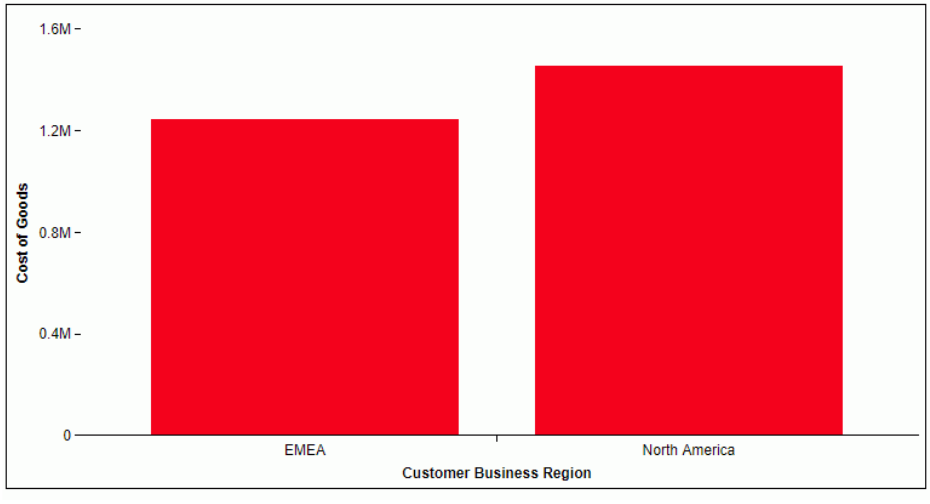
出力結果は次のとおりです。下図のように、Y 軸の最小値はゼロより大きい数値になっています。



次のリクエストでは、Y 軸の mustIncludeZero プロパティを true に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania' OR 'South America'
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=y-axis,$
type=data, column= business_region, bucket=x-axis,$
*GRAPH_JS
"axisList": {
"x1": {
    "majorGrid": {"visible": false}
},
"y1": {
    "majorGrid": {"visible": false},
    "mustIncludeZero": true
}
}
END
```

下図のように、Y 軸の目盛りに 0 (ゼロ) が追加されています。



補助罫線と目盛りのフォーマット設定

minorGrid プロパティは、補助罫線と目盛りの表示およびフォーマットを制御します。

構文　　補助罫線と目盛りのフォーマット設定

```
"axisname": {
"minorGrid": {
  "visible": boolean,
  "count": number,
  "lineStyle": {
    "width": number,
    "color": "string",
    "dash": "string"
   },
    "ticks": {
      "length": number,
      "visible": boolean,
      "style": "string",
      "lineStyle": {
        "width": number,
        "color": "string"}
    }
  }
}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"minorGrid":`

補助罫線のプロパティを定義します。

`"visible": boolean`

補助罫線の表示を制御します。有効な値には、次のものがあります。

- true - 補助罫線を描画します。
- false - 補助罫線を描画しません。これがデフォルト値です。

`"count": number`

主罫線間に描画する補助罫線の本数です。デフォルト値は undefined です。

`"lineStyle":`

補助罫線の線スタイルを定義します。

`"width": number`

補助罫線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

色名または数値指定の文字列で補助罫線の色を定義します。デフォルト値は "black" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"dash": "string"

破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"ticks":

補助罫線目盛りのプロパティを定義します。

"length": number

目盛りの長さをピクセル単位で定義します。デフォルト値は 5 です。

"visible": boolean

目盛りの表示を制御します。有効な値は、次のとおりです。

- true - 目盛りを描画します。
- false - 目盛りを描画しません。これがデフォルト値です。

"style": "string"

目盛りスタイルを定義する文字列です。有効な値には、次のものがあります。

- "inner" - 目盛りを補助罫線の内側に配置します。これがデフォルト値です。
- "outer" - 目盛りを補助罫線の外側に配置します。
- "span" - 目盛りの半分を補助罫線の内側に、半分を補助罫線の外側に表示します。

"lineStyle":

補助罫線目盛りの線スタイルを定義します。

"width": number

目盛りの幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

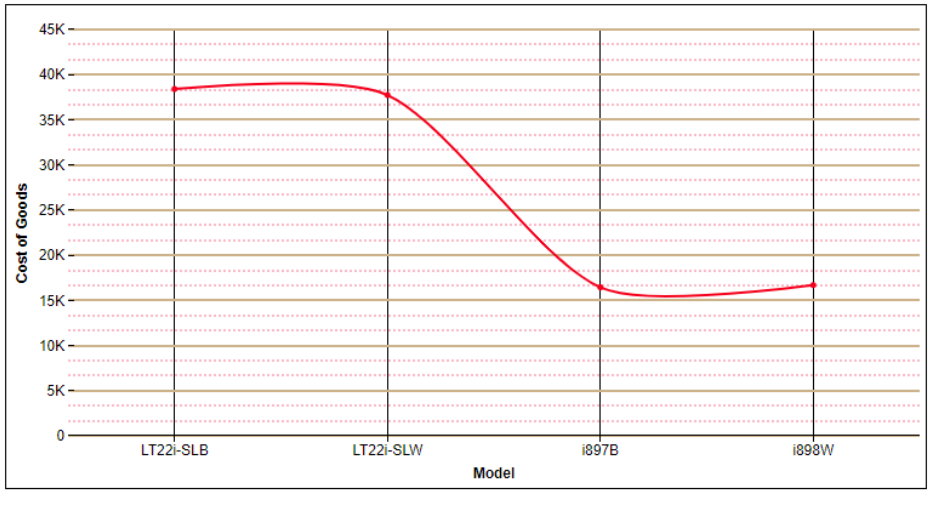
色名または数値指定の文字列で目盛りの色を定義します。デフォルト値は "black" です。

例　補助罫線と目盛りのフォーマット設定

次のリクエストは、Y1 軸の主罫線を黄褐色の実線、補助罫線をピンク色の破線に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY MODEL
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {"lineConnection": "curved"},
"axisList": {
"y1": {
   "majorGrid": {"visible": true, "lineStyle": {
      "width": 2, "color": "tan"}},
   "minorGrid": {"visible": true, "lineStyle": {
      "width": 2, "color": "pink", "dash": "2 2"}
   }}}
*END
ENDSTYLE
END
```

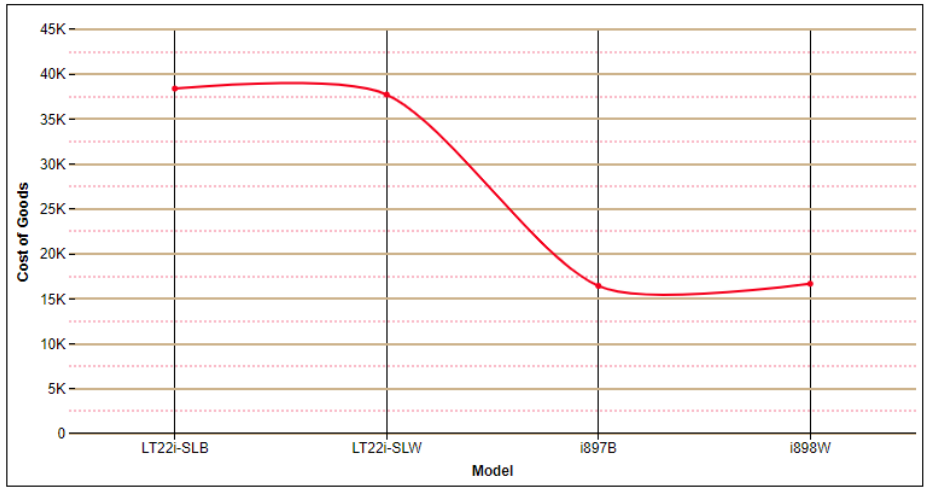
出力結果は次のとおりです。



主罫線間には、デフォルト設定で 2 本の補助罫線が描画されています。主罫線間の補助罫線数を 1 本にするには、Y1 軸のプロパティを次のように変更します。

```
"y1": {
"majorGrid": {"visible": true, "lineStyle": {
        "width": 2, "color": "tan"}},
"minorGrid": {"visible": true, "count": 1, "lineStyle": {
        "width": 2, "color": "pink", "dash": "2 2"}}}
```

出力結果は次のとおりです。



X 軸のラベルレイアウトの制御

"x-xaxis": "LabelLayout" プロパティは、X 軸のスクロールを有効にするか、またラベルを交互に表示するか、交互に省略するか、末尾を切り捨てるかを制御します。

構文　X 軸のラベルレイアウトの制御

```
"axisname": {
         "labelLayout": {
             "stagger": boolean,
             "skip": value,
             "truncate": boolean                }
      }
```

説明

"axisname"
　axisList の X 軸または xaxis です。

`"labelLayout":`
X 軸のラベルレイアウトを定義します。

`"stagger": boolean`
ラベルを交互に表示するかどうかを制御します。

注意：一般には、X 軸のラベルのみ交互に表示することができます。

有効な値には、次のものがあります。

- "auto" - この設定では、グラフエンジンによって自動決定されます。これがデフォルト値です。
- true - ラベルのスペースが足りない場合は、ラベルを交互に表示します。
- false - ラベルを交互に表示しません。

`"skip": boolean, "skip": number`
ラベルを交互に省略するかどうかを制御します。有効な値には次のものがあります。

- "auto" - この設定では、グラフエンジンによって自動決定されます。これがデフォルト値です。
- true - ラベルのスペースが足りない場合は、ラベルを交互に省略します。
- false - ラベルを交互に省略しません。
- 0 (ゼロ) より大きい数値 - 省略するラベル数です。

`"truncate": boolean`
ラベルの末尾を切り捨てるかどうかを制御します。有効な値には次のものがあります。

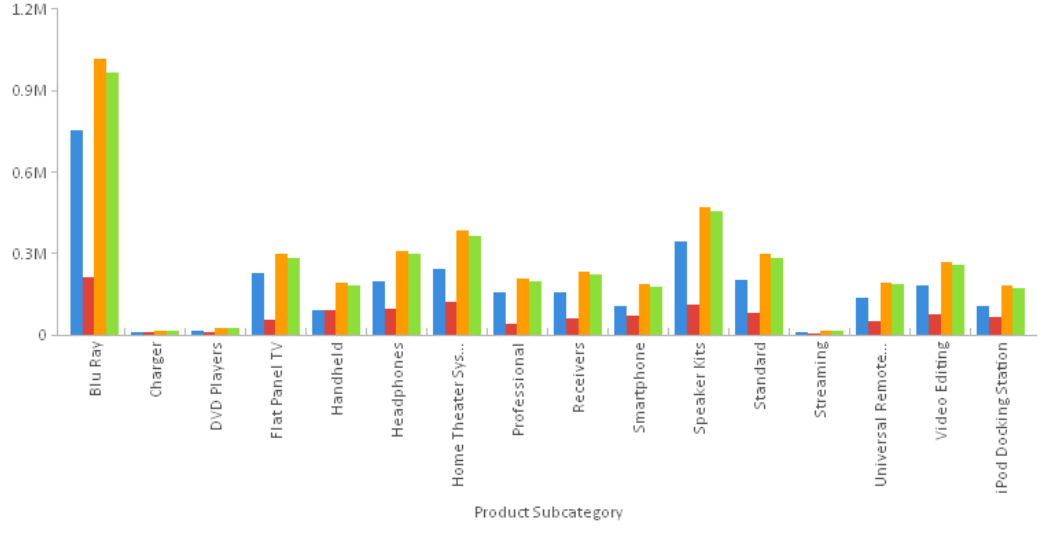
- "auto" - この設定では、グラフエンジンによって自動決定されます。これがデフォルト値です。
- true - ラベルのスペースが足りない場合は、ラベルの末尾を切り捨てます。
- false - ラベルの末尾を切り捨てません。

例　　X 軸のラベルレイアウトの制御

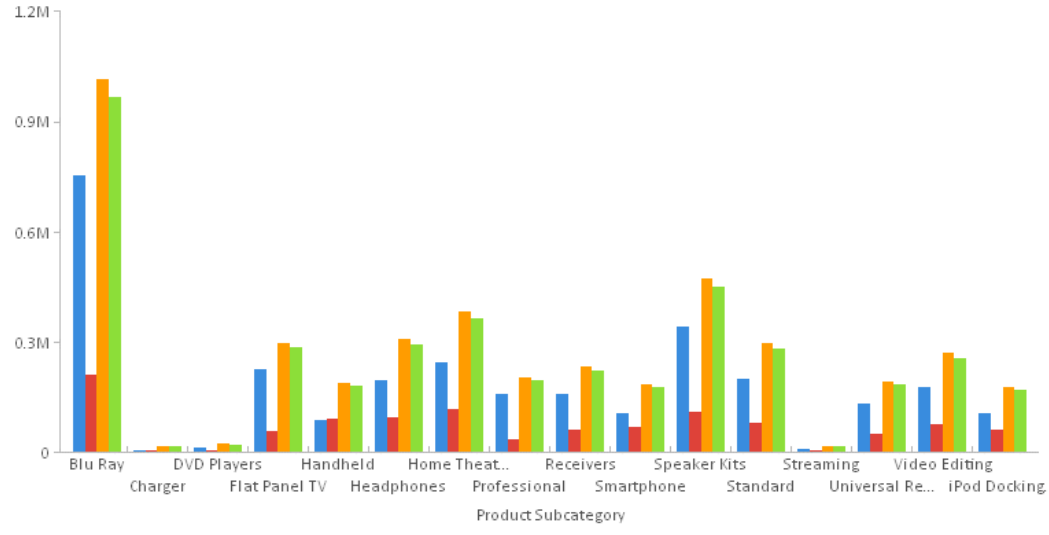
次のリクエストでは、デフォルト設定の X 軸ラベルレイアウトを使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US MSRP_US REVENUE_US
BY PRODUCT_SUBCATEG
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"xaxis": {"labelLayout": {
    "stagger": "auto",
    "skip": "auto",
    "truncate": "auto"
    }}
*END
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N5, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



stagger プロパティを true に変更すると、下図のグラフが生成されます。



X 軸スクロールのフォーマット設定

"xaxis": "groupFit" プロパティは、X 軸のスクロールを有効または無効にします。スクロールを有効にした場合でも、必要な場合にのみスクロールが実装されます。スクロールバーの形式には、次の 2 種類があります。

- 標準のスクロールバー
- ミニグラフ - メイングラフの縮小版が背景に表示されるスクロールバーです。スクロールバーを使用してグラフの一部にフォーカスを移動すると、その部分のメイングラフがグラフフレーム内で移動します。このタイプのグラフは、「フォーカス依存型グラフ」と呼ばれます。

構文　X 軸スクロールの生成

```
"axisname": {
        "groupFit": {"rule": "maxCount", "value": number},
   "scroll": {
     "style": "string",
     "miniChartProperties": {
          "height": number,
          "dataSelection": {
            "selectionRect": {"fill": "rgba(r, g, b, a)" }
                           }
                         }
        }
     }
```

説明

`"axisname"`
axisList の X 軸の名前または xaxis です。

`"groupFit": {"rule": "maxCount", "value": number}`
一度に表示する最大グループ数を指定します。デフォルト値は undefined です。

`"scroll":`
X 軸スクロールのプロパティを定義します。

`"style": "string"`
スクロールバーのタイプを指定します。有効な値には、次のものがあります。

- "simple" - 標準のスクロールバーを生成します。これがデフォルト値です。
- "miniChart" - メイングラフの縮小版が背景に表示されるスクロールバーを生成します。

`"miniChartProperties":`
ミニグラフが表示されるスクロールバーのプロパティを定義します。

`"height": number`
ミニグラフの高さをピクセル単位で指定するか、メイングラフの高さに対する割合をパーセント文字列で指定します。スクロール可能なミニグラフ上の選択ボックスは、常にミニグラフ自体の高さと同一になります。デフォルト設定では、ミニグラフの高さは、メイングラフの高さの 20% です。

```
"dataSelection": "selectionRect": "fill": "rgba(r, g, b, a)"
```

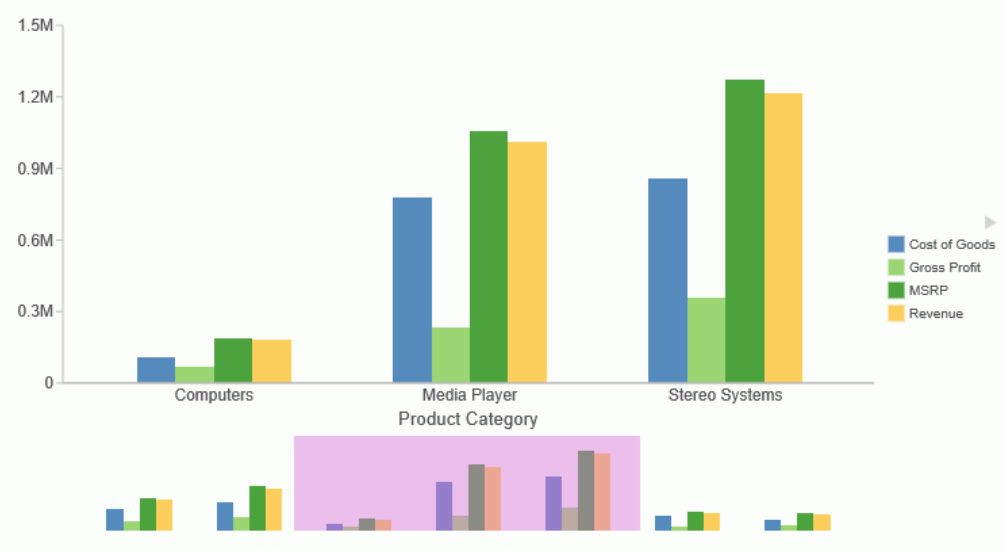
選択ボックスの色定義および透明度です。透明色を指定せずに色定義を使用すると、選択ボックスが完全な不透明になり、ミニグラフのグループ上に選択ボックスを移動した際にグループが見えなくなります。

例　X 軸でのミニグラフスクロールバーの生成

次のリクエストは、X 軸にミニグラフスクロールバーを生成します。選択ボックスは赤紫色で、不透明度は 45% です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US MSRP_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS
"axisList": {
"x1": {
    "groupFit": {"rule": "maxCount", "value": 3},
    "scroll": {
        "style": "miniChart",
        "miniChartProperties": {
            "height": 70,
            "dataSelection": {"selectionRect": {"fill": "rgba(218, 112, 214, 0.45)"}}
        }}}
}
*END
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=MSRP_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

下図は、出力結果を示しています。選択ボックスでフォーカスする範囲は最大で 3 つのグループで、下図では 2 つ目のグループブロックがフォーカスされています。



横方向スクロールバーの初期位置の設定

横軸スクロールバーの初期位置を、次のいずれかに設定することができます。

- **単一整数値**　スクロールバーの位置を指定するグループ番号に設定します。
- **単一パーセント値**　スクロールバーの位置を軸に沿ったパーセント値に設定します。
- **2 つのメンバーの配列**　スクロールバーの開始位置および終了位置を設定します。複数のグループメンバーまたはパーセント値を指定することができます。

initialPosition プロパティは、ベーシックスクロールバーおよび miniChart スクロールバーで動作します。

構文　横方向スクロールバーの初期位置の設定

```
"axisList": {
    "x1": {
        "scroll": {
            "initialPosition": value
        }
    }
}
```

説明

`"initialPosition": value`

次のいずれかの値です。

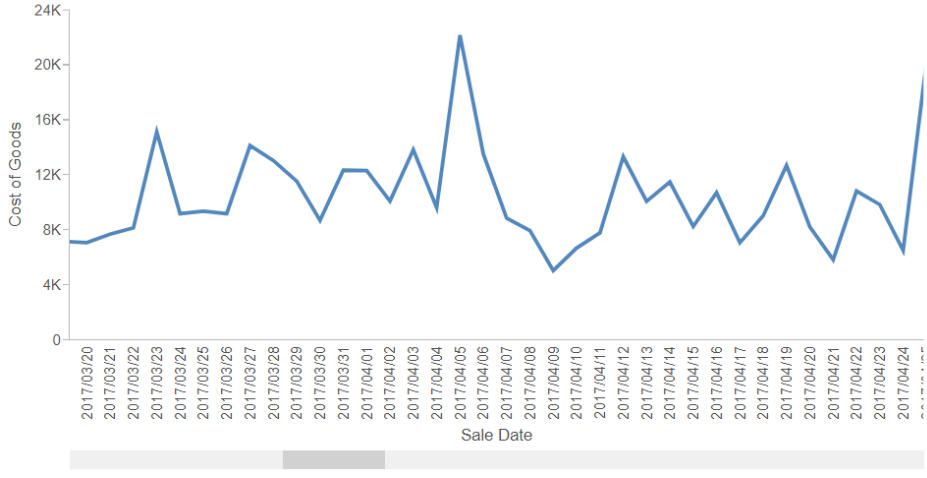
- **単一整数値** スクロールバーの位置を指定するグループ番号に設定します (例、"initialPosition": 3)。
- **単一パーセント値** スクロールバーの位置を軸に沿ったパーセント値に設定します (例、"initialPosition": "25%")。
- **2 つのメンバーの配列** スクロールバーの開始位置および終了位置を設定します (例、"initialPosition": ["25%", "75%"])。複数のグループメンバーまたはパーセント値を指定することができます。開始値および終了値のタイプを同一にする必要はありません。たとえば、"initialPosition": [3, "75%"] と指定することができます。スクロールバーは、スクロール時に、設定されたこれらの長さを保持します。

例 横方向スクロールバーの初期位置の設定

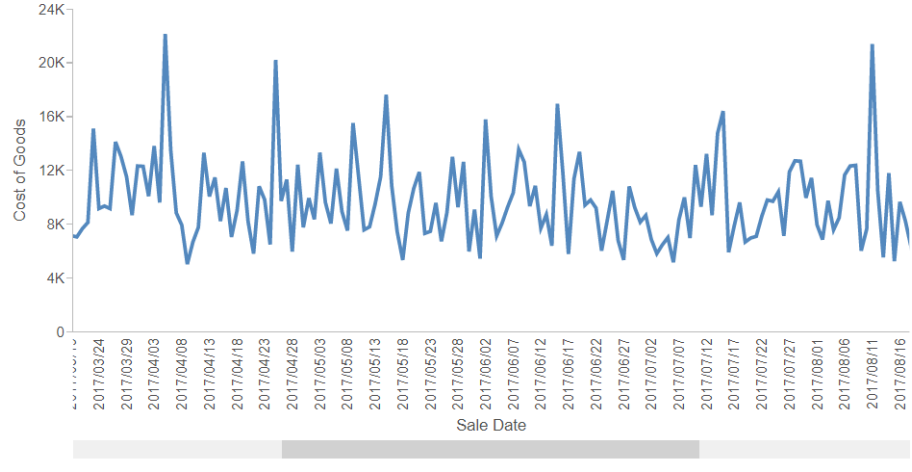
次のリクエストは、スクロールバーの初期位置を X 軸に沿った 25 パーセントに設定します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY TIME_DATE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS
"axisList": {
    "x1": {
        "scroll": {
            "initialPosition": "25%"
        }
    }
}
*END
ENDSTYLE
END
```

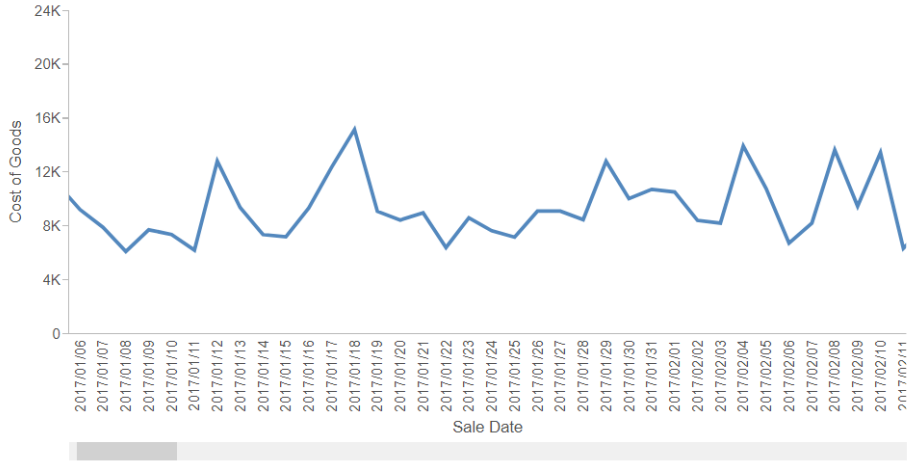
出力結果は次のとおりです。



initialPosition プロパティを "initialPosition": ["25%", "75%"] に変更すると、下図のように、スクロールバーの開始位置が X 軸に沿った 25 パーセントの位置に、終了位置が 75% パーセントの位置に、それぞれ変更されます。



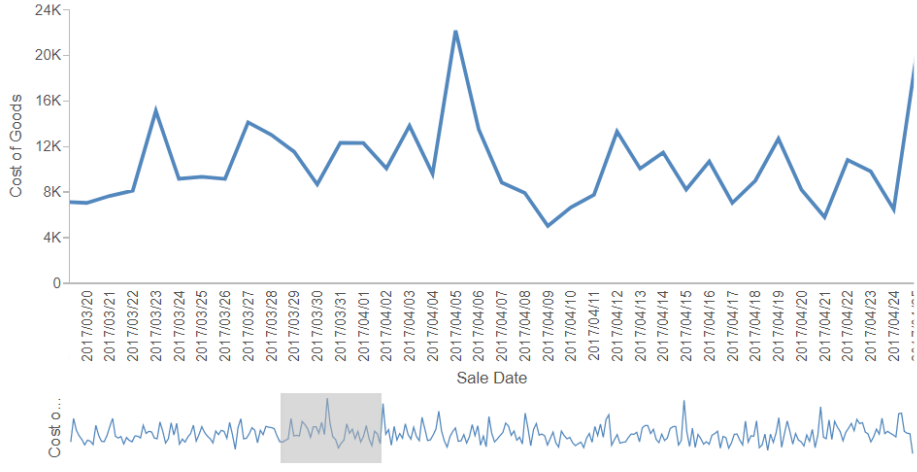
"initialPosition": 3 に変更すると、下図のように、スクロールバーの初期位置はグループ 3 から開始されます。グループ 0 (ゼロ) および グループ 1 は、スクロールされ、表示から除外されています。



次のリクエストでは、miniChart スクロールバーの初期位置を X 軸沿いの 25 パーセントの位置に設定します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY TIME_DATE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS
"axisList": {
    "x1": {
        "scroll": {"style": "minChart",
            "initialPosition": "25%"
        }
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



軸モードの設定

mode プロパティは、軸モードを設定します。通常、軸モードは undefined に設定されています。この場合、グラフエンジンがデータに基づいてモードを自動的に設定します。

構文　軸モードの設定

```
"axisname": {
  "mode": "string"
  }
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、zaxis のいずれかにすることもできます。

`"mode": "string"`

軸モードを指定する文字列です。この値が undefined の場合、データに基づいてモードが自動的に選択されます。それ以外の場合、サポートされる値は次のとおりです。

- "ordinal"
- "numeric"
- "count"

このモードは数値軸でサポートされ、不連続の値を含むメジャーに割り当てられた軸に整数目盛りを表示します。このモードは、不連続のメジャーを使用するヒストグラムおよびその他のグラフを表示する際に役立ちます。

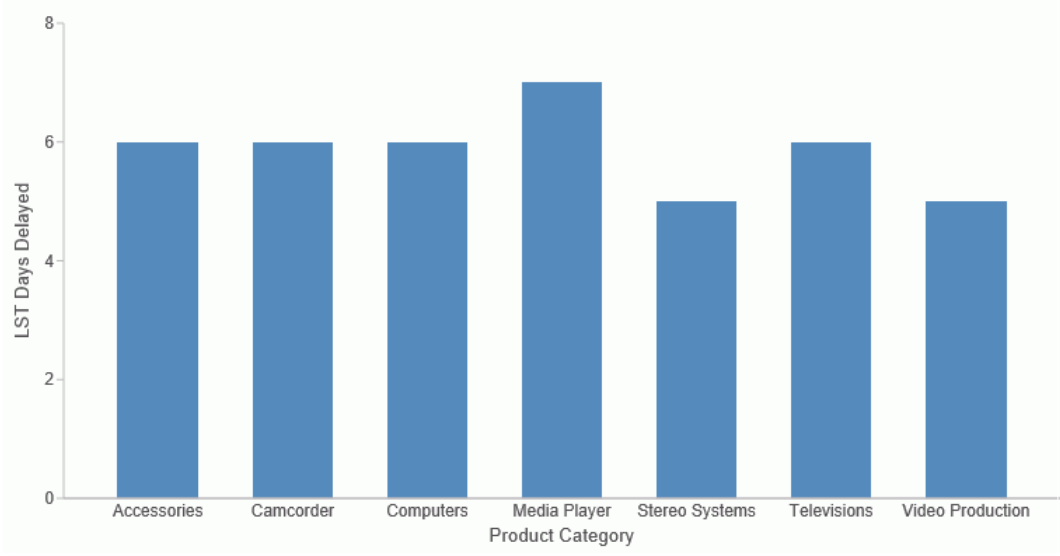
- "time"

例　数値軸での Count モードの使用

次のリクエストは、製品区分別の遅延日数を示すグラフを生成します。Y 軸のモードが "count" に設定されているため、Y 軸には整数のみが表示されます。

```
GRAPH FILE wf_retail_lite
SUM LST.DAYSDELAYED
BY PRODUCT_CATEGORY
WHERE DAYSDELAYED GT 0
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET AUTOFIT ON
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS
 "axisList:"{
     "y1": {"mode": "count"}
           },
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



数値軸ラベルのフォーマットの制御

数値軸では、numberFormat プロパティを使用して、数値軸ラベルのフォーマットを制御することができます。数値フォーマットは、JSON オブジェクト、フォーマット文字列、ユーザ定義関数のいずれかで指定することができます。

数値フォーマット定義の詳細および例は、86 ページの 「 数値のフォーマット設定 」 を参照してください。

軸のデフォルト表示位置の入れ替え

swapChartSide プロパティは、軸線および軸ラベルの位置を制御します。デフォルト構成では、Y 軸の軸線および軸ラベルは、横方向グラフでは下側に、縦方向グラフでは左側に表示されます。Y2 軸の軸線および軸ラベルは、横方向グラフでは上側に、縦方向グラフでは右側に表示されます。バブルグラフおよび散布図では、一般に X 軸はグラフの下側に、Y 軸はグラフの左側に表示されます。このプロパティを使用して、これらのデフォルト表示位置を入れ替えることができます。

構文　　軸のデフォルト表示位置の入れ替え

```
"axisname": {
  "swapChartSide": boolean  }
```

説明

"axisname"
axisList の軸にすることも、xaxis、yaxis、y2axis のいずれかにすることもできます。

"swapChartSide": boolean
次のいずれかの値を指定します。

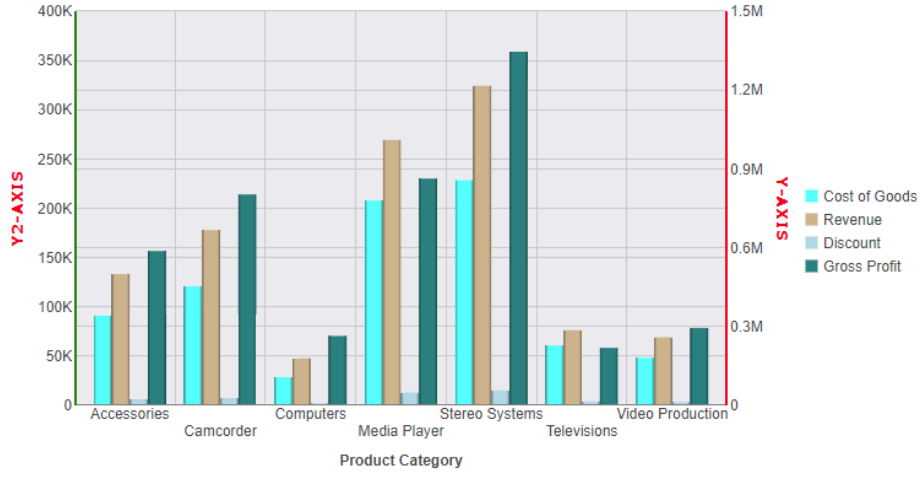
- true - 軸ラベルと軸線の表示位置を入れ替えます。
- false - 軸のデフォルト表示位置を使用します。これがデフォルト値です。

例　軸のデフォルト表示位置の入れ替え

次のリクエストは、シリーズ 3 を Y2 軸に割り当て、Y2 軸と Y 軸を入れ替えます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
REVENUE_US
DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"yaxis": {
   "title": {
      "visible": true,"color": "red", "font": "bold 10pt Verdana",
      "text": "Y-AXIS"
      },
   "bodyLineStyle": {"width": 2, "color": "red"},
   "swapChartSide": true},
"y2axis": {
   "title": {
      "visible": true, "color": "red", "font": "bold 10pt Verdana",
      "text": "Y2-AXIS"
      },
   "bodyLineStyle": {"width": 2, "color": "green"}},
"series": [
   {"series": 0, "color": "cyan"},
   {"series": 1, "color": "tan"},
   {"series": 2, "color": "lightblue"},
   {"series": 3, "color": "teal","yAxisAssignment": 2}]
*END
ENDSTYLE
END
```

出力結果では、Y 軸がグラフの右側に移動し、Y2 軸が左側に移動しています。



次のリクエストは、バブルグラフを生成し、Y 軸を右側に、X 軸を上側に表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"axisList": {
"y1": {"swapChartSide": true},
"x1": {"swapChartSide": true}
  },
"series": [
    {"series": "all", "marker": {"shape": "circle"}},
    {"series": 0, "color": "red"},
    {"series": 1, "color": "green"},
    {"series": 2, "color": "blue"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

軸タイトルのフォーマット設定

title プロパティは、軸タイトルのテキスト、表示、フォーマットを制御します。

構文　　軸タイトルのフォーマット設定

```
"axisname": {
  "title": {
    "text": "string",
    "visible": boolean,
    "font": "string",
    "color": "string",
    "position": "string"
}
}
```

説明

`"axisname"`

axisList の軸にすることも、xaxis、yaxis、y2axis、zaxis のいずれかにすることもできます。

`"text": "string"`

軸タイトルのテキストを定義する文字列です。X 軸のデフォルト値は、ソートフィールドの名前またはタイトル、あるいは AS 句で割り当てられた名前です。

"visible": boolean

軸タイトルの表示を制御します。有効な値には、次のものがあります。

- true - 軸タイトルを表示します。
- false - 軸タイトルを表示しません。これがデフォルト値です。

"font": "string"

軸タイトルのサイズ、スタイル、タイプフェースを定義する文字列です。X 軸のデフォルト値は、"bold 9pt Sans-Serif" です。

"color": "string"

色名または数値指定の文字列で軸タイトルの色を定義します。デフォルト値は "black" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"position": "string"

次のいずれかの値です。

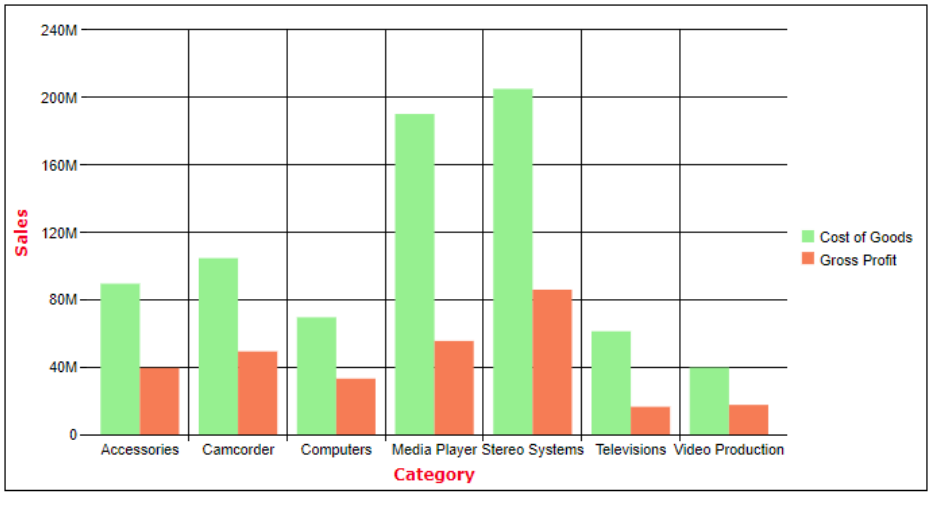
- **"orthogonal"** Y 軸と Y2 軸に適用されます。タイトルを軸線の上側の中央に配置します。
- **"flip"** Y2 軸に適用されます。Y 軸タイトルを反転 (180 度回転) させます。

例　軸タイトルのフォーマット設定

次のリクエストは、軸タイトルを表示し、赤色に設定します。また、軸タイトルのフォントを太字、10pt、Verdana に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS_FINAL
"xaxis": {"title": {
    "visible": true, "color": "red", "font": "bold 10pt Verdana", "text":
"Category"}},
"yaxis": {"title": {
    "visible": true, "color": "red", "font": "bold 10pt Verdana", "text":
"Sales"}},
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"}]
*END
ENDSTYLE
END
```

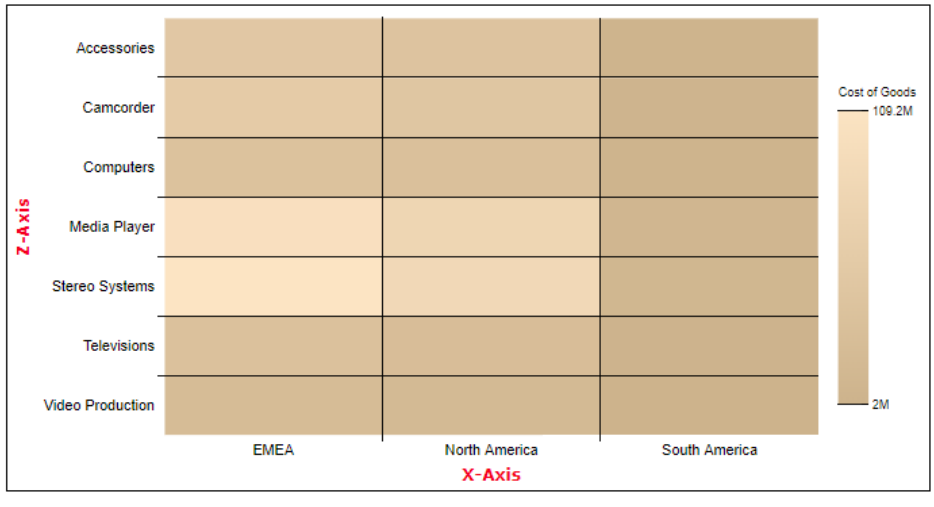
出力結果は次のとおりです。



次のリクエストは、ヒートマップを生成し、X 軸および Z 軸のタイトルを赤色、太字、10pt、Verdana フォントに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH SET LOOKGRAPH HEATMAP
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=y-axis, $
type=data, column=business_region, bucket=x-axis, $
type=data, column=cogs_us, bucket=color, $
*GRAPH_JS_FINAL
"colorScale": {"colors": ["tan", "bisque"]},
"xaxis": {"title": {
    "visible": true, "color": "red", "font": "bold 10pt Verdana", "text": "X-Axis"}},
"zaxis": {"title": {
    "visible": true, "color": "red", "font": "bold 10pt Verdana", "text": "Z-Axis"}}
*END
ENDSTYLE
END
```

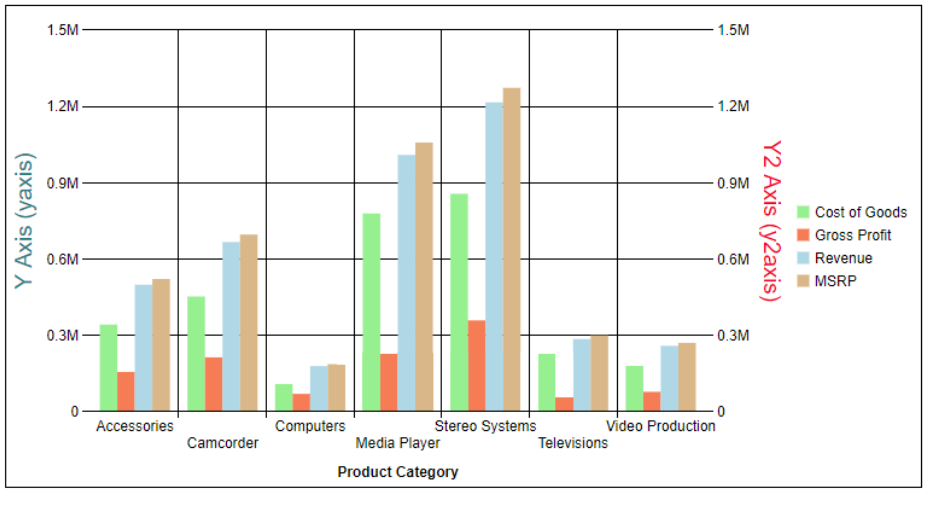
出力結果は次のとおりです。



次のリクエストは、Y 軸に 2 つのシリーズ、Y2 軸に 2 つのシリーズを割り当て、これらの軸タイトルのフォーマットを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis(2),$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis(2),$
*GRAPH_JS_FINAL
"axisList": {
"y1": {"title": {
   "visible": true, "text": "Y Axis (yaxis)", "font": "14pt Sans-Serif", "color":
"teal"}},
"y2": {"title": {
   "visible": true, "text": "Y2 Axis (y2axis)", "font": "14pt Sans-Serif", "color":
"red"}}
},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "coral"},
   {"series": 2, "color": "lightblue"},
   {"series": 3, "color": "burlywood"}]
*END
ENDSTYLE
END
```

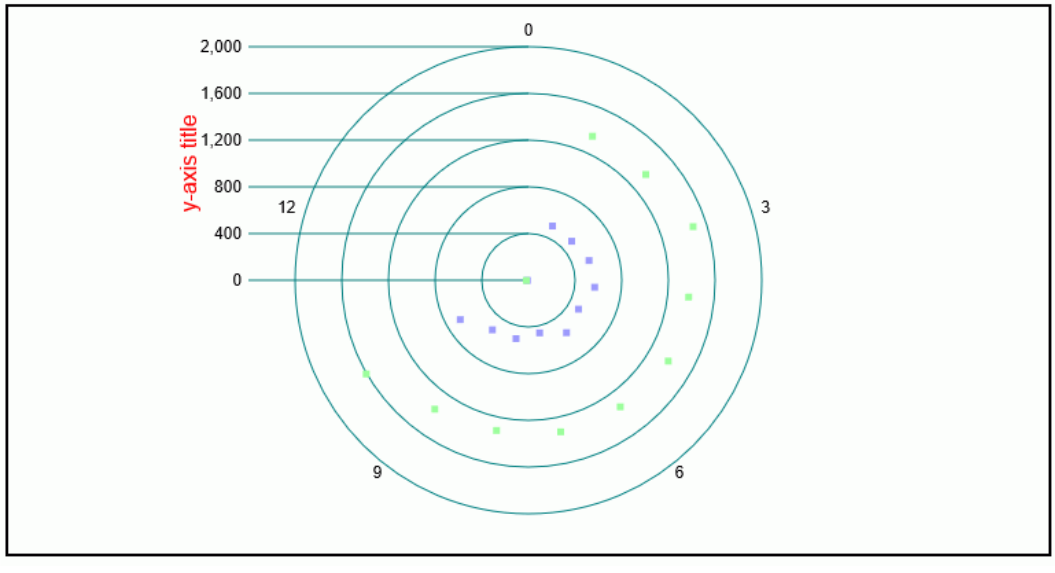
出力結果は次のとおりです。



次のリクエストは、極グラフを生成し、Y 軸タイトルのフォーマットを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
ACROSS TIME_MTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH POLAR
ON GRAPH SET STYLE *
*GRAPH_JS
"legend": {"visible": false},
"polarProperties": {"straightGridLines": false, "extrudeAxisLabels": true},
"yaxis": {
    "title": {"visible": true,
        "text": "y-axis title", "font": "12pt Sans-Serif", "color": "red"},
    "majorGrid": {"visible": true, "lineStyle": {"width": 1, "color":
"teal"}
        }}
*END
ENDSTYLE
END
```

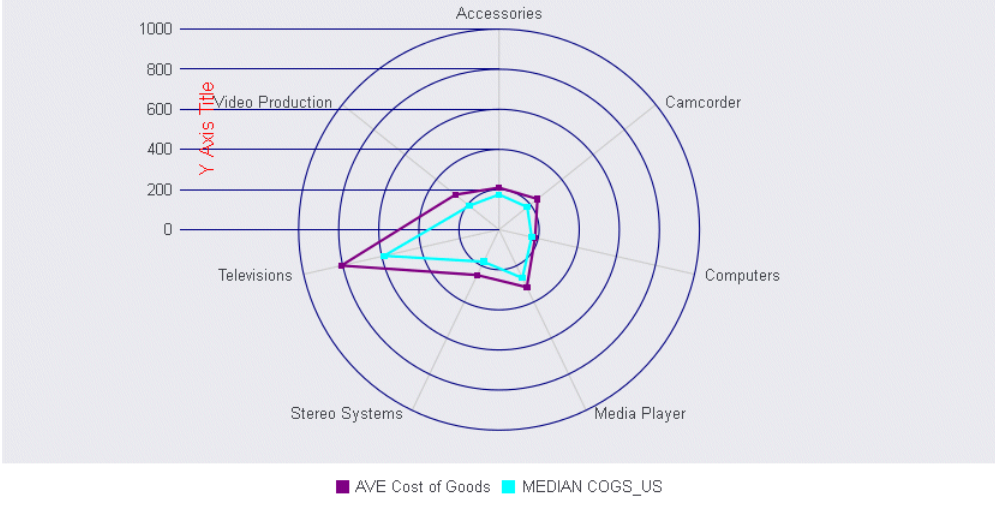
出力結果は次のとおりです。



次のリクエストは、レーダーグラフを生成し、Y 軸タイトルを表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH RADARL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": true},
"polarProperties": {"extrudeAxisLabels": true},
"yaxis": {
    "title": {"visible": true, "text": "Y Axis Title",
        "font": "12pt Sans-Serif", "color": "red"},
    "majorGrid": {"lineStyle": {"width": 1, "color": "navy"}}},
"series": [
    {"series": 0, "color": "purple", "border": {"width": 2}},
    {"series": 1, "color": "cyan", "border": {"width": 2}}]
*END
END
```

出力結果は次のとおりです。



Y 軸または Y2 軸のタイトル位置の定義

"position": "orthogonal" プロパティを使用すると、Y 軸または Y2 軸のタイトルが、Y 軸または Y2 軸の上側の中央に配置されます。

構文　Y 軸線の上側への Y 軸タイトルの配置

```
"axisname": {
   "title":{
     "visible": true,
     "position": "orthogonal"
}
}
```

説明

`"axisname"`

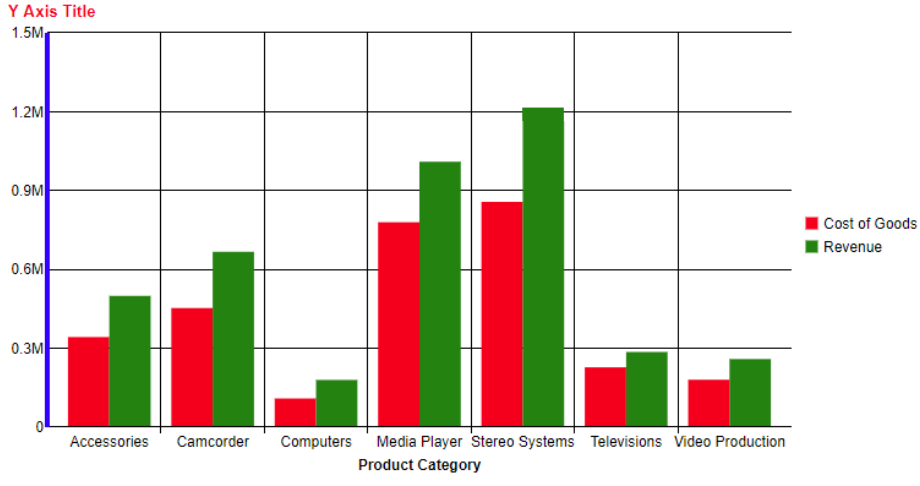
axisList の Y 軸にすることも、yaxis、y2axis のいずれかにすることもできます。

例　Y 軸線の上側への Y 軸タイトルの配置

次のリクエストは、Y 軸タイトルを赤色、太字、10pt Arial に設定し、タイトルを Y 軸線の上側に配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 0},
"axisList": {
 "y1": {
     "title": {"visible": true, "position": "orthogonal",
        "text": "Y Axis Title", "font": "bold 10pt Arial", "color": "red"},
     "bodyLineStyle": {"width": 4, "color": "blue"}}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Y2 軸タイトル表示の反転

"title": {"position": "flip"} プロパティを使用して、Y2 軸タイトルを反転 (180 度回転) させることができます。

構文　　Y2 軸タイトル表示の反転

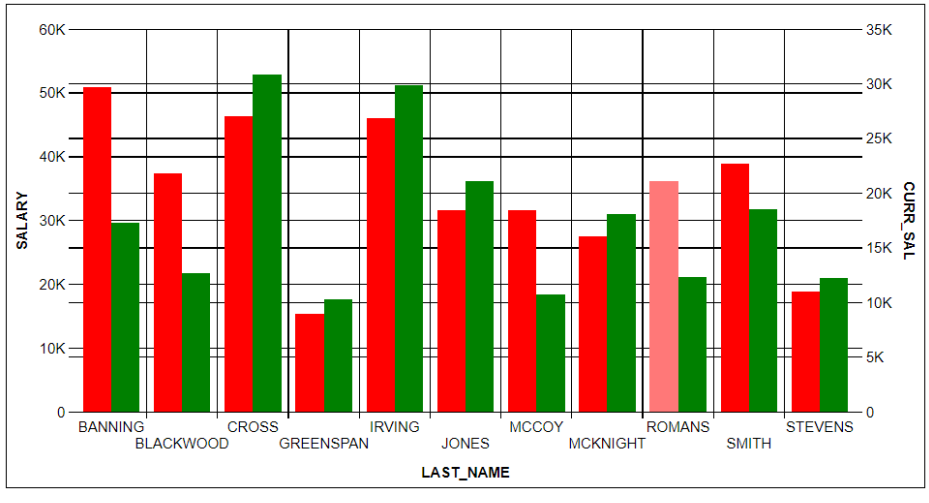
```
"axisList": {
  "y2": {
    "title": {"position": "flip"}
  }
}
```

例　Y2 軸タイトル表示の反転

次のリクエストは、Y2 軸タイトルをデフォルト設定の方向で表示します。

```
GRAPH FILE EMPLOYEE
SUM CURR_SAL
SALARY
BY LAST_NAME
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis(2), $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis(1), $
*GRAPH_JS
"legend": {"visible": false}
*END
ENDSTYLE
END
```

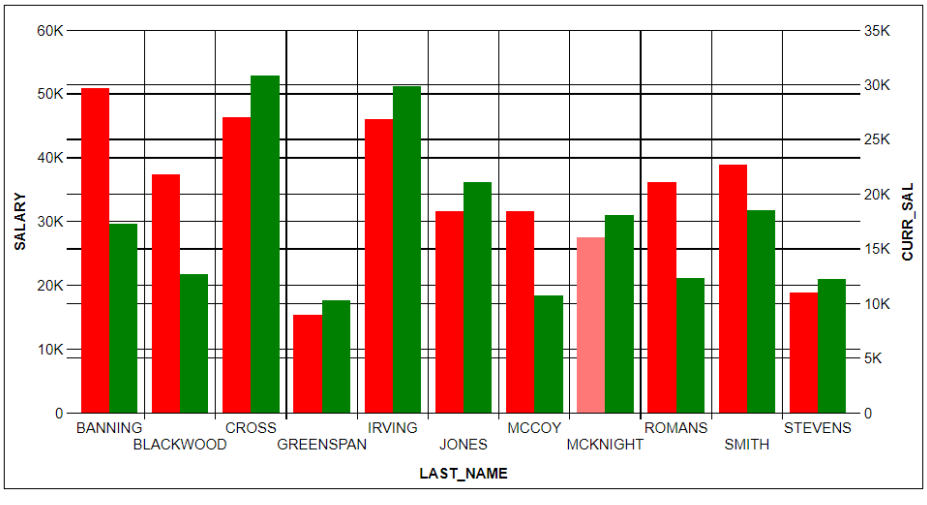
出力結果は次のとおりです。



次のリクエストは、Y2 軸タイトルを反転させます。

```
GRAPH FILE EMPLOYEE
SUM CURR_SAL
SALARY
BY LAST_NAME
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH PCHOLD FORMAT JSCHART

ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis(2), $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis(1), $
*GRAPH_JS
"legend": {"visible": false},
"axisList": {
  "y2": {
    "title": {"position": "flip" }
  }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。タイトルの「CURR_SAL」が反転しています。



時間軸のフォーマット設定

時間軸は、軸上に日付、時間、または日付時間値を表示します。

時間軸は、X 軸スクロールをサポートします。単純スタイルおよびフォーカスコンテキスト (ミニグラフ) スタイルの両方がサポートされます。

構文　時間軸のフォーマット設定

軸に日付と時間の値が含まれている場合、軸に配置する日付時間の値およびプロットするデータポイントが多数存在すると、グラフが判読困難になります。

注意：現在、時間軸は、マトリックスグラフではサポートされていません。

時間軸のプロパティを有効にするには、次のスタイルシート属性を、日付時間値を表示する軸の宣言に追加する必要があります。軸ラベルの時間間隔が自動的に選択されるため、より使いやすくなっています。

```
axis-type = time_auto
```

次の軸プロパティは、グラフエンジンが日付時間ローデータを集計し、より少数のポイント (可能な場合は集計済みポイント) を提供することを可能にします。これらのプロパティは、日付時間値を表示する軸のプロパティに追加する必要があります。

```
"timeAxis":
{
      "dataAggregation":
     {
         "enabled": boolean,
         "interval": "string",
         "round": boolean,
         "method": "string"
         "runningRange": number
     }
}
```

説明

`"timeAxis"`
　時間軸データの集計プロパティを定義します。

　時間軸は、日付時間値を表示する軸のスタイルシート宣言に、axis-type=time_auto を追加することで有効になります。

　`"dataAggregation"`
　　データセットを簡素化するローリング集計を制御する、データ集計プロパティを定義します。

　　`"enabled": boolean`
　　　データ集計を有効にするかどうかを定義します。有効な値は、true または false です。デフォルト値は false です。

`"interval": "string"`
次のいずれかの値です。

- "auto" - グラフ上のデータ値の間隔を自動的に設定します。これがデフォルト値です。下表は、"auto" の規則を示しています。

下限 (この値を超える)	上限 (この値を含む)	集計精度
0	1 時間	秒
1 時間	12 時間	分
12 時間	3 日	3 分
3 日	2 週間	15 分
2 週間	1 か月	時間
1 か月	6 か月	3 時間
6 か月	3 年	日
3 年	10 年	週
10 年	20 年	2 週間
20 年		月

- "autods" - データサイエンスに特化した規則に基づいて、グラフ上のデータ値の間隔を自動的に設定します。下表は、これらを示しています。

下限 (この値を超える)	上限 (この値を含む)	集計精度
0	5 日	30 分
5 日	1 年	日
1 年	5 年	週
5 年		4 週間

- 秒数

"round": boolean
: true の場合、間隔に応じて開始点の端数処理を行います。たとえば、間隔が "hour" の場合、開始点が切り上げられて、1 時間になります。false の場合、端数処理は行われません。デフォルト値は undefined です。

"method": "string
: 使用する集計タイプを定義します。有効な値は、"average"、"min"、"max"、"first"、"last" です。デフォルト値は "average" です。

"runningRange": number
: 現在の点から前後に移動して、ランニング集計に使用する間隔の数です。デフォルト値は undefined です。この場合、現在の点から開始して、1 間隔分前方に移動します (ランニング平均の計算に使用)。

timeAxis プロパティは、axis-type=time_auto スタイルシート属性と連動し、より高パフォーマンスなグラフを表示します。たとえば、3 か月間の株価を観察する場合、入力ローデータが 5 分間隔で 90 日分の株価を含んでいても、線上のデータポイントは日単位で表示されている方が望ましい場合があります。

デフォルトのプロパティは、これらの利点を提供します。必要に応じて、データ集計プロパティをカスタマイズすることもできます。

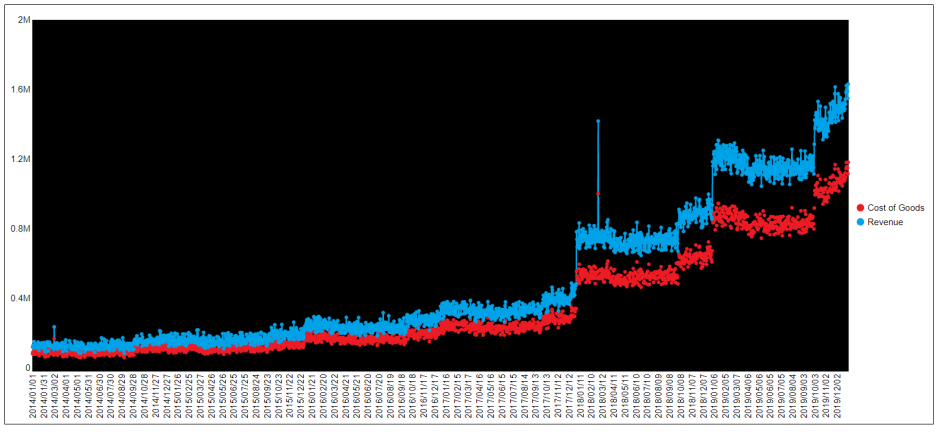
データにギャップが存在する場合は、"showNullGroups": false を使用して、空の期間をスキップします。

例　時間軸の表示

次のリクエストは、X 軸に日付値を表示するグラフを生成します。時間軸プロパティは設定されていません。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
REVENUE_US
BY TIME_DATE
WHERE TIME_DATE GT '1900/01/01'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis,$
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS
"series":
     [
        {
            "series": 0,
            "color": "rgba(237, 28, 36, 1)"
        },
        {
          "series": 1,
            "color": "rgba(0, 162, 232, 1)"

        }
    ],
*END
ENDSTYLE
END
```

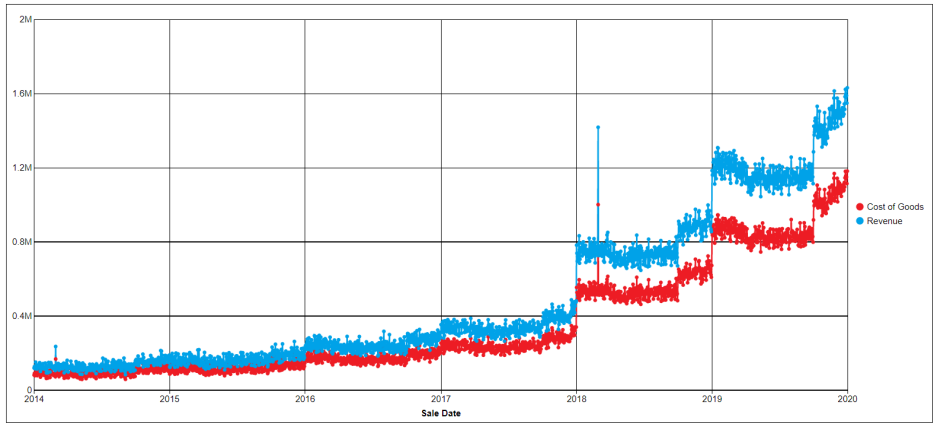
出力結果は次のとおりです。目盛りが互いに近接しすぎるため、背景は黒に見えます。



次のリクエストには、x-axis 属性カテゴリに axis-type=time_auto 属性が追加されています。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
REVENUE_US
BY TIME_DATE
WHERE TIME_DATE GT '1900/01/01'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, axis-type=time_auto, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS
"series":
     [
        {
            "series": 0,
            "color": "rgba(237, 28, 36, 1)"
        },
        {
          "series": 1,
            "color": "rgba(0, 162, 232, 1)"

        }
    ],
*END
ENDSTYLE
END
```

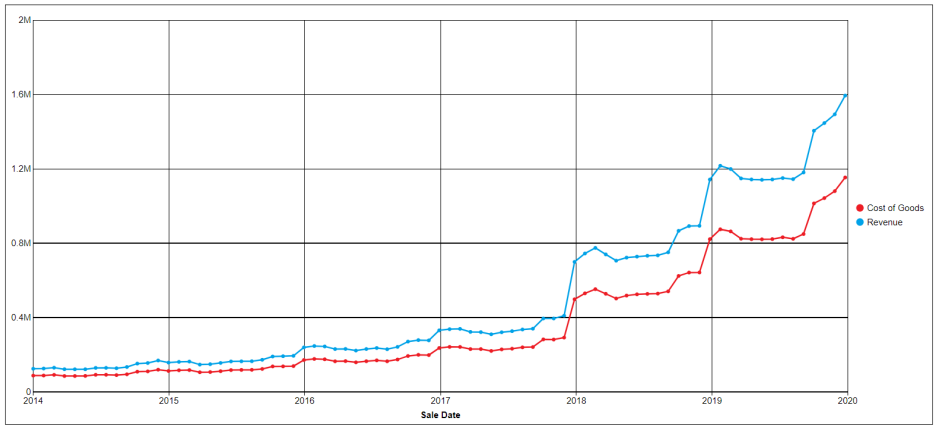
出力結果は次のとおりです。X 軸ラベルは読みやすくなっていますが、データポイントの判別は困難です。



次のリクエストには、x-axis 属性カテゴリに axis-type=time_auto スタイルシート属性が追加され、GRAPH_JS ブロックの xaxis プロパティに時間軸データ集計プロパティが追加されています。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
REVENUE_US
BY TIME_DATE
WHERE TIME_DATE GT '1900/01/01'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, axis-type=time_auto, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS
"xaxis": {
 "timeAxis": {
     "dataAggregation": {"enabled": true, "interval": "autods"}
           }
},
"series":
     [
        {
            "series": 0,
            "color": "rgba(237, 28, 36, 1)"
        },
        {
          "series": 1,
            "color": "rgba(0, 162, 232, 1)"

        }
    ],
*END
ENDSTYLE
END
```

下図のように、出力結果は分析が容易になります。

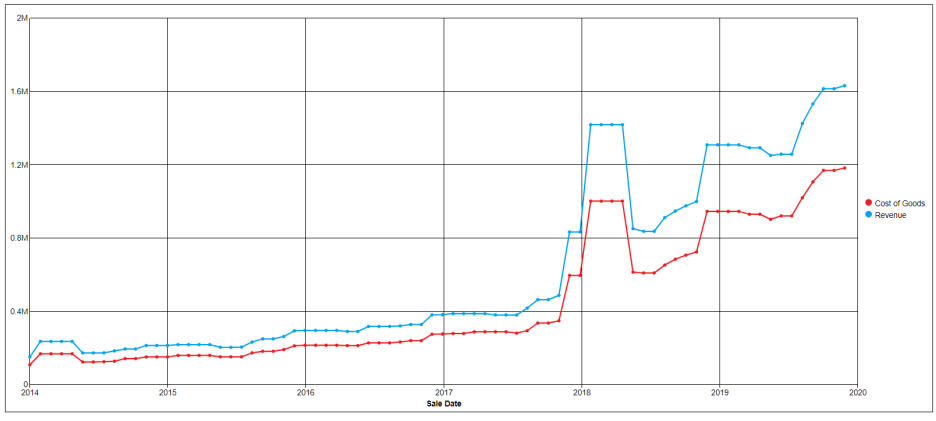


例　データ集計プロパティのカスタマイズ

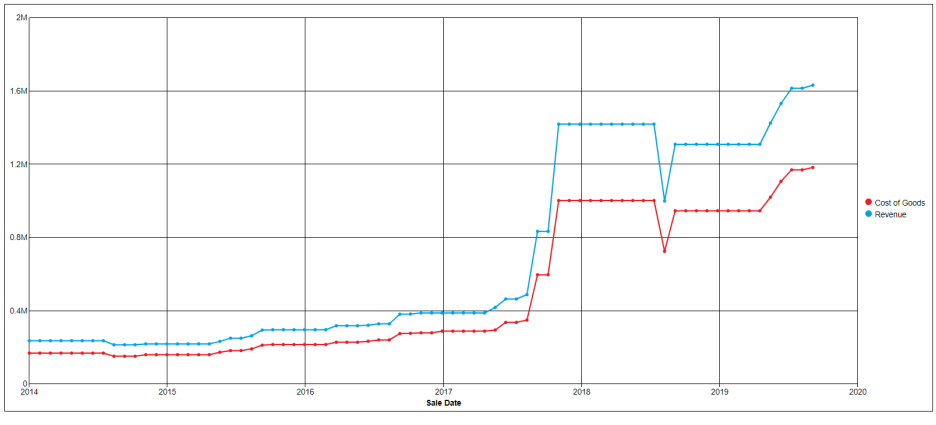
次のリクエストは時間軸を有効にし、データ集計方法を "max"、実行範囲を 2、間隔を "autods" に設定しています。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
REVENUE_US
BY TIME_DATE
WHERE TIME_DATE GT '1900/01/01'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, axis-type=time_auto,$
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS
"xaxis":
{
 "timeAxis":
  {
     "dataAggregation":
     {
   "enabled": true,
       "interval": "autods",
       "method": "max",
       "runningRange": 2
     }
  }
},
"series":     [
        {
            "series": 0,
            "color": "rgba(237, 28, 36, 1)"
        },
        {
          "series": 1,
            "color": "rgba(0, 162, 232, 1)"
        }
    ],
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



実行範囲を 5 に変更すると、下図の出力結果が生成されます。

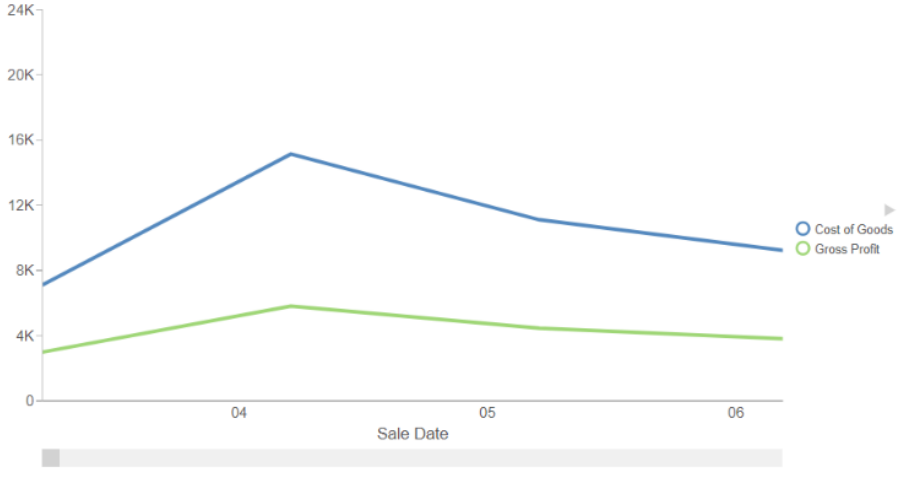


例　時間軸での X 軸スクロールの使用

次のリクエストは、時間軸と単純スクロールバーを表示するグラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DATE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS_FINAL
 "xaxis":
      {
        "groupFit": {"rule": "maxCount", "value": 3},
        "scroll":
                {
                 "style": "simple"
                },
        "timeAxis": { "dataAggregation": {"enabled": true,
                                          "interval":"auto"} },
       }
*END
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=TIME_DATE, BUCKET=x-axis,
axis-type=time_auto,
$
ENDSTYLE
END
```

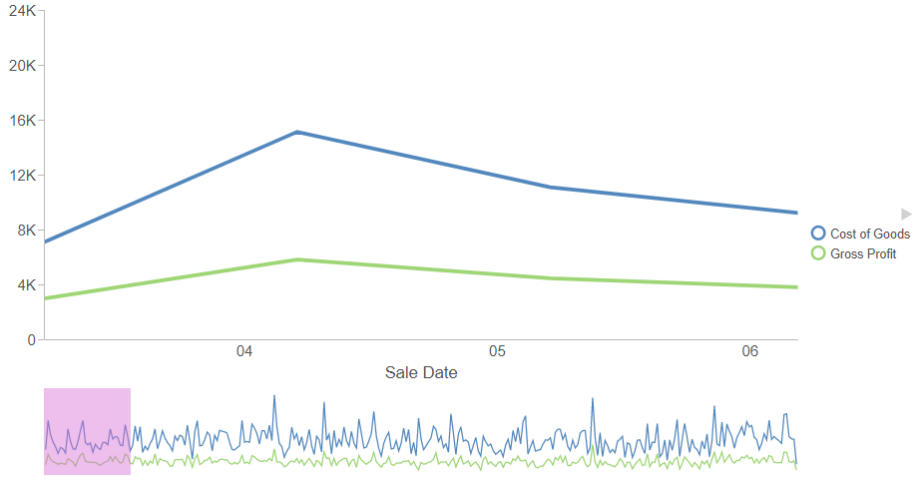
出力結果は次のとおりです。"interval": "auto" プロパティによって、X 軸ラベルが集約され、月単位で表示されています。



次のリクエストでは、スクロールバーにミニグラフフォーマットが設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DATE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS_FINAL
 "xaxis":
      {
        "groupFit": {"rule": "maxCount", "value": 3},
        "scroll":
                {
                 "style": "minichart",
                 "groups": {"maxCount": 3},
                 "miniChartProperties":
                     {
                         "height": 70,
                         "dataSelection": {
                            "selectionRect":
                            {
                              "fill": "rgba(218, 112, 214, 0.45)"}
                            }
                      }
                },
        "timeAxis": { "dataAggregation": {"enabled": true,
                                          "interval":"autods"} },
       }
*END
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=TIME_DATE, BUCKET=x-axis,
axis-type=time_auto,
$
ENDSTYLE
END
```

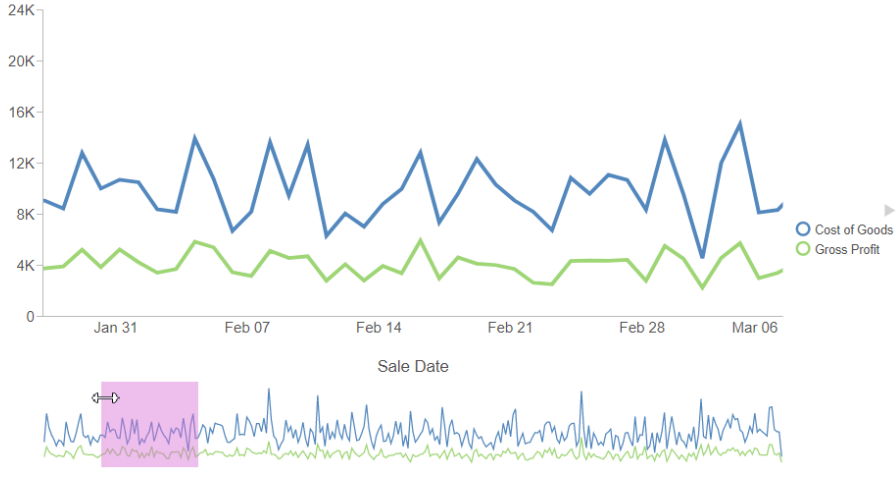
出力結果は次のとおりです。"interval": "autods" プロパティによって、X 軸ラベルが集約され、月単位で表示されています。



注意

- ミニグラフ上のピンク色の四角形は、グラフ出力上で現在表示されている部分を示しています。時間データのその他の部分を表示するには、単純スクロールバーと同様に、このピンク色の四角形を選択し、左右に移動します。

- ピンク色の四角形は、左右どちら側からでもサイズを変更することができ、実際のグラフ上で、一度に表示する点の数を増減することができます。下図では、ピンク色の四角形の位置が移動され、サイズが変更されています。



フォーカスコンテキストスクロールバーについての詳細は、390 ページの 「X 軸スクロールのフォーマット設定」 を参照してください。

9 シリーズ固有のプロパティ

ここでは、特定のシリーズおよびグループの外観を制御する方法について説明します。

トピックス

- シリーズ固有のプロパティの概要
- 特定シリーズの選択
- 特定グループの選択
- シリーズライザの境界線の定義
- シリーズライザの色の定義
- データテキストラベルの表示およびフォーマット設定
- 円グラフからの円項目の削除
- 円グラフからの円項目の分離
- 特定シリーズへのラベルの割り当て
- シリーズマーカーのサイズ、境界線、色、形状、回転の定義
- 棒グラフ、折れ線グラフ、面グラフのシリーズライザ形状の定義
- 散布図でのマーカーの接続
- シリーズライザおよびマーカーでの塗りつぶしパターン名の使用
- 棒グラフのライザ角形状の曲線化
- ライザのツールヒントの定義または非表示
- バブルグラフおよび散布図でのシリーズ傾向線の描画
- 棒グラフ、折れ線グラフ、面グラフでのシリーズ傾向線の描画
- 特定シリーズの表示の制御
- 軸へのシリーズの割り当て

シリーズ固有のプロパティの概要

シリーズ固有のプロパティを使用して、特定のシリーズおよびグループの外観を制御することができます。

注意: データラベルのプロパティはシリーズ固有のプロパティとなったため、showDataValues プロパティは廃止され、不要になりました。

次の構文は、シリーズのプロパティを示しています。

```
series: [
{
   "series": number,
   "group": number,
   "visible": true,
   "label": "string",
   "color": "string",
   "riserShape": "string",
   "connectMarkers": false,
   "border": {
      {"cornerRadius":
           {"x": undefined,
            "y": undefined},
      }
      "width": number,
      "color": "string",
      "dash": "string"
   },
   "marker": {
      "visible": boolean,
      "color": "string",
      "size": number,
      "shape": "string",
      "rotation": number,
      "position": "string",
      "fillEffect": "string",
      "border": {
         "width": number,
         "color": "string",
         "dash": "string"
      }
   },

"dataLabels": {
   "visible": false,
   "displayMode": "x",
   "position": "top",
   "font": "7.5pt Sans-Serif",
   "color": "black",
   "useNegativeColor": false,
   "negativeColor": "red",
   "numberFormat": "auto",
   "formatCallback": undefined
   "feelerLine": {
      "visible": true,
      "width": 1,
      "color": "black",
      "dash": ""}
},
```

```
   // Per-series trendline
   "trendline": {
      "enabled": false,
      "mode": undefined,
      "order": 3,
      "lineStyle": {
         "width": 1,
         "color": undefined,
         "dash": ""
      },
      "equationLabel": {
         "visible": false,
         "font": "8pt Sans-Serif",
         "color": undefined,
         "mode": "equation"
      }
   },
   "explodeSlice": number,
   "deleteSlice": boolean,
   "yAxisAssignment": number,
   "tooltip": tool_string_or_function
}
]
```

次の構文セグメントは、デフォルト設定を示しています。

```
"series": [
    {"series": "all", "color": "blue",
        "showDataValues": false, "border": {"width": 2},
        "marker": {"size": 8, "border": {
            "width": 1, "color": "black"}}},
    {"series": 0, "color": "red"},
    {"series": 1, "color": "green"},
    {"series": 2, "color": "orange"}]
]
```

次の構文セグメントは、塗りつぶしパターン名のプロパティを示しています。

```
 "color":{
                "type": "pattern",
                 "color": undefined,
                    "backgroundColor": undefined,
                 "lineWidth": 1,
                    "antialias": true,
                 "shape": undefined,
                    "size": 20,
                 "pad": 1
                }
```

特定シリーズの選択

特定のシリーズを選択するには、0 (ゼロ) を基準とするシリーズ番号、または文字列を使用します。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

構文　特定シリーズの選択

```
"series": [
{"series": number, "property":value, ..., property:value},
]
```

説明

`"series": number`

指定したプロパティを適用するシリーズを選択します。有効な値には、次のものがあります。

- シリーズ番号 - 0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。
- "all" - 指定したプロパティをすべてのシリーズに適用します。
- "reset" - 指定したプロパティを適用する前にデフォルトプロパティを削除します。

`"property":value`

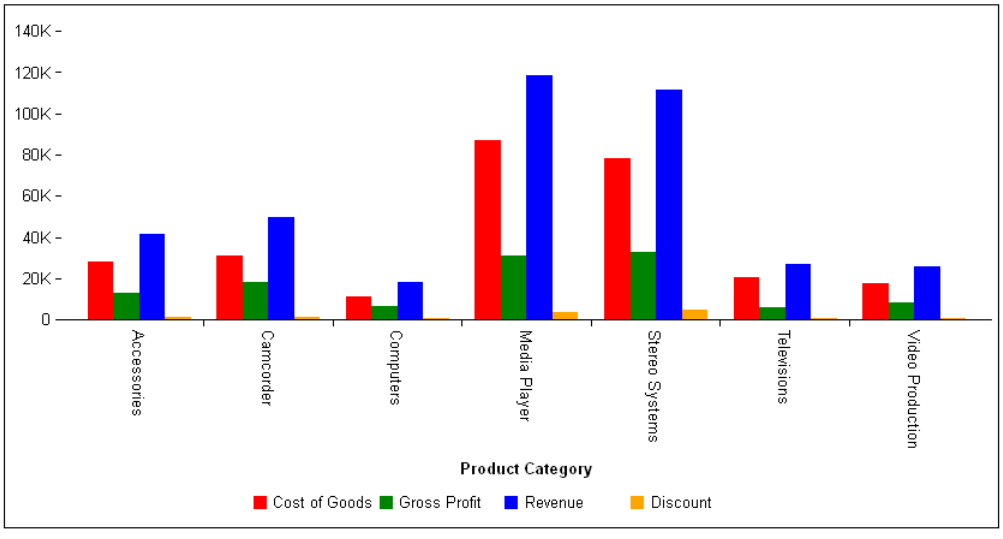
サポートされているシリーズプロパティおよび値です。

例　特定シリーズの選択

次のリクエストは、4 つのシリーズを表示する縦棒グラフを生成し、10 個のシリーズに色を指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "red"},
   {"series": 1, "color": "green"},
   {"series": 2, "color": "blue"},
   {"series": 3, "color": "orange"},
   {"series": 4, "color": "lightgreen"},
   {"series": 5, "color": "yellow"},
   {"series": 6, "color": "slateblue"},
   {"series": 7, "color": "lavender"},
   {"series": 8, "color": "limegreen"},
   {"series": 9, "color": "red"}
   ]
*END
ENDSTYLE
END
```

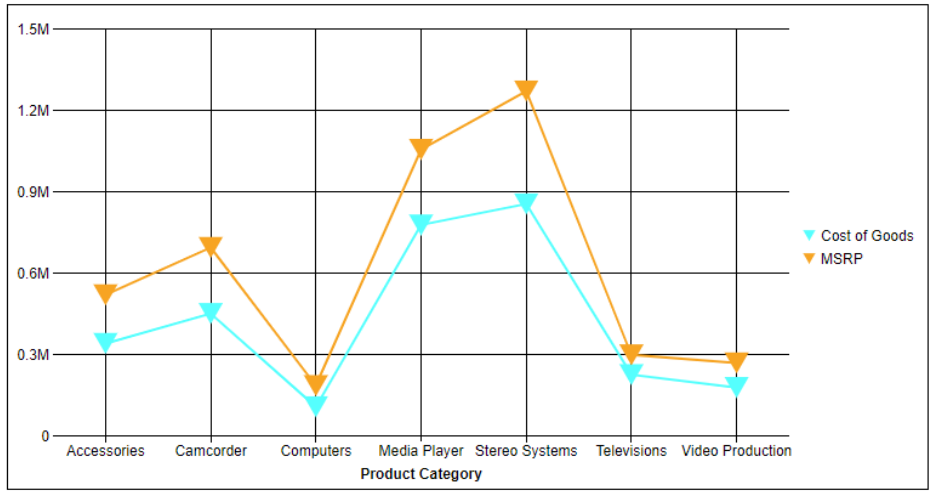
グラフに表示されないシリーズの色は無視されます。出力結果は次のとおりです。



次のリクエストは、縦折れ線グラフを生成し、すべてのシリーズのマーカー形状を三角形、サイズを 20 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"series": [
    {"series": "all", "marker": {"shape": "triangle", "size": 20}},
    {"series": 0, "color": "cyan"},
    {"series": 1, "color": "orange"}]
*END
ENDSTYLE
END
```

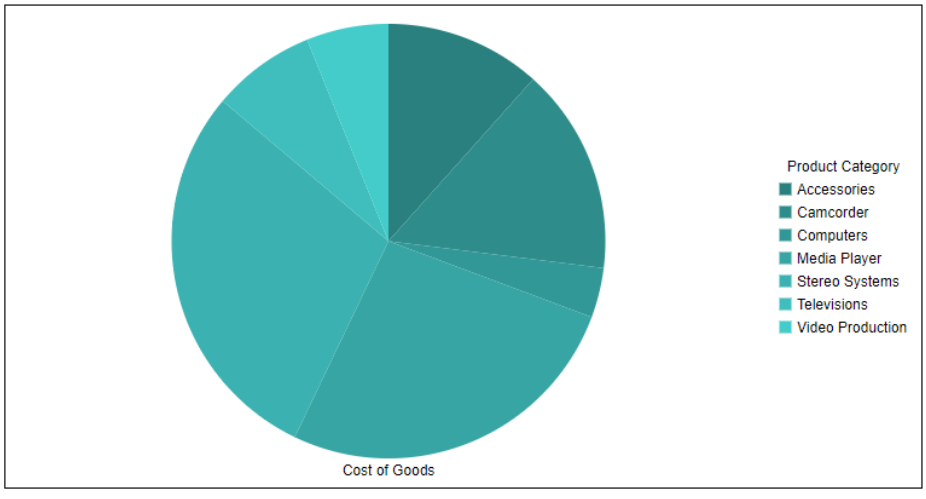
出力結果は次のとおりです。



次のリクエストでは、定義された色を周期的に繰り返す riserCycleEndLightness プロパティを設定します。このプロパティを使用すると、色が定義されたシリーズ数より多くのシリーズがグラフに存在する場合に、残りのシリーズに陰影効果が適用されます。ただし、この効果を適用するには、デフォルト色を削除する必要があります。削除しない場合、陰影効果の代わりにデフォルト色が使用されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"riserCycleEndLightness": 0.4,
"series": [
    {"series": "reset", "color": undefined},
    {"series": 0, "color": "teal"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



特定グループの選択

リクエストで特定のグループを選択するには、0 (ゼロ) を基準とするグループ番号を使用します。

構文　　特定グループの選択

```
{"series": number, "group": number,
   "property": value, ..., "property": value}
```

説明

`"series": number`

選択するグループのシリーズを定義します。

`"group": number`

0 (ゼロ) を基準とする番号です。選択したグループがグラフに存在しない場合、そのプロパティは無視されます。グループ番号を指定しない場合、そのプロパティがシリーズ内のすべてのライザに適用されます。

`"property": value`

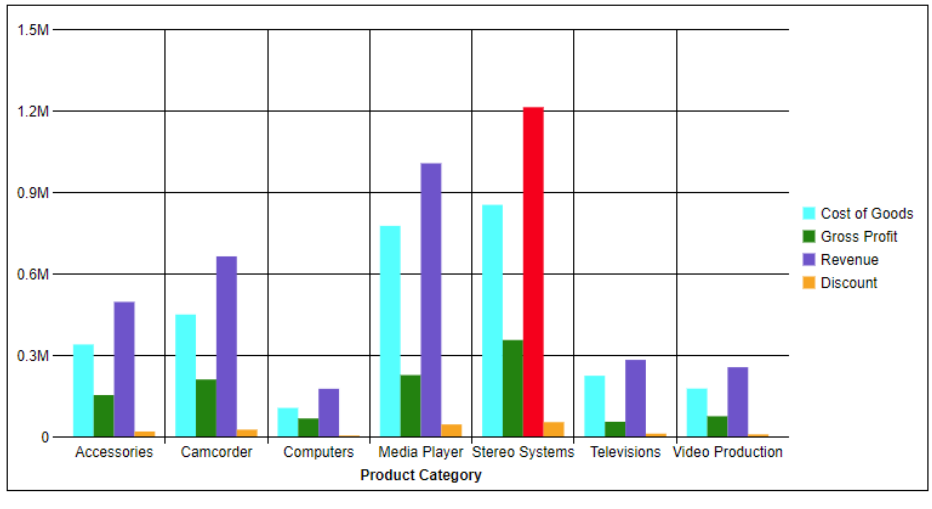
サポートされているシリーズプロパティおよび値です。

例　特定グループの選択

次のリクエストは、縦棒グラフを生成し、各シリーズの色を定義します。シリーズ 2 のグループ 4 には、別の色 (赤色) を定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=Y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "cyan"},
   {"series": 1, "color": "green"},
   {"series": 2, "group": 4, "color": "red"},
   {"series": 2, "color": "slateblue"},
   {"series": 3, "color": "orange"}
   ]
*END
ENDSTYLE
END
```

出力結果では、シリーズ 2 のライザはすべて青色で表示されていますが、シリーズ 2 のグループ 4 のみが赤色で表示されています。



シリーズライザの境界線の定義

border プロパティは、すべてのライザの境界線、特定シリーズのすべてのライザの境界線、またはシリーズ番号とグループ番号で識別されるライザの境界線を定義します。このプロパティは、面グラフのライザ、棒グラフのライザ、じょうごグラフのセグメント、メータグラフのメータ針、円グラフの円項目に使用することができます。

構文　シリーズライザの境界線の定義

```
"series": [
   {
      "series": number,
      "group": number, // optional
      "border": {
         "width": number,
         "color": "string",
         "dash": "string"
      },
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"group": number`

0 (ゼロ) を基準とするグループ番号です (オプション)。選択したグループがグラフに存在しない場合、そのプロパティは無視されます。グループ番号を指定しない場合、指定したシリーズのすべてのライザに境界線が適用されます。

`"border":`

ライザ境界線のプロパティを定義します。

`"width": number`

境界の幅を定義する数値をピクセル単位で定義します。

`"color": "string"`

色名または数値指定の文字列で定義する色です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"dash": "string"

境界線の破線スタイルを定義する文字列です。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定することができます。下表のような線スタイルのパターン名を使用することもできます。

パターン名	説明	サンプル	対応するピクセルプロパティ
"dash"	破線で構成されます。		"dash": "5 4"
"dash_dot"	破線と点が交互に使用されます。		"dash": "5 4 2 4"
"dash_dot_dot"	破線の後に 2 つの点が使用されます。		"dash": "10 4 2 4 2 4"
"dot"	点で構成されます。		"dash": "2 4"
"long_dash"	長破線で構成されます。		"dash": "10 4"
"long_dash_dot"	長破線と点が交互に使用されます。		"dash": "10 4 2 4"
"short_dash"	破線で構成されます。		"5 2"
"short_dash_dot"	破線と点が交互に使用されます。		"dash": "5 2 2 2"
"short_dash_dot_dot"	破線の後に 2 つの点が使用されます。		"dash": "5 2 2 2 2 2"

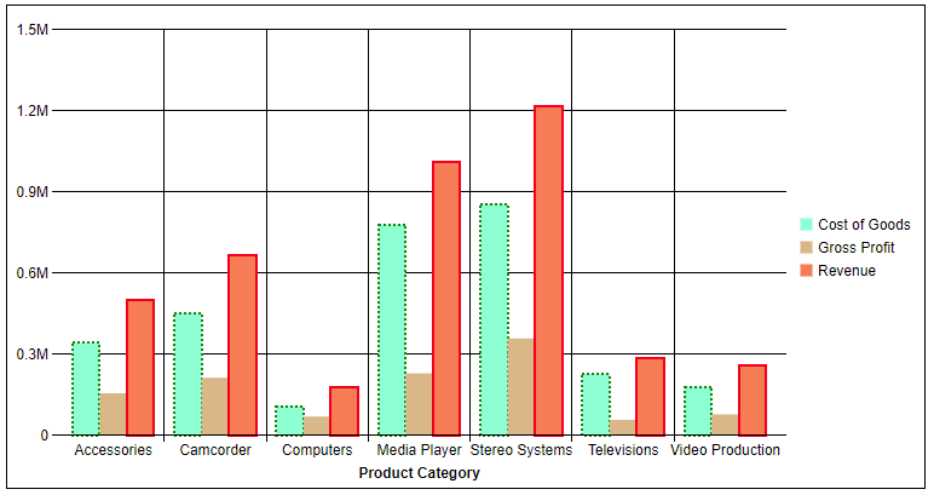
パターン名	説明	サンプル	対応するピクセルプロパティ
"short_dot"	点で構成されます。		"dash": "2 2"
"solid"	実線です。		"dash": ""

例　シリーズライザの境界線の定義

次のリクエストは、棒グラフを生成し、3 つのシリーズを表示します。シリーズ 0 のライザの境界線には緑色の破線、シリーズ 2 のライザの境界線には赤色の実線を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "aquamarine", "border": {
      "width": 2, "color": "green", "dash": "2 2"}},
   {"series": 1, "color": "burlywood"},
   {"series": 2, "color": "coral", "border": {
      "width": 2, "color": "red"}}
   ]
*END
ENDSTYLE
END
```

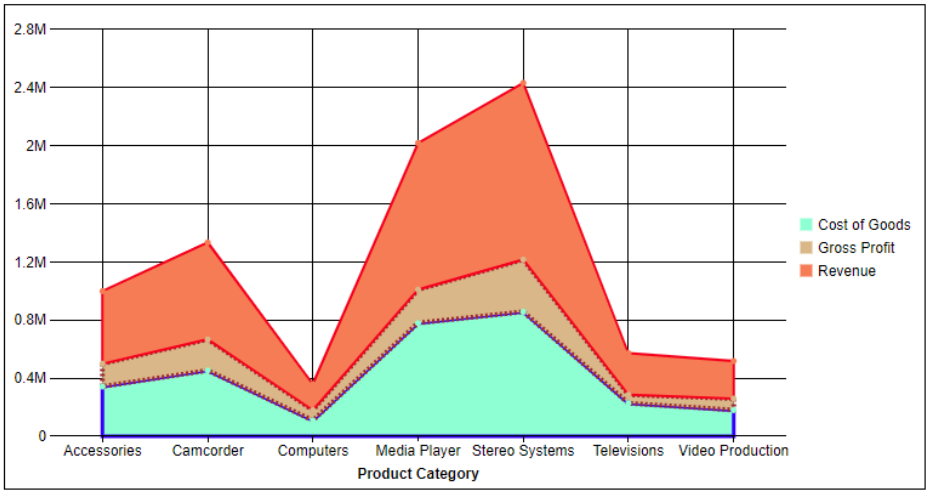
出力結果は次のとおりです。



次のリクエストは、積み上げ面グラフを生成します。シリーズ 0 のライザの境界線には青色の実線、シリーズ 1 のライザの境界線には茶色の破線、シリーズ 2 のライザの境界線には赤色の実線を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH AREA
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-SERIES-LAYOUT=STACKED,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
*GRAPH_JS
"series": [
    {"series": 0, "color": "aquamarine", "border": {
        "width": 3, "color": "blue"}},
    {"series": 1, "color": "burlywood", "border": {
        "width": 4, "color": "brown", "dash": "2 2"}},
    {"series": 2, "color": "coral", "border": {
        "width": 2,"color": "red"}}
    ]
*END
ENDSTYLE
END
```

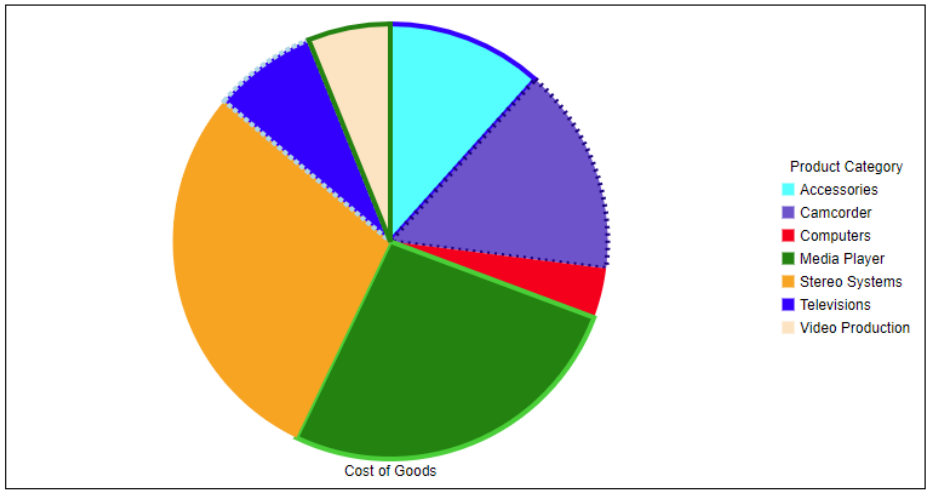
出力結果は次のとおりです。



次のリクエストは、円グラフを生成し、一部の円項目に境界線を定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"series": [
    {"series": 0, "color": "cyan", "border": {
        "width": 4, "color": "blue"}},
    {"series": 1, "color": "slateblue", "border": {
        "width": 4, "color": "navy", "dash": "2 4"}},
    {"series": 2, "color": "red"},
    {"series": 3, "color": "green", "border": {
        "width": 4, "color": "limegreen"}},
    {"series": 4, "color": "orange"},
    {"series": 5, "color": "blue", "border": {
        "width": 4, "color": "lightblue", "dash": "4 2"}},
    {"series": 6, "color": "bisque","border": {
        "width": 4, "color": "green"}}
    ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

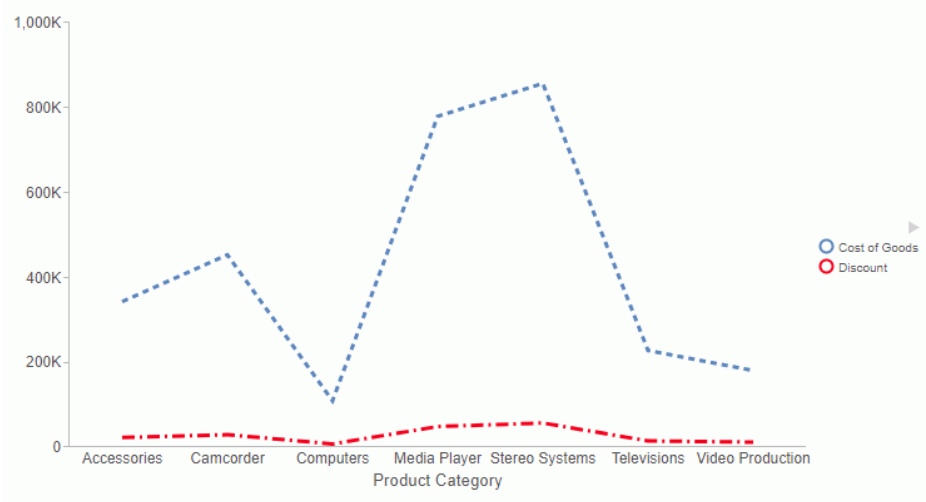


例　線スタイルパターン名の使用

次のリクエストは、シリーズ 0 に "dash" スタイルを使用し、シリーズ 1 に "long_dash_dot" スタイルを使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis,$
TYPE=DATA, COLUMN=DISCOUNT_US, BUCKET=y-axis,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS
"series":[
 {"series":0, "border": {"dash": "dash"}},
 {"series":1, "color": "red", "border": {"dash": "long_dash_dot"}}
 ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



シリーズライザの色の定義

color プロパティは、すべてのライザの色、特定シリーズのすべてのライザの色、またはシリーズ番号とグループ番号で識別されるライザの色を定義します。グループ番号を指定しない場合、定義した色が凡例領域のシリーズアイコンにも適用されます。

構文　シリーズライザの色の定義

```
"series": [
   {
      "series": number,
      "group": number, // Optional
      "color": "string"
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"group": number`

0 (ゼロ) を基準とするグループ番号です (オプション)。選択したグループがグラフに存在しない場合、そのプロパティは無視されます。グループ番号を指定しない場合、定義した色がシリーズ内のすべてのライザに適用されます。

"color": "string"

色名または数値指定の文字列で定義する色、あるいは文字列で定義するグラデーションです。

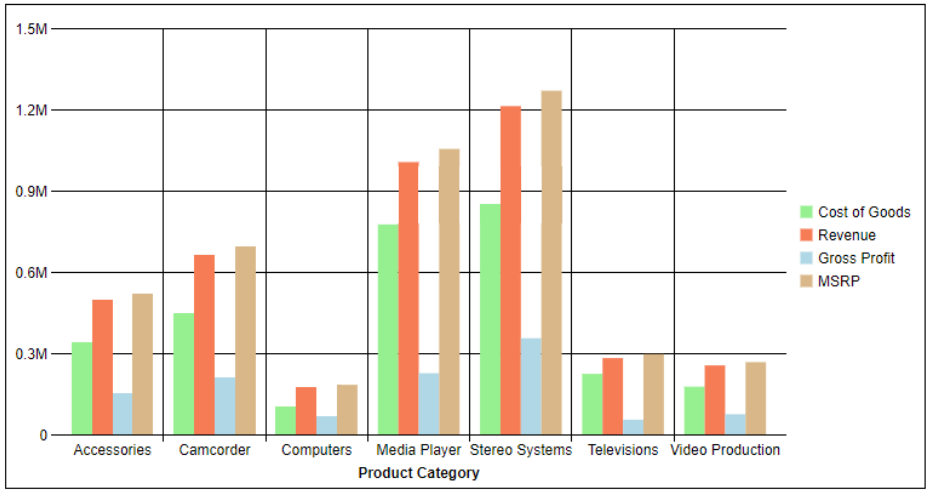
色およびグラデーションの定義についての詳細は、63 ページの「 色およびグラデーション 」 を参照してください。

例　シリーズライザの色の定義

次のリクエストは、縦棒グラフを生成し、各シリーズの色を定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US GROSS_PROFIT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"series": [
    {"series": 0, "color": "lightgreen"},
    {"series": 1, "color": "coral"},
    {"series": 2, "color": "lightblue"},
    {"series": 3, "color": "burlywood"}]
*END
ENDSTYLE
END
```

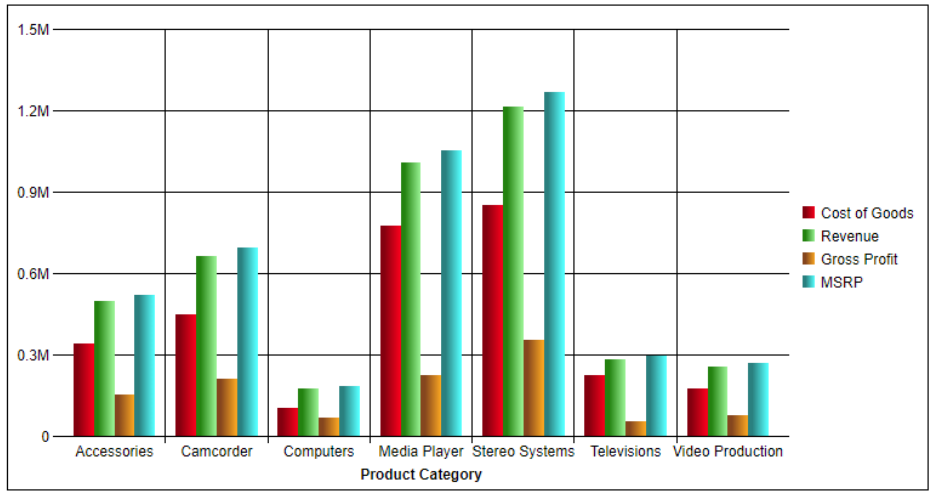
出力結果は次のとおりです。



次のリクエストでは、各シリーズに線状グラデーションを適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US GROSS_PROFIT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"series": [
    {"series": 0, "color":
        "linear-gradient(0%,0%,100%,0%, 20% darkred, 95% red)"},
    {"series": 1, "color":
        "linear-gradient(0%,0%,100%,0%, 20% green, 95% lightgreen)"},
    {"series": 2, "color":
        "linear-gradient(0%,0%,100%,0%, 20% saddlebrown, 95% orange)"},
    {"series": 3, "color":
        "linear-gradient(0%,0%,100%,0%, 20% teal, 95% cyan)"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

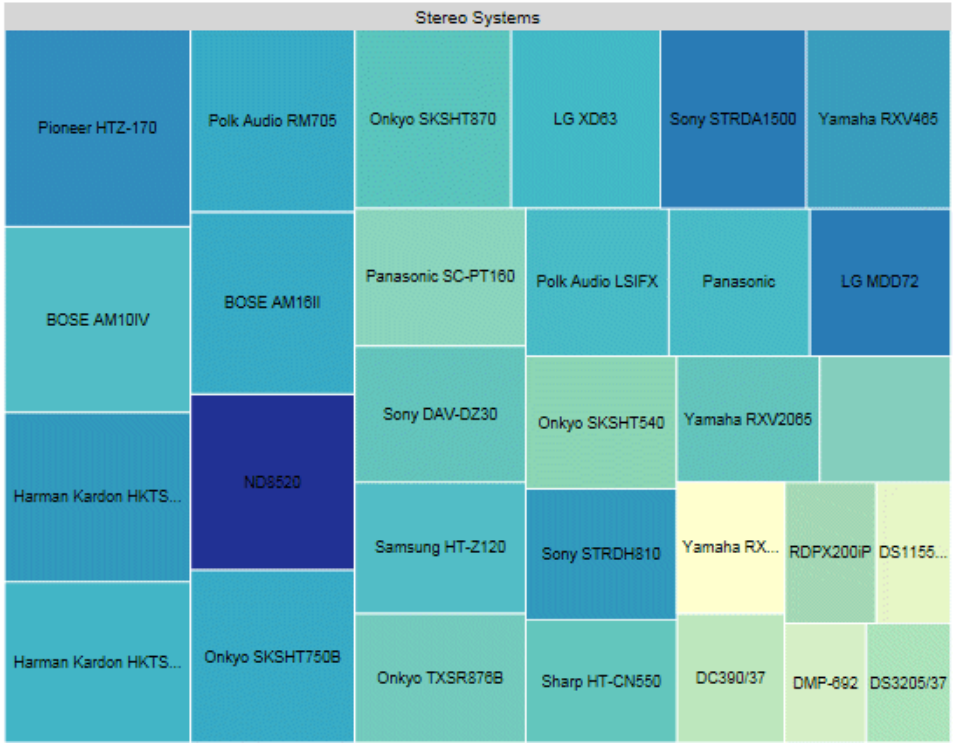


データテキストラベルの表示およびフォーマット設定

dataLabels プロパティは、すべてのシリーズまたは特定のシリーズのデータテキストラベルの表示およびフォーマット設定を制御します。

円グラフで "series": "dataLabels": "position" プロパティを "outside" に設定した場合、feelerLine プロパティを使用して、円グラフとデータテキストラベル間のフィーラ線の外観を制御することができます。

下図のように、ツリーマップのデータラベルが長方形に収まらない場合は、末尾が切り捨てられ、省略記号 (...) が表示されます。



データラベルの一部のプロパティは、シリーズオブジェクトに含まれません。スタンドアロンの dataLabels オブジェクトがこれらのプロパティを制御します。スタンドアロンの dataLabels オブジェクトを使用して、WebFOCUS Reporting Server で設定されたデフォルトのプロパティ値を上書きする例については、109 ページの 「 グラフテンプレートエンジンを使用したツールヒント、タイトル、データラベルのカスタマイズ 」 を参照してください。これらのプロパティは、WebFOCUS スタイルシートの GRAPH_JS_FINAL ブロックに追加する必要があります。

構文　コンテナに収まらないデータラベルの表示の制御

スタンドアロンの dataLabels オブジェクトで制御される clipToContainer プロパティは、コンテナに収まらないデータテキストラベルを表示するかどうかを指定します。デフォルト設定では、これらのデータラベルは描画されません。構文は次のとおりです。

```
"dataLabels": {
    "clipToContainer": boolean
                }
```

説明

`"clipToContainer": boolean`

コンテナに収まらないデータテキストラベルを描画するかどうかを指定します。有効な値には、次のものがあります。

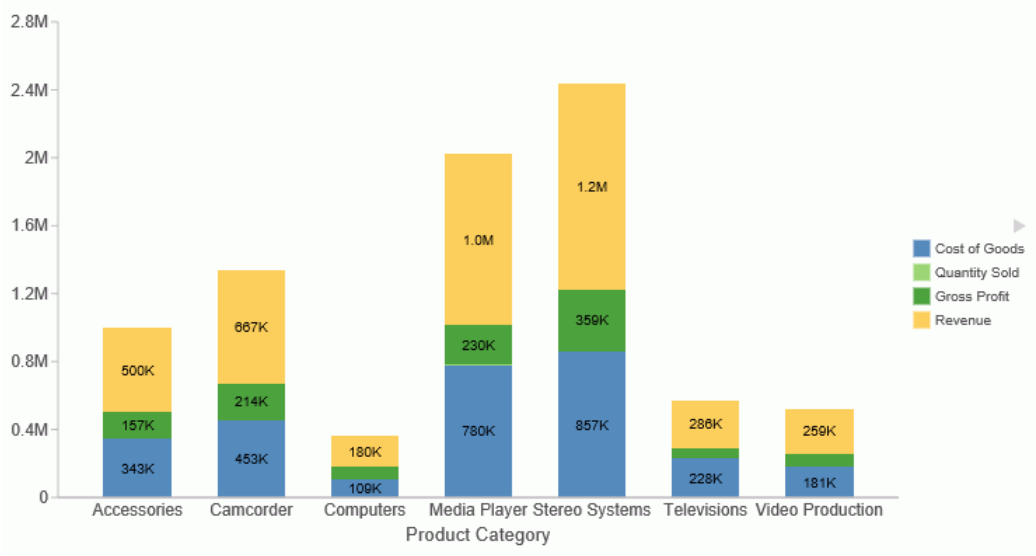
- true - コンテナに収まらないデータテキストラベルを描画しません。これがデフォルト値です。
- false - すべてのデータテキストラベルを描画します。

例　コンテナに収まらないデータラベルの表示の制御

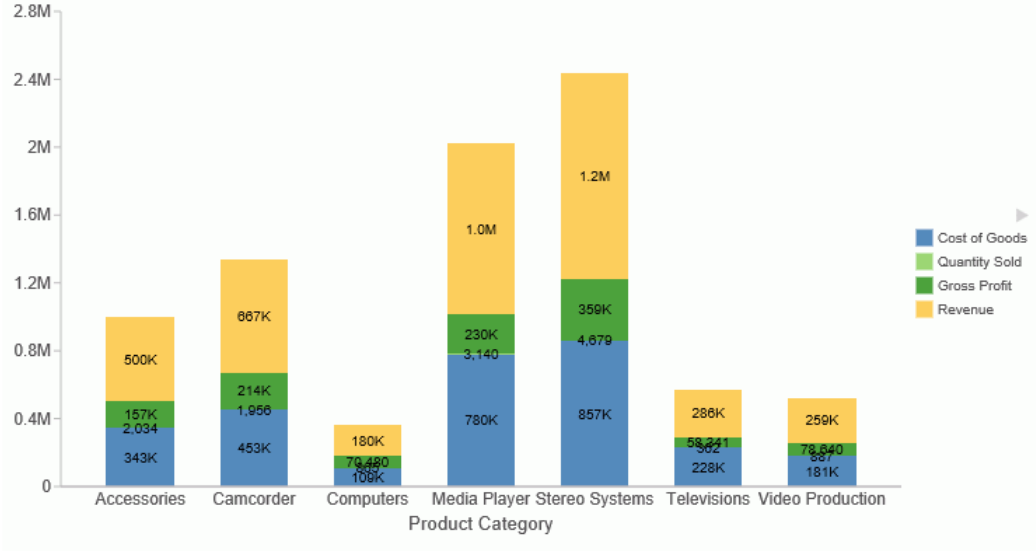
次のリクエストは、積み上げ棒グラフを生成し、clipToContainer をデフォルト設定の true に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US QUANTITY_SOLD GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT OFF
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-SERIES-LAYOUT=stacked,$
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N5, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
*GRAPH_JS
"dataLabels": {"visible":true, "position": "inside",
    "clipToContainer": true}
*END
ENDSTYLE
END
```

下図のように、出力結果では、コンテナに収まらないラベルは省略されています。



下図のように、clipToContainer の設定を false に変更すると、すべてのデータラベルが表示されます。



構文　　シリーズデータテキストラベルの表示およびフォーマット設定

```
"series":[
{"series": s,
   "dataLabels": {
       "visible": boolean,
       "content": "string",
       "content": function(),
       "position": "string",
       "font": "string",
       "color": "string",
       "useNegativeColor": boolean,
       "negativeColor": "string",
       "numberFormat": "numformat",
       "feelerLine": {
          "visible": boolean,
          "width": number,
          "color": "string",
          "dash": "string"
   }
}
} ...
]
```

説明

`s`

シリーズの識別子です。数値、"all"、"reset" のいずれかを指定します。

`"visible": boolean`

次のいずれかの値を指定します。

- true - データテキストラベルを表示します。visible を true に設定した場合、各シリーズのデータラベルを個別に非表示にするには、シリーズ固有の showDataValues プロパティを使用します。
- false - データテキストラベルを表示しません。これがデフォルト値です。
- "auto" - データラベルを表示するかどうかをグラフエンジンが決定します。

`"content": "string"`

データテキストの表示を制御します。有効な値は、"x"、"y"、"z"、"%"、"%+"、"cumulative"、"groupLabel"、"seriesLabel" です。コロプレスマップでは、"name" を使用して地名を表示することができます。デフォルト値は "x" です。

- バブルグラフの場合、"x"、"y"、"z" のいずれかを使用して、表示する値を識別します (X 軸値、Y 軸値、サイズ (z))。

- 折れ線グラフのデータセットにマーカーサイズ値が含まれている場合、"x" または "y" を使用して Y 軸値を表示します。マーカーサイズ値を表示するには、"z" を使用します。
- 散布図の場合、"x" または "y" を使用して、表示する値を識別します (X 軸値、Y 軸値)。
- 円グラフの場合、"%" を使用して円項目のパーセント値を表示することも、"value" (円グラフのデフォルト値) を使用して円項目のメジャー値を表示することもできます。

  円グラフで % モードを使用する場合、値をパーセント値として表示するには、numberFormat プロパティを設定する必要があります (例、"numberFormat": "[>9]#,#%;[<0]-#.#%;[<=9]#.##%;[=0]0")。数値のフォーマット設定についての詳細は、86 ページの 「 数値のフォーマット設定 」 を参照してください。
- 積み上げ棒グラフ、積み上げ折れ線グラフ、積み上げ面グラフの場合 ("blaProperties": {"seriesLayout": "stacked"})、積み上げ合計を表示するには "cumulative"、積み上げ値をパーセントとして表示するには "%"、パーセント値と累計値を表示するには "%+" を使用します。
- Leaflet のコロプレスマップの場合、"name" を使用して地名を表示することができます。例については、871 ページの 「 マップサポート 」 を参照してください。
- また、content プロパティを使用して、複数の値を表示する関数を定義することもできます。

`"content": function()`

各データラベルの呼び出し関数を指定します。デフォルト値は undefined です。 この関数は、最大で 4 つの引数を取ることができます (データ値、シリーズ ID、グループ ID、データオブジェクト)。この関数は、表示される文字列を返します。コールバック関数についての詳細は、102 ページの 「 ラベル値を返すコールバック関数の定義 」 を参照してください。

`"position": "string"`

ライザを基準としたデータラベルの位置を定義する文字列です。有効な値は、"bottom"、"center"、"insideBottom"、"insideTop"、"left"、"outside"、"right"、"top"、"auto"' です。デフォルト値は "top" です。

- "insideBottom" の設定は棒ライザのみに適用され、棒ライザ底辺の内側にデータラベルが描画されます。
- "insideTop" の設定は棒ライザおよび円項目のみに適用され、棒ライザまたは円項目の上端の内側にデータラベルが描画されます。

- 円グラフの場合、有効な設定は "center" (円項目上に表示)、"insideTop" (円項目の上端に表示) および "outside" (円項目の外側にフィーラ線付きで表示) のみです。
- 棒ライザの場合、位置はグラフの方向に対応します (例、"top" は横棒の "right" に対応)。
- ブレットグラフの場合、マーカーの位置を定義するシリーズ固有のプロパティも参照してください。このプロパティは、データテキストの位置を基準にしてマーカーの位置を定義します。
- "auto" 設定を使用すると、ラベルの位置をグラフエンジンが決定します。

注意：JSCHART フォーマットでは、位置の「上端」はサポートされません。

`"font": "string"`

データテキストラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は、"font": "7.5pt Sans-Serif" です。

`"color": "string"`

データテキストラベルの色を定義する文字列です。色名、数値指定の文字列、グラデーション定義の文字列のいずれかを指定することができます。デフォルト値は "black" です。

`"useNegativeColor": boolean`

負の値のライザのデータテキストラベルに別の色を使用するかどうかを制御します。次のいずれかの値を指定します。

- true - 負の値のライザのデータテキストラベルに、negativeColor プロパティで定義された色を割り当てます。
- false - すべてのデータテキストラベルに同一の色を使用します。これがデフォルト値です。

`"negativeColor": "string"`

負の値のライザのデータテキストラベル色を定義する文字列です。色名、数値指定の文字列、グラデーション定義の文字列のいずれかを指定することができます。デフォルト値は "red" です。

`"numberFormat": "numformat"`

データラベルの数値フォーマットです。数値フォーマットは、JSON オブジェクト、フォーマット文字列、ユーザ定義関数のいずれかで指定することができます。 デフォルト値は "auto" です。autoNumberFormats を使用して、データテキストラベルのすべてに数値フォーマットを適用することもできます。数値フォーマットについての詳細は、86 ページの「 数値のフォーマット設定 」 を参照してください。

"feelerLine":
フィーラ線のプロパティを定義します。

"visible": boolean
有効な値には、次のものがあります。

- true - フィーラ線を表示します。これがデフォルト値です。
- false - フィーラ線を表示しません。

"width": number

フィーラ線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

色名または数値指定の文字列でフィーラ線の色を定義します。デフォルト値は "black" です。

色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"dash": "string"

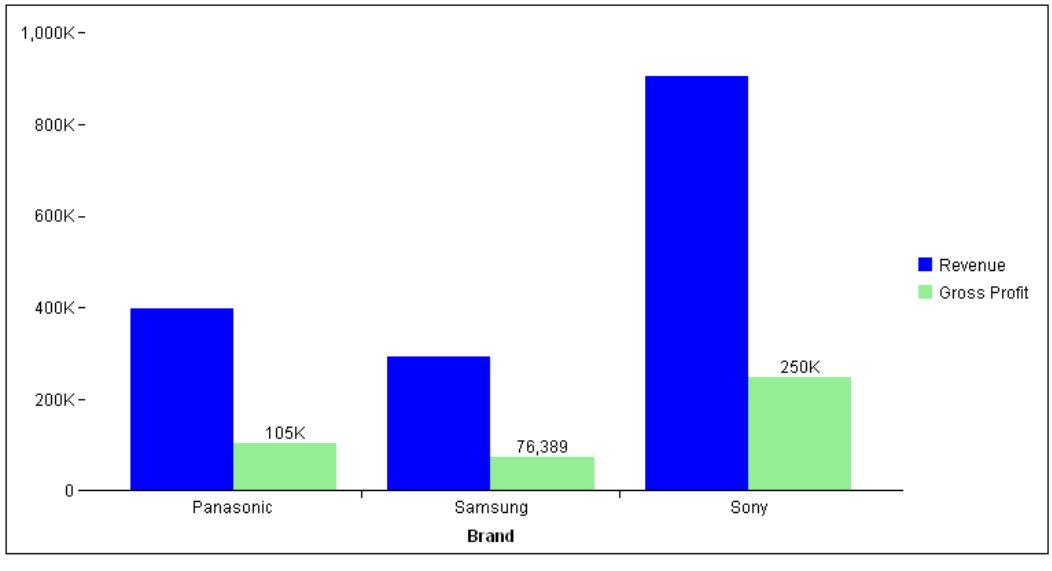
破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　特定シリーズのデータテキストラベルの表示

次のリクエストは、シリーズ 1 のデータテキストラベルを生成します。シリーズ 0 にはデータテキストラベルを生成しません。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY BRAND
WHERE BRAND EQ 'Sony' OR 'Samsung' OR 'Panasonic'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=brand, bucket=x-axis,$
type=data, column=revenue_us, bucket=y-axis,$
type=data, column=gross_profit_us, bucket=y-axis,$
*GRAPH_JS
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"series": [
    {"series": 0, "color": "blue"},
    {"series": 1, "color": "lightgreen", "dataLabels": {"visible": true}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

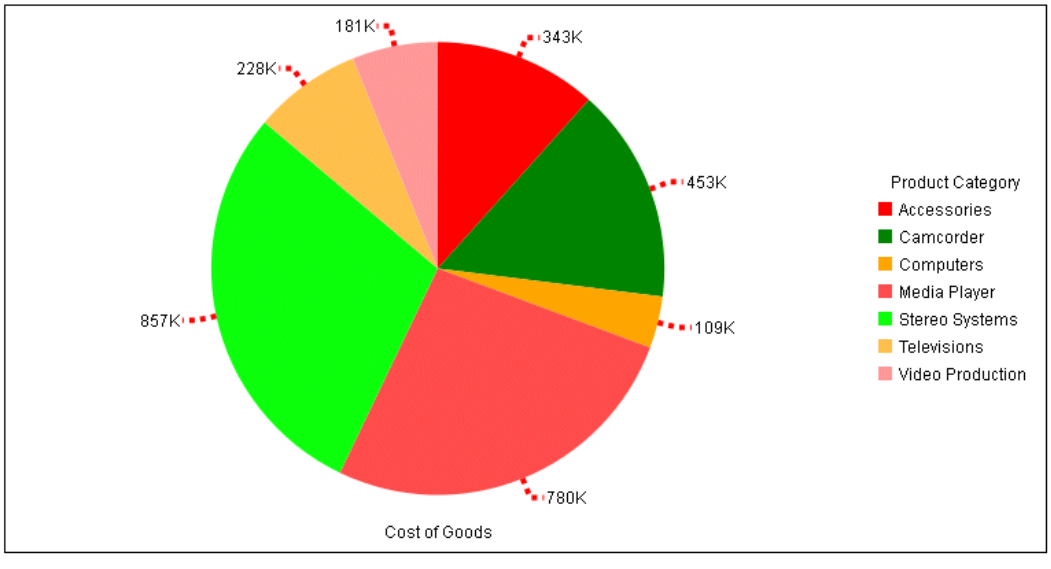


例　円グラフのフィーラ線の表示制御

次のリクエストは、フィーラ線を赤色、破線、4 ポイント幅に設定します。フィーラ線を生成するには、データラベルを円グラフの外側に配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=measure,$
type=data, column=product_category, bucket=color,$
*GRAPH_JS
"series": [{"series": "all", "dataLabels": {
    "position": "outside", "visible": true, "feelerLine": {
        "visible": true, "width": 4, "color": "red", "dash": "4 4"}
    }}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

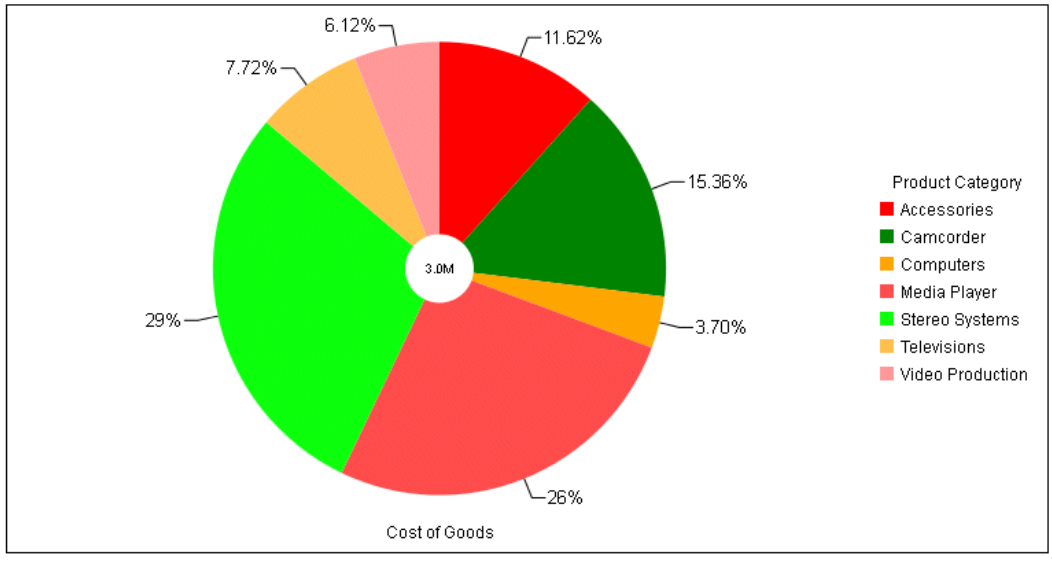


例　円グラフのデータラベルのフォーマット設定

次のリクエストは、円グラフを生成します。データラベルは、円項目の外側に 10pt Sans-Serif フォントのパーセントモードで表示し、データ値に応じてさまざまなフォーマットで表示されるよう設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=measure,$
type=data, column=product_category,bucket=color,$
*GRAPH_JS
"pieProperties": {"totalLabel": {"visible": true}, "holeSize": 25},
"series": [{"series": "all", "dataLabels": {
    "visible": true,
    "position": "outside",
    "feelerLine": {"visible": true, "width": 1, "color": "black"},
    "content": "%",
    "font": "10pt Sans-Serif",
    "numberFormat": "[>.2]#,#%;[<0]-#.#%;[<=.2]#.##%;[=0]0"}}]
*END
ENDSTYLE
END
```

グラフ出力では、データラベルは円の外側に表示され、フィーラ線で接続されています。

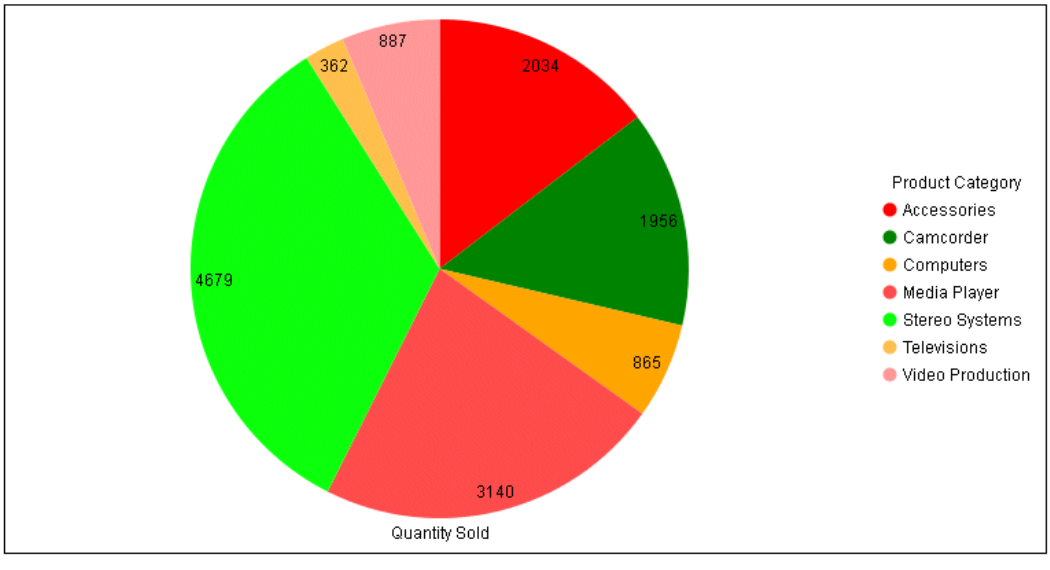


例　円項目の上端内側のデータラベル配置

次のリクエストは、円項目上端の内側にデータラベルを配置した円グラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=product_category, BUCKET=color, $
TYPE=DATA, COLUMN=quantity_sold, BUCKET=measure, $
*GRAPH_JS
"dataLabels": {"visible": true, "position": "insideTop"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

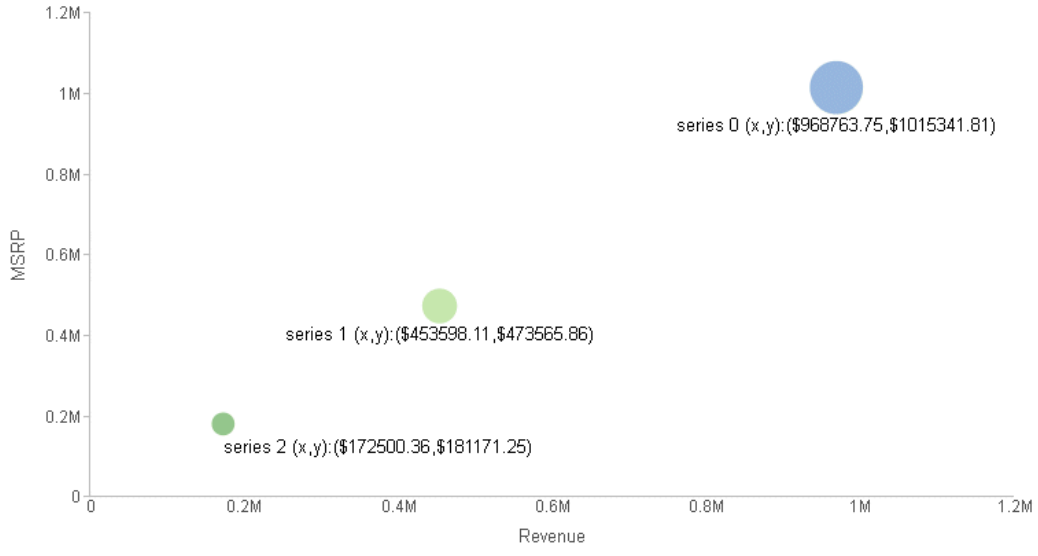


例　コールバック関数によるデータラベルのフォーマット設定

次のリクエストでは、データラベル表示を有効にし、データラベルをバブルの下側に配置します。また、ラベルを返す関数を使用しています。この関数は、引数としてデータ値およびシリーズ番号を取り、ラベルを返します。ラベルには、指定された接頭語とともに、データのシリーズ番号およびデータの x,y 値を表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US MSRP_US DISCOUNT_US
BY PRODUCT_SUBCATEG
WHERE PRODUCT_SUBCATEG EQ 'iPod Docking Station' OR 'Blu Ray' OR 'Speaker Kits'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=x-axis,$
type=data, column=msrp_us, bucket=y-axis,$
type=data, column=discount_us, bucket=size,$
type=data, column=product_subcateg, bucket=color,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"legend": {"visible": "false"},
"series": [{"series": "all", "dataLabels": {
    "visible": true,
    "position": "bottom",
    "font": "10pt Sans-Serif",
    "content": function(d, s)
        {return "series " + s+" (x,y):" +"("+"$"+d.x+","+"$"+d.y+")";
    }}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

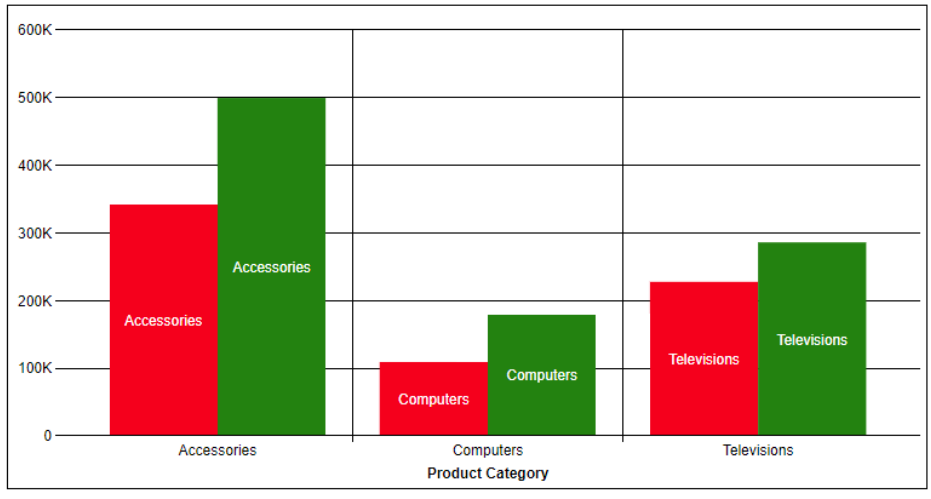


例　グループラベルまたはシリーズラベルをデータラベルとして使用

次のリクエストでは、グループラベルをデータテキストとして使用し、これらのラベルをライザの内側に配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Accessories' OR 'Computers' OR 'Televisions'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": false},
"series": [{"series": "all", "dataLabels": {
   "visible": true, "color": "white",
   "position": "inside", "content": "groupLabel"}
   }]
*END
ENDSTYLE
END
```

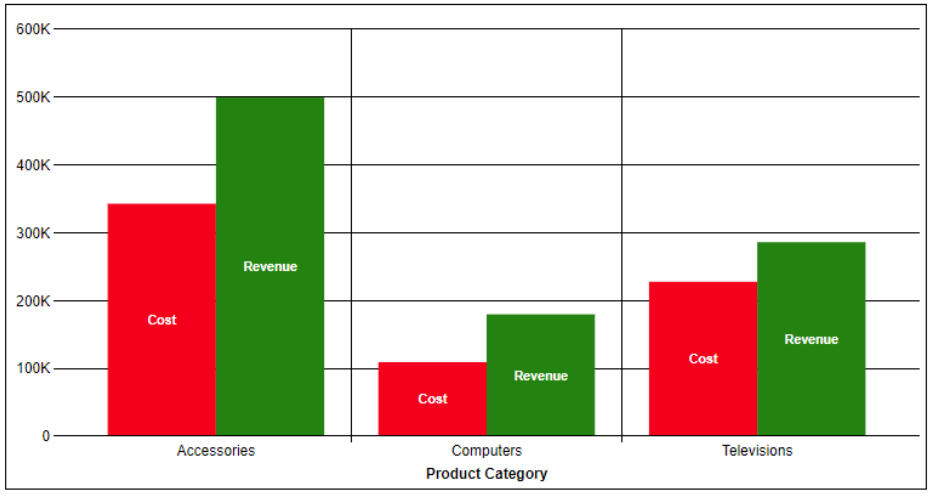
出力結果では、グループ (ソートフィールド) のラベル (製品区分) がデータラベルとして使用されています。



次のリクエストでは、シリーズラベルをデータテキストとして使用します。シリーズは、リクエストの表示フィールドから生成されます。このリクエストでは、デフォルトのフィールド見出しが AS 名で置き換えられます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US AS Cost
REVENUE_US AS Revenue
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Accessories' OR 'Computers' OR 'Televisions'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": false},
"series": [{"series": "all", "dataLabels": {
   "visible": true, "color": "white", "font": "Bold 8pt Sans-Serif",
   "position": "inside", "content": "seriesLabel"}}]
*END
ENDSTYLE
END
```

出力結果では、シリーズ (表示フィールド) の AS 名がデータラベルとして使用されています。

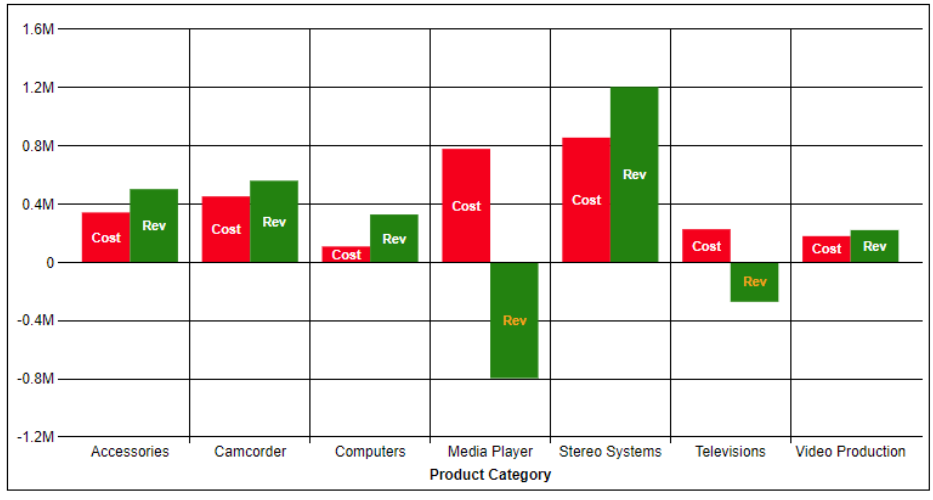


例　負のライザのデータラベルに別の色を使用

次のリクエストでは、シリーズラベルをデータテキストとして使用し、負の値のライザのデータテキストをオレンジ色に設定します。シリーズは、リクエストの表示フィールドから生成されます。このリクエストでは、デフォルトのフィールド見出しが AS 名で置き換えられます。

```
DEFINE FILE WF_RETAIL_LITE
REV= IF (COGS_US * 1.3) GT REVENUE_US THEN (REVENUE_US * -3) ELSE
(REVENUE_US *2) ;
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US AS Cost
REV AS Rev
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": false},
"series": [{"series": "all", "dataLabels": {
   "visible": true, "color": "white",
   "useNegativeColor": true, "negativeColor": "orange",
   "font": "Bold 8pt Sans-Serif", "position": "inside", "content":
"seriesLabel"}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



円グラフからの円項目の削除

deleteSlice プロパティは、円グラフから円項目を削除します。実際には、円項目に透明色が割り当てられます。

構文　　円グラフからの円項目の削除

```
"series":
[
    {
      "series": number,
      "group": number,
      "deleteSlice": boolean,
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準としたシリーズ番号です。"all" を指定すると、すべての円項目が削除されます。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"group": number`

複数円グラフの場合、0 (ゼロ) を基準としたグループ (円グラフ) 番号です (オプション)。選択したグループがグラフに存在しない場合、そのプロパティは無視されます。グループ番号を指定しない場合、すべての円グラフから円項目が削除されます。

`"deleteSlice": boolean`
次のいずれかの値を指定します。

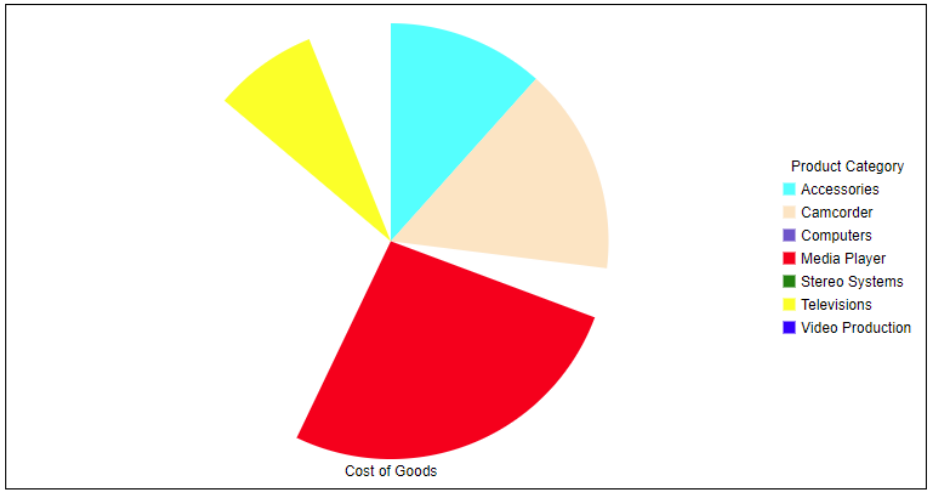
- true - 円項目を削除します。
- false - 円項目を元に戻します。これがデフォルト値です。

例　円項目の削除

次のリクエストは、円グラフを生成し、シリーズ 2、4、6 の円項目を削除します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "cyan"},
   {"series": 1, "color": "bisque"},
   {"series": 2, "color": "slateblue", "deleteSlice": true},
   {"series": 3, "color": "red"},
   {"series": 4, "color": "green", "deleteSlice": true},
   {"series": 5, "color": "yellow"},
   {"series": 6, "color": "blue", "deleteSlice": true}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



円グラフからの円項目の分離

explodeSlice プロパティは、円グラフから円項目を分離する距離をピクセル単位で定義します。

構文　円グラフからの円項目の分離

```
series:
[
    {
      "series": number,
      "group": number,
      "explodeSlice": "value"   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"group": number`

複数円グラフの場合、0 (ゼロ) を基準としたグループ (円グラフ) 番号です (オプション)。選択したグループがグラフに存在しない場合、そのプロパティは無視されます。グループ番号を指定しない場合、このプロパティがすべての円グラフに適用されます。

```
"explodeSlice": "value"
```

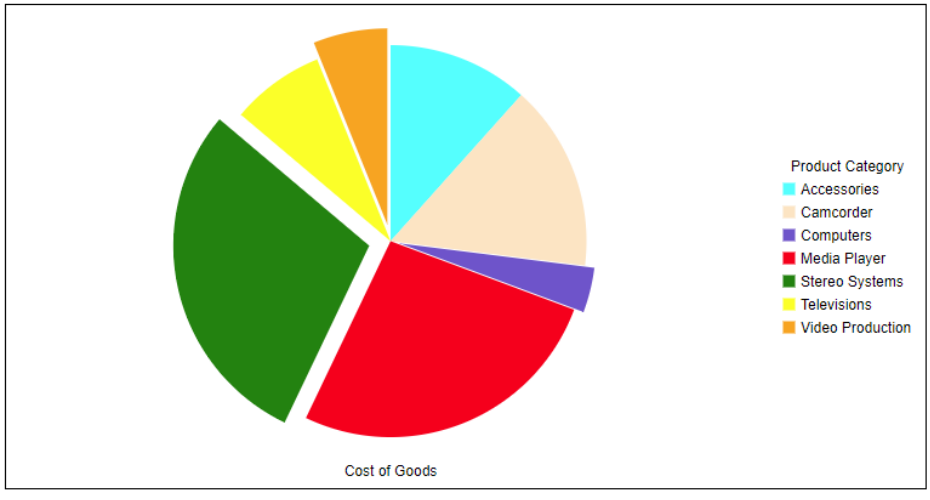
円グラフから円項目を分離する距離 (ピクセル単位) を定義する数値、または円の半径に対する割合を示すパーセント文字列を指定します。この文字列は、パーセント記号 (%) を含めて二重引用符 (") で囲みます (例、"25%")。たとえば、円の直径が 200 ピクセルの場合 (半径 100 ピクセル)、「"explodeSlice": "50%"」と指定すると、円項目が円の中心から 50 ピクセル外側へ分離されます。

例　円グラフからの円項目の分離

次のリクエストは、円グラフを作成し、シリーズ 2、4、6 の円項目を分離します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "cyan"},
   {"series": 1, "color": "bisque"},
   {"series": 2, "color": "slateblue", "explodeSlice": 8},
   {"series": 3, "color": "red"},
   {"series": 4,"color": "green", "explodeSlice": 18},
   {"series": 5, "color": "yellow"},
   {"series": 6, "color": "orange", "explodeSlice": 14}]
*END
ENDSTYLE
END
```

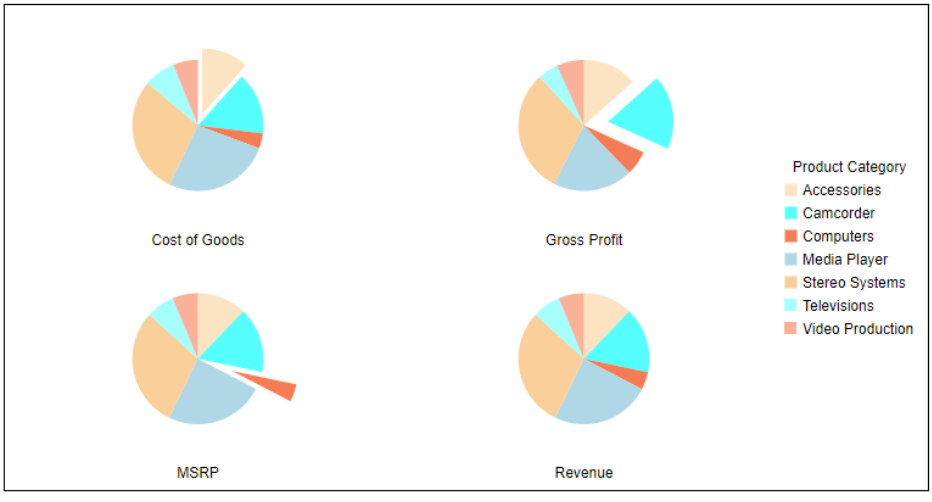
出力結果は次のとおりです。



次のリクエストは、複数円グラフを生成し、円グラフ 0 のシリーズ 0、円グラフ 1 のシリーズ 1、円グラフ 2 のシリーズ 2 の円項目をそれぞれ分離します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US MSRP_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
type=data, column=n3, bucket=measure,$
type=data, column=n4, bucket=measure,$
type=data, column=n5, bucket=measure,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "bisque"},
   {"series": 0, "group": 0, "explodeSlice": 10},
   {"series": 1, "color": "cyan"},
   {"series": 1, "group": 1, "explodeSlice": 20},
   {"series": 2, "color": "coral"},
   {"series": 2, "group": 2, "explodeSlice": 30},
   {"series": 3, "color": "lightblue"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



特定シリーズへのラベルの割り当て

label プロパティは、特定のシリーズにラベルを割り当てます。このラベルは、グラフの凡例領域に表示されます。シリーズラベルのフォーマットおよび色を定義するには、legend:labels プロパティを使用します。

構文　シリーズラベルの割り当て

```
"series": [
   {
      "series": number,
      "label": "string",
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"label": "string"`

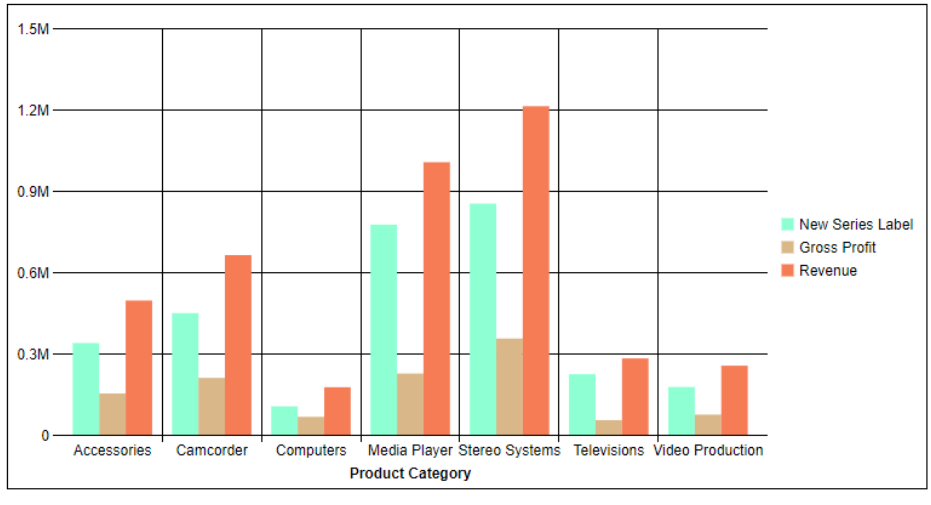
シリーズラベルを定義する文字列です。

例　シリーズラベルの割り当て

次のリクエストは、縦棒グラフを生成し、シリーズ 0 に新しいラベルを割り当てます。新しいラベルは、グラフ右側の凡例領域に表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
*GRAPH_JS_FINAL
"legend": {"position": "right"},
"series": [
   {"series": 0, "color": "aquamarine", "label": "New Series Label"},
   {"series": 1, "color": "burlywood"},
   {"series": 2, "color": "coral"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



シリーズマーカーのサイズ、境界線、色、形状、回転の定義

marker プロパティは、シリーズマーカーのサイズ、境界線、色、形状、回転を定義します。

構文　シリーズマーカーのサイズ、境界線、色、形状、回転の定義

```
"series": [
   {
      "series": number,  "color": "string",
      "group": number,
      "marker": {
         "visible": boolean,  // Line Charts Only
         "size": number,
         "shape": "string",
         "rotation": number,
         "position": "string",
         "fillEffect": "string",
         "border": {"width": number, "color": "string", "dash": "string"}
      }
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"group": number`

0 (ゼロ) を基準とするグループ番号です (オプション)。選択したグループがグラフに存在しない場合、そのプロパティは無視されます。グループ番号を指定しない場合、指定したマーカー定義がシリーズ内のすべてのライザに適用されます。

`"visible": boolean`

マーカーの表示を制御します (折れ線グラフの場合のみ)。有効な値には、次のものがあります。

- true - マーカーを表示します。これがデフォルト値です。
- false - マーカーを表示しません。

`"color": "string"`

マーカーの色を色名または数値指定の文字列で定義するか、グラデーションを定義する文字列で指定します。

色およびグラデーションの定義についての詳細は、63 ページの「 色およびグラデーション 」 を参照してください。

`"size": number または "size": "n%"`

マーカーのサイズをピクセル単位で指定する数値またはパーセント文字列です。パーセントは、グラフフレーム内部の幅と高さの最小値を基準に計算されます。

"shape": "string"

シリーズのマーカー形状を定義する文字列です。次の標準形状のいずれかを使用できます。また、グラフテンプレートエンジンを使用して、シリーズまたはグループごとのイメージを指定する URL を参照し、これらのイメージをマーカー形状として表示することもできます。

有効な標準形状の値には次のものがあります。

- "arrow"
- "bar"
- "circle"
- "circleMinus" (円の中心にマイナス記号 (-) が配置された丸マーカー)
- "circlePlus" (円の中心にプラス記号 (+) が配置された丸マーカー)
- "cross"
- "diamond"
- "fill" (マトリックスマーカーグラフで使用可能。各セルをマーカー色で塗りつぶし、ヒートマップに類似した外観にします)
- "fiveStar"
- "hexagon"
- "hourglass"
- "house"
- "pie" (バブルグラフおよび散布図のみ)
- "pin"
- "pirateCross"
- "plus"
- "rectangle"
- "rectangleThin"
- "ring"
- "sixStar"

- "square" (これがデフォルト値です)
- "thinPlus"
- "tick"
- "triangle"

bar、circlePlus、circleMinus、cross、tick マーカーを使用する場合は、境界線の幅と色を指定する必要があります。

次の構文は、グラフテンプレートエンジンを使用して、フィールドに格納された URL 参照で指定するイメージを取得します。

```
"url:{{macro}}"
```

macro は、シリーズまたはグループのマーカーとして URL 参照で指定するイメージを表示します。macro には、URL 参照を格納するフィールドが割り当てられたテンプレートエンジンマクロを指定することも仮想属性カテゴリを指定することもできます。

`"rotation": number`

マーカーの回転角度 (度数) を定義する 0 (ゼロ) から 360 までの数値です。

`"position": "string"`

データテキストの位置を基準にしてマーカーの相対位置を定義します (ブレットグラフのみ)。有効な値には、次のものがあります。

- "bottom" - データテキストラベルをマーカーの上側に描画します。
- "middle" - データテキストラベルを全般プロパティの "dataLabels": "position" に基づいて描画します。
- "top" - データテキストラベルをマーカーの下側に描画します。

`"fillEffect": "string"`

マーカーの塗りつぶし効果を定義します。有効な値には、次のものがあります。

- マーカー塗りつぶし色の透明度を定義するパーセント文字列。デフォルト値は "100%" (完全な不透明) です。
- undefined - 現在のマーカー境界色および塗りつぶし色を使用します。
- "seriesHollow" - マーカーの塗りつぶし領域を白抜き (透明) にし、マーカーの境界線には "series": "color" で設定された色を使用します。境界線には 0 (ゼロ) 以外の幅を定義する必要があります。

- "seriesFill" - マーカーの境界線を描画せず、塗りつぶし領域には "series": "color" プロパティで設定された色を使用します。
- "seriesWhite" - マーカーの塗りつぶし領域を白色にし、マーカーの境界線には "series": "color" プロパティで設定された色を使用します。境界線には 0 (ゼロ) 以外の幅を定義する必要があります。
- "seriesAuto" - グラフタイプに応じて、グラフエンジンが次の塗りつぶしプロパティでマーカーを塗りつぶします。

  "seriesHollow" - 散布図およびレーダーグラフで使用されます。

  "seriesLighten" - バブルグラフ、マップグラフ、極グラフで使用されます。

  "seriesWhite" - 折れ線グラフで使用されます。

  "seriesFill" - その他のグラフタイプに適用され、塗りつぶし色にはシリーズ色が使用されます。

`"border":`

マーカー境界線のプロパティを定義します。

`"width": number`

境界の幅を定義する数値をピクセル単位で定義します。

`"color": "string"`

色名または数値指定の文字列でマーカー境界色を定義します。

`"dash": "string"`

境界線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。

注意

- bar、circlePlus、circleMinus、cross、tick マーカー形状を使用する場合は、境界線の幅と色を指定する必要があります。
- 特別な pie マーカー形状はバブルグラフおよび散布図でのみ使用し、"series": "all', {"marker": {"shape": "pie"}} を使用する必要があります。詳細は、513 ページの 「 グラフ固有のプロパティ 」 で、バブルグラフおよび散布図のプロパティを参照してください。
- 散布図のマーカーを接続することはできません。
- リングマーカーには任意のサイズを指定できますが、リングの厚みは外観的に優れたパーセントにハードコードされます。中心は透明ですが、リングの marker fill プロパティはサポートされます。

- rectangleThin マーカーは目盛りに類似していますが、サイズを変更することができ、marker fill および rotation プロパティをサポートします (マーカーは、回転角度を 90 度に設定して横方向から縦方向に変更することができます)。

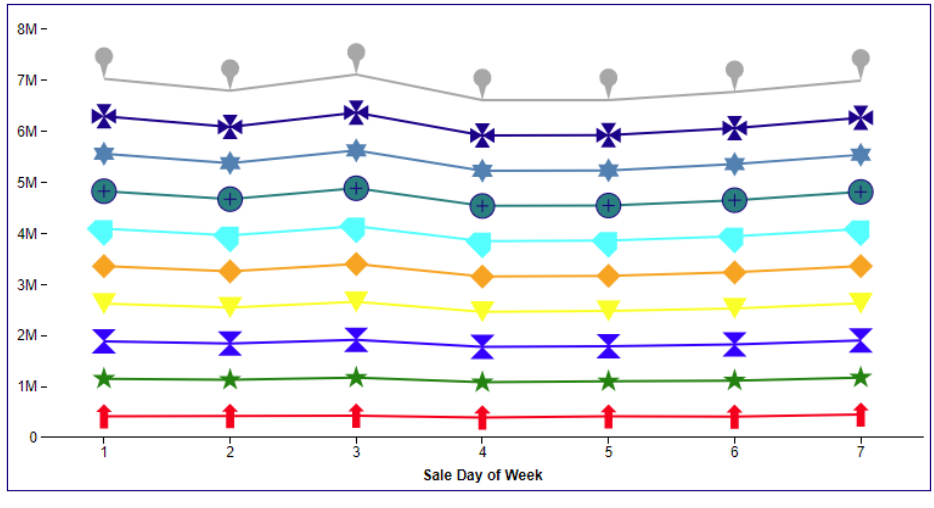
例　シリーズマーカーの定義

次のリクエストは、折れ線グラフを生成し、さまざまなマーカー形状を定義します。

```
DEFINE FILE WF_RETAIL_LITE
COGS1 = COGS_US;
COGS2 = COGS1 +500;
COGS3 = COGS1 +1000;
COGS4 = COGS1 +1500;
COGS5 = COGS1 +2000;
COGS6 = COGS1 +2500;
COGS7 = COGS1 +3000;
COGS8 = COGS1 +3500;
COGS9 = COGS1 +4000;
COGS10 = COGS1 +4500;
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
COGS2 COGS3 COGS4 COGS5 COGS6 COGS7 COGS8 COGS9 COGS10
BY TIME_DAYOFWEEK
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
type=data, column=n7, bucket=y-axis,$
type=data, column=n8, bucket=y-axis,$
type=data, column=n9, bucket=y-axis,$
type=data, column=n10, bucket=y-axis,$
type=data, column=n11, bucket=y-axis,$
```

```
*GRAPH_JS
"border": {"color": "navy"},
"legend": {"visible": false},
"xaxis": {"majorGrid": {"visible": false}, "labels": {"rotation": 0}},
"yaxis": {"majorGrid": {"visible": false}},
"series": [
   {"series": 0, "color": "red", "marker": {"shape": "arrow", "size": 20}},
   {"series": 1, "color": "green", "marker": {"shape": "fiveStar", "size":
20}},
   {"series": 2, "color": "blue", "marker": {"shape": "hourglass", "size":
20}},
   {"series": 3, "color": "yellow", "marker": {"shape": "triangle", "size":
20}},
   {"series": 4, "color": "orange", "marker": {"shape": "diamond", "size":
20}},
   {"series": 5, "color": "cyan", "marker": {"shape": "house", "size":
20,"rotation": 45}},
   {"series": 6, "color": "teal", "marker": {
      "shape": "circlePlus", "size": 20, "border": {"width": 1, "color":
"navy"}}},
   {"series": 7, "color": "steelblue", "marker": {"shape": "sixStar",
"size": 20}},
   {"series": 8, "color": "darkblue", "marker": {"shape": "pirateCross",
"size": 20}},
   {"series": 9, "color": "darkgrey", "marker": {"shape": "pin", "size":
50}}]
*END
ENDSTYLE
END
```

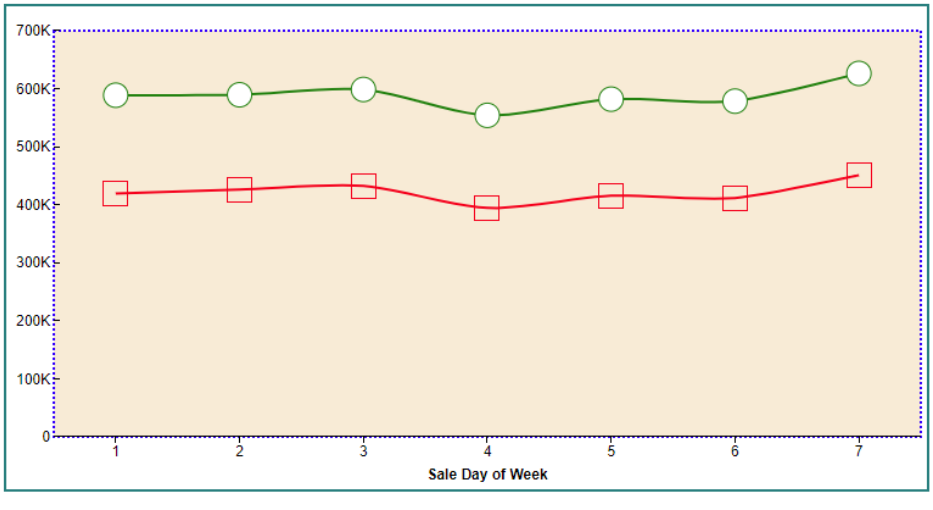
出力結果では、house マーカー形状が 45 度回転され、circlePlus 形状マーカーには必須条件の境界幅と色が定義されています。



次のリクエストでは、シリーズ 0 の塗りつぶし効果が seriesHollow に設定され、シリーズ 1 の塗りつぶし効果が seriesWhite に設定されています。いずれの場合も、境界幅を定義する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY TIME_DAYOFWEEK
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": false},
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"border": {"width": 2, "color": "teal"},
"chartFrame": {"fill": {"color": "antiquewhite"},
   "border": {"width": 2, "color": "blue", "dash": "2 2"}},
"blaProperties": {"lineConnection": "curved"},
"series": [
   {"series": 0, "color": "red", "marker": {
      "border": {"color": "red", "width": 1},
      "shape": "square", "size": 20, "fillEffect": "seriesHollow"}},
   {"series": 1, "color": "green", "marker": {
      "border": {"color": "green", "width": 1},
      "shape": "circle", "size": 20, "fillEffect": "seriesWhite"}}]
*END
ENDSTYLE
END
```

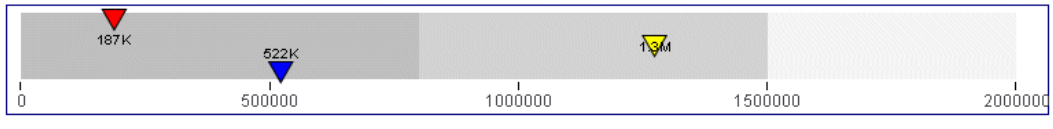
出力結果は次のとおりです。



次のリクエストは、ブレットグラフを生成し、マーカーの位置と形状を設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MSRP_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ  'Accessories' OR 'Computers' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VAXIS 80
ON GRAPH SET LOOKGRAPH CUSTOM
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"chartType": "bullet",
"border": {"width": 1, "color": "navy"},
"bulletProperties": {"drawFirstValueAsBar": false},
"dataLabels": {"visible": true, "font": "6pt"},
"yaxis": {"colorBands": [
    {"start": 0, "stop": 800000, "color": "silver"},
    {"start": 800000, "stop": 1500000, "color": "lightgrey"},
    {"start": 1500000, "stop": 2000000, "color": "whitesmoke"}]},
"series": [
    {"series": "all", "showDataValues": true},
    {"series": 0, "group": 0, "color": "blue",
    "marker": {"size": 15, "position": "bottom", "shape": "triangle"}},
    {"series": 0, "group": 1, "color": "red",
    "marker": {"size": 15, "position": "top", "shape": "triangle"}},
    {"series": 0, "group": 2, "color": "yellow",
    "marker": {"size": 15, "position": "middle", "shape": "triangle"}}]
*END
ENDSTYLE
END
```

出力結果では、黄色のマーカーがデータテキストの中央、赤色のマーカーがデータテキストの上、青色のマーカーがデータテキストの下に配置されています。

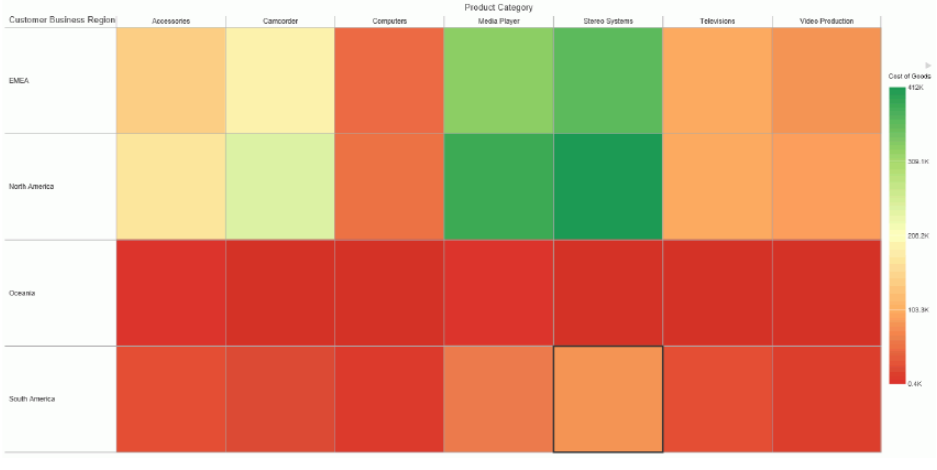


例　マトリックスマーカーグラフのセルの塗りつぶし

次のリクエストは、マトリックスマーカーグラフを生成し、fill マーカー形状を使用します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH MARKER
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=row, $
TYPE=DATA, COLUMN=N2, BUCKET=column, $
TYPE=DATA, COLUMN=N3, BUCKET=color, $
*GRAPH_JS
"series": [
           {
            "series": "all",
                 "marker": {
                             "shape": "fill"
                           }
            },
         ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

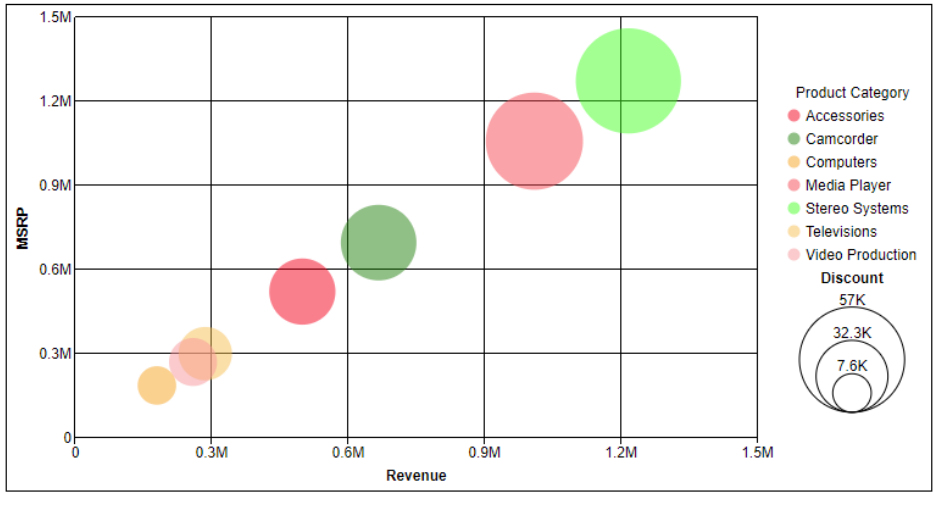


例　バブルマーカーの半透明化

次のリクエストでは、"fillEffect": "50%" プロパティを使用して、バブルマーカーを半透明にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US MSRP_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"series": [
    {"series": "all", "marker": {"shape": "circle", "fillEffect": "50%"}}]
*END
ENDSTYLE
END
```

下図のように、出力結果では、バブルマーカーが 50% の透明度で表示されています。

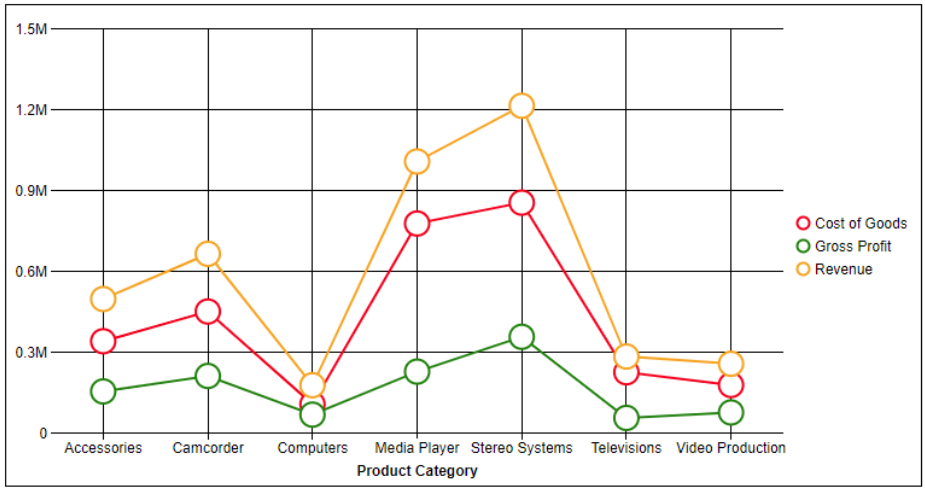


例　マーカーの自動塗りつぶし効果の使用

次のリクエストは、折れ線グラフを生成し、"seriesAuto" 塗りつぶし効果を適用します。折れ線グラフの場合、この塗りつぶし効果は "seriesWhite" と同一です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
*GRAPH_JS
"series": [
   {"series": "all", "marker": {"border": {"width": 2},
      "size": 20, "fillEffect": "seriesAuto"}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



例　URL イメージを使用したマーカー形状

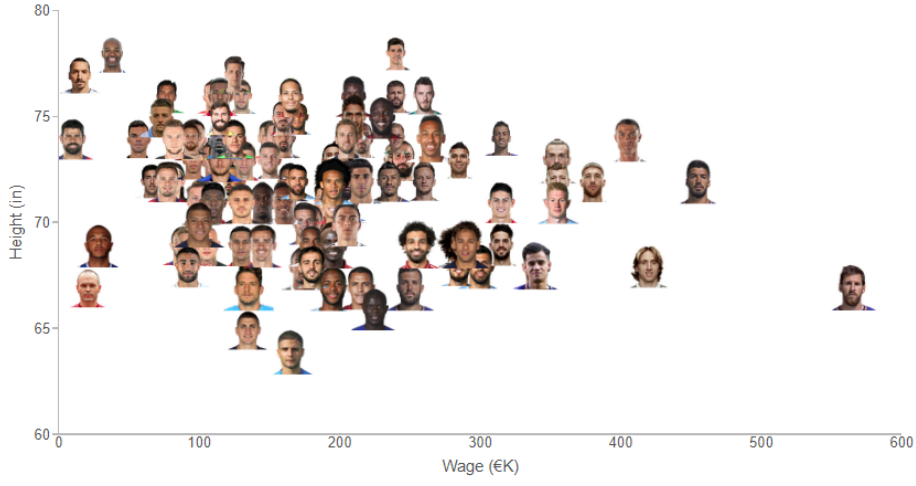
次の例は、サッカーに関するデータを格納する .csv ファイルを使用します。このファイルは、オンラインで一般公開されており、Reporting Server にアップロードしたものです。以下は、生成されたマスターファイルのリストの一部です。

```
FILENAME=FIFI_PLAYERS_2019_TOPRATED, SUFFIX=DIREXCEL,
 DATASET=baseapp/fifi_players_2019_toprated.xlsx, $
  SEGMENT=FIFI_PLAYERS_2019_TOPRATED, SEGTYPE=S0, $
    FIELDNAME=PLAYER_ID, ALIAS='Player ID', USAGE=I9, ACTUAL=A11V,
      MISSING=ON,
      TITLE='Player ID', $
    FIELDNAME=NAME, ALIAS=Name, USAGE=A19V, ACTUAL=A19V,
      MISSING=ON,
      TITLE='Name', $
    FIELDNAME=AGE, ALIAS=Age, USAGE=I5, ACTUAL=A11V,
      MISSING=ON,
      TITLE='Age', $
    FIELDNAME=PHOTO, ALIAS=Photo, USAGE=A46V, ACTUAL=A46V,
      MISSING=ON,
      TITLE='Photo', $
    FIELDNAME=NATIONALITY, ALIAS=Nationality, USAGE=A18V, ACTUAL=A18V,
      MISSING=ON,
      TITLE='Nationality', $
    FIELDNAME=FLAG, ALIAS=Flag, USAGE=A36V, ACTUAL=A36V,
      MISSING=ON,
      TITLE='Flag', $
    FIELDNAME=OVERALL, ALIAS=Overall, USAGE=I5, ACTUAL=A11V,
      MISSING=ON,
      TITLE='Overall', $
    FIELDNAME=POTENTIAL, ALIAS=Potential, USAGE=I5, ACTUAL=A11V,
      MISSING=ON,
      TITLE='Potential', $
    FIELDNAME=CLUB, ALIAS=Club, USAGE=A30V, ACTUAL=A30V,
      MISSING=ON,
      TITLE='Club', $
    FIELDNAME=CLUB_LOGO, ALIAS='Club Logo', USAGE=A47V, ACTUAL=A47V,
      MISSING=ON,
      TITLE='Club Logo', $
    FIELDNAME=VALUE__€M_, ALIAS='Value (€M)', USAGE=D8.1, ACTUAL=A64V,
      MISSING=ON,
      TITLE='Value (€M)', $
    FIELDNAME=WAGE__€K_, ALIAS='Wage (€K)', USAGE=D8.1, ACTUAL=A64V,
      MISSING=ON,
      TITLE='Wage (€K)', $
    ...
```

次のリクエストは、バブルグラフを生成し、PHOTO フィールド値で指定したイメージをマーカー形状に使用します。PHOTO フィールドには、選手のイメージを指定する URL 参照が含まれています。この例では、series_label テンプレートマクロを使用してイメージが生成されています。

```
GRAPH FILE baseapp/fifi_players_2019_toprated
SUM HEIGHT__IN_
WAGE__€K_
ACCELERATION
BY PHOTO
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=PHOTO, BUCKET=color, $
TYPE=DATA, COLUMN=HEIGHT__IN, BUCKET=y-axis, $
TYPE=DATA, COLUMN=WAGE__€K, BUCKET=x-axis, $
TYPE=DATA, COLUMN=ACCELERATION, BUCKET=size, $
*GRAPH_JS_FINAL
"series": [{
                "series": "all",
                "marker": {
                    "shape": "url({{series_label}})",
                }
}],
"legend": {
    "visible": false
},
"bubbleMarker": {
    "maxSize": "12.0%"
},
"yaxis": {
    "min": 60.0,
    "max": 80.0
},
*END
ENDSTYLE
END
```

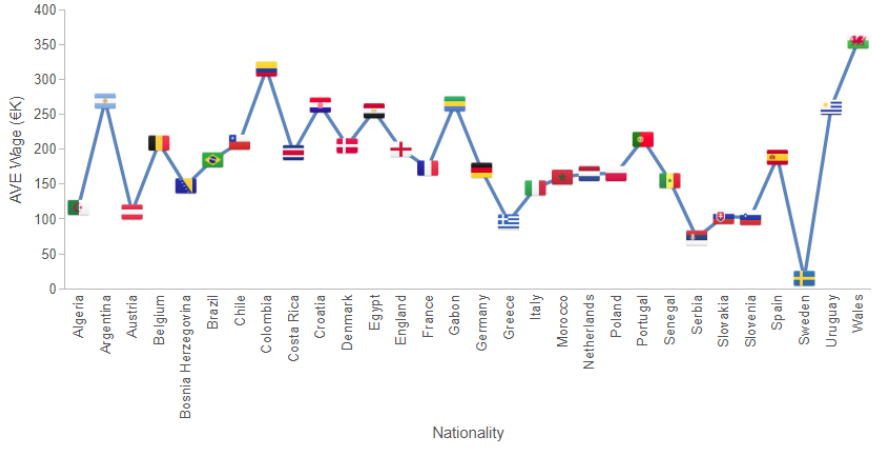
出力結果は次のとおりです。



次のリクエストは、折れ線グラフを生成し、FLAG フィールド値で指定したイメージをマーカー形状に使用します。各国旗は、グラフリクエスト内のグループ (NATIONALITY フィールド) に対応しています。FLAG フィールドは、flag という名前の仮想属性カテゴリに割り当てられます。国旗のイメージは、X 軸上の対応する国籍グループのマーカーとして表示されます。

```
GRAPH FILE baseapp/fifi_players_2019_toprated
SUM
AVE.WAGE__€K_
FLAG
BY NATIONALITY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,
$TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=>flag, $
*GRAPH_JS_FINAL
"series": [{
                "series": 0,
                "marker": {"visible": true, "size": 20,
                    "shape": "url({{flag}})",
                }
}],
"legend": {
    "visible": false
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



棒グラフ、折れ線グラフ、面グラフのシリーズライザ形状の定義

comboCharts プロパティとともに riserShape プロパティを使用して、複合グラフを定義します。複合グラフは、棒ライザ、面ライザ、線ライザの組み合わせで構成されるグラフです。棒グラフ、折れ線グラフ、面グラフでは、これらのプロパティを使用して、シリーズごとにライザ形状 (棒、線、面) を定義することができます。comboCharts プロパティは、各シリーズのライザレイアウト (stacked、absolute、percent、sideBySide) を制御します。

構文　棒グラフ、折れ線グラフ、面グラフのシリーズライザ形状の定義

```
"series":
[
    (
      "series": number,
      "riserShape": "string",
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

"riserShape": "string"

指定したシリーズのライザ形状を定義する文字列です。有効な値には、次のものがあります。

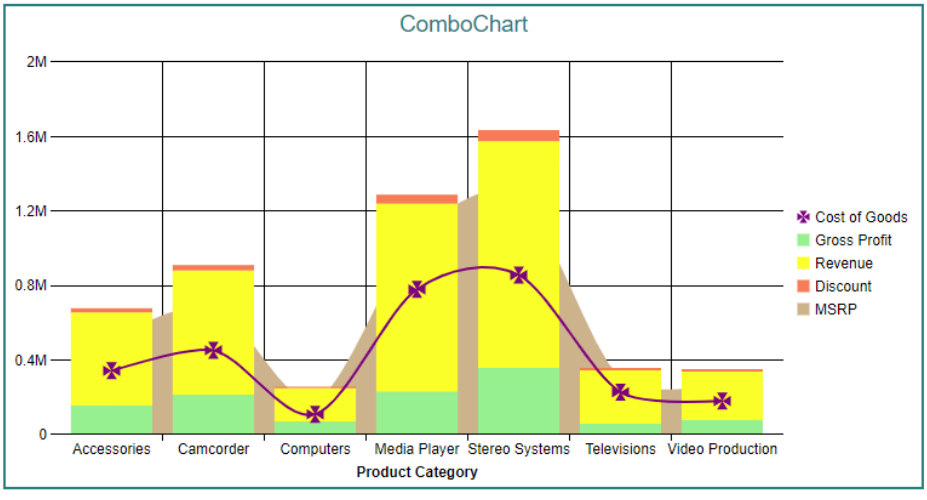
- "bar"
- "line"
- "area"

例　棒グラフ、折れ線グラフ、面グラフのライザ形状の定義

次のリクエストは、縦棒グラフを生成し、comboCharts プロパティで、棒シリーズレイアウトを stacked、線シリーズレイアウトを absolute、面シリーズレイアウトを undefined として定義します。各シリーズのプロパティで、それぞれのシリーズのライザ形状を定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": true},
"border": {"width": 2, "color": "teal"},
"title": {"visible": true, "text": "ComboChart", "font": "14pt Sans-Serif",
"color": "teal"},
"blaProperties": {
   "lineConnection": "curved",
   "comboCharts": {
      "barSeriesLayout": "stacked",
      "lineSeriesLayout": "absolute",
      "areaSeriesLayout": "undefined"}},
"series": [
   {"series": 0, "color": "purple", "riserShape": "line", "border":
{"width": 2},
      "marker": {"shape": "pirateCross", "size": 14, "visible": true}},
   {"series": 1, "color": "lightgreen", "riserShape": "bar"},
   {"series": 2, "color": "yellow", "riserShape": "bar"},
   {"series": 3, "color": "coral", "riserShape": "bar"},
   {"series": 4, "color": "tan", "riserShape": "area"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



散布図でのマーカーの接続

散布図でパターンを分析する場合、ポイントを線でつなぐことによってデータのパターンが明確になります。connectMarkers プロパティを使用して、散布図のポイントを接続することができます。この場合、BY フィールドは color カテゴリではなく detail 属性カテゴリに割り当てられます。

構文　　散布図でのマーカーの接続

```
"series":
 [{
  "series": 0,
  "connectMarkers": boolean,
  "border": {
            "width": number,
             "color": "string"
            }
  }
 ]
```

説明

`"connectMarkers": boolean`

マーカーを接続するかどうかを指定します。有効な値は、true (マーカーを接続する) または false (マーカーを接続しない) です。デフォルト値は false です。

マーカーを接続するには、BY フィールドは detail 属性カテゴリに割り当てる必要があります。

"border"
マーカーを接続する線のプロパティを定義します。

"width": number
線の幅をピクセル単位で指定します。

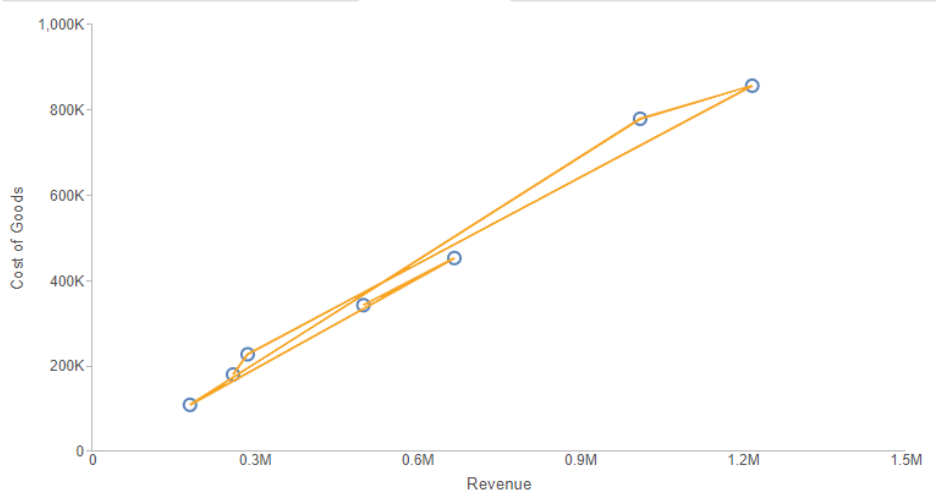
"color": "string"
線の色を定義する色指定の文字列です。

例　散布図でのマーカーの接続

次のリクエストは、散布図を生成し、マーカーを接続する線を幅 2 ピクセルのオレンジ色で描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=x-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=detail, $
*GRAPH_JS
"series": [{
    "series": 0,
    "connectMarkers": true,
    "border": {
        "width": 2,
        "color": "orange"
    }
}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



シリーズライザおよびマーカーでの塗りつぶしパターン名の使用

シリーズライザおよびマーカーによく使用される塗りつぶしパターンには名前が指定されています。

構文　　ライザおよびマーカーでの塗りつぶしパターン名の指定

パターンのプロパティで、シリーズの色指定文字列を置換します。

```
{"series": number,
      "color":{
               "type": "pattern",
               "color": "string",
               "backgroundColor": "string",
               "lineWidth": number,
               "antialias": true,
               "shape": "string",
               "size": number,
               "pad": number
               }
}
```

説明

`"color": "string"`
　塗りつぶしパターンの色です。undefined に設定した場合、パターンは表示されません。

`"backgroundColor": "string"`
　塗りつぶし文字間の背景色です。

`"lineWidth": number`
パターン内のストロークの幅をピクセル単位で指定します (存在する場合)。デフォルト値は 1 です。

`"antialias": boolean`
線にアンチエイリアス効果 (スムージング効果) を適用するかどうかを定義します。undefined に設定した場合、選択したパターンに基づいて適切な設定が自動的に選択されます。次のいずれかの値です。

- true - パターンにアンチエイリアス効果を適用します。これがデフォルト値です。
- false - パターンにアンチエイリアス効果を適用しません。
- undefined - 選択したパターンに基づいて、グラフエンジンが自動的に適切な設定を選択します。

`"shape": "string"`
次の塗りつぶしパターン名のいずれかを指定できます。

- **hline** - 連続した横線
- **vline** - 連続した縦線
- **hatch** - 連続したハッチパターン (#)
- **slash** - 連続したスラッシュ (/)
- **backslash** - 連続した円記号 (¥)
- **diagonalhatch** - 斜め方向の連続したハッチパターン (#)
- **dots** - 連続した点
- **diagonaldots** - 斜め方向の連続した点
- **solid** - 指定した色の単色塗りつぶし
- **empty** - 塗りつぶしなし

`"size": number`
パターンを繰り返す領域の幅と高さをピクセル単位で指定します。デフォルト値は 20 です。

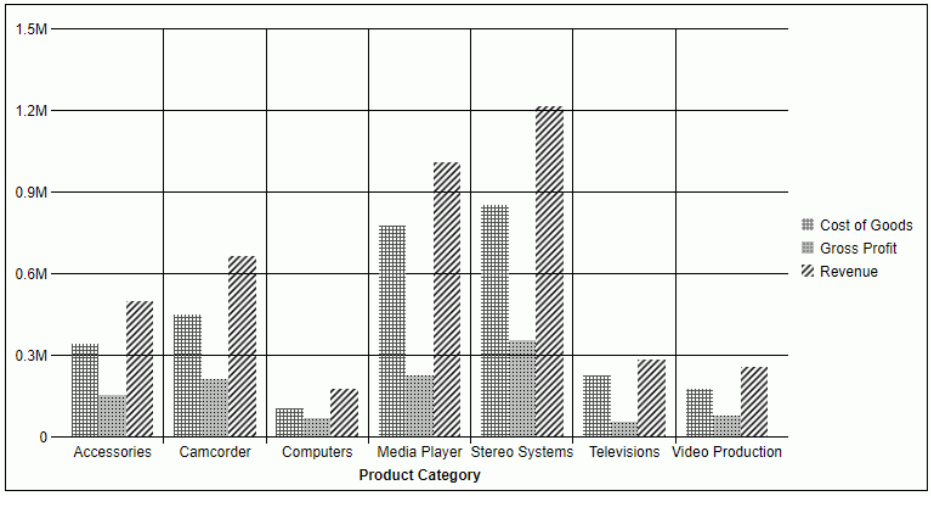
`"pad": number`
繰り返しパターンの周囲に追加する間隔をピクセル数で指定します。デフォルト値は 1 です。

例　棒グラフライザでの塗りつぶしパターン名の使用

次のリクエストは、シリーズ 0 に hatch 塗りつぶしパターンを指定し、シリーズ 1 には dots、シリーズ 2 には slash を指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=GROSS_PROFIT_US, bucket=y-axis,$
type=data, column=REVENUE_US, bucket=y-axis,$
type=data, column=PRODUCT_CATEGORY, bucket=x-axis,$
*GRAPH_JS
series:[
{"series": 0, "color": {"type": "pattern", "color": "#555",
              "shape": "hatch", "pad": 2, "lineWidth": 1}} ,
{"series": 1, "color": {"type": "pattern", "color": "black",
              "shape": "dots", "pad": 1.5, "lineWidth": 1.5}} ,
{"series": 2, "color": {"type": "pattern", "color": "black",
              "shape": "slash", "pad": 2.5, "lineWidth": 1.5}}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



棒グラフのライザ角形状の曲線化

デフォルト設定では、棒グラフのライザは直角の角形状で表示されます。cornerRadius シリーズプロパティを使用して、ライザ角形状を曲線にして斬新な外観にすることができます。

注意：グラフのライザ角形状を曲線にした場合でも、凡例マーカーの角形状は曲線になりません。

構文　棒グラフのライザ角形状の曲線化

```
"series": number, "border": {"cornerRadius": {"x": value, "y": value}}
```

説明

`"series": number`
角形状を曲線にするシリーズ番号を指定するか、"all" を指定して cornerRadius プロパティをすべてのライザに適用します。"all" シリーズに対して指定した角形状は、cornerRadius プロパティが個別に指定されているシリーズを除き、すべてのシリーズに適用されます。

`"border"`
ライザ周囲の境界のプロパティを定義します。

`"cornerRadius": {"x": value, "y": value}`
ライザ角形状のプロパティを定義します。

cornerRadius プロパティには、1 つまたは 2 つの値を指定することができます。単一値の 0 (ゼロ) を指定すると、角形状は直角になります。これがデフォルト角形状です。曲線の角形状は、円形または楕円形にすることができます。x および y 値は、円形の半径、または楕円形の長半径と短半径を表します。ここで x および y 値はグラフの向きに対応します。x は序数軸を表し、y は数値軸を表します。

x および y の有効値は次のとおりです。

- 数値 - 角形状の半径をピクセル単位で指定します。
- パーセント文字列 (例、"20%") - ライザの序数軸または数値軸のパーセントを表します。
- CSS 長さ文字列 (例、"5em"、"10px") - 角形状の半径をピクセル単位で指定します。CSS 長さ文字列についての詳細は、https://developer.mozilla.org/en-US/docs/Web/CSS/length を参照してください。

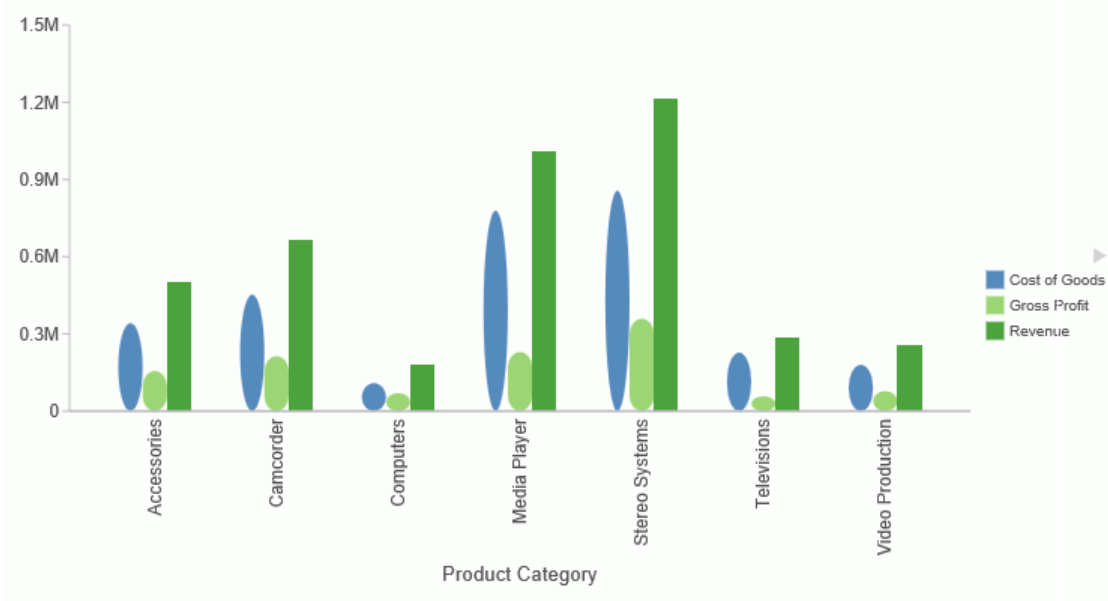
- "x" および "y" プロパティで定義されるオブジェクト (ここで "x" および "y" には上記のいずれかを指定可能) - ライザの序数軸 (x) および数値軸 (y) の角形状の半径を指定します。

例　棒グラフのライザ角形状の制御

次のリクエストでは、シリーズ 0 (COG_US) に楕円形の角形状、シリーズ 2 (REVENUE_US) に直角の角形状を適用し、シリーズ 1 (GROSS_PROFIT_US) には "all" シリーズに対して指定された円形の角形状を適用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET STYLE *

INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$

TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=y-axis, $

*GRAPH_JS
series: [{series: 'all', border: {cornerRadius: 10}},
{series: 0, border: {cornerRadius: {x: "10%", y: "30%"}}},
{series: 2, border: {cornerRadius: 0}}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ライザのツールヒントの定義または非表示

tooltip プロパティは、1 つまたは複数のライザのツールヒントを定義します。ツールヒントは、ライザ上にマウスポインタを置いたときに表示されます。ツールヒントは、すべてのライザとマーカー、単一のライザとマーカー (シリーズとグループで識別)、または特定のシリーズのライザとマーカーに割り当てることができます。

シリーズのツールヒントに null プロパティを使用して、特定のライザのツールヒントを非表示にすることができます。

グラフ上の複数のオブジェクト (例、ライザ) が互いに近接し、ツールヒントにメニュー (例、ドリルダウンメニュー) が表示される場合、オブジェクトのいずれかをクリックした後、近傍の別のオブジェクトにマウスポインタを置いた際に、そのオブジェクトに関連するツールヒントが正しく表示されない場合があります。

また、マクロを使用してグラフテンプレートエンジンでツールヒントをカスタマイズすることもできます。詳細は、61 ページの「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

autoNumberFormats を使用して、ツールヒントに数値フォーマットを適用することもできます。数値フォーマットについての詳細は、86 ページの 「 数値のフォーマット設定 」 を参照してください。

注意

- グラフ属性構文を使用する場合、サーバ生成のツールヒントを上書きするためには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックを使用する必要があります。グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。
- グラフ属性構文を使用するリクエストでは、新しいスタイルのツールヒントが生成されますが、グラフ属性構文を使用しないグラフリクエストの場合、古いスタイルのツールヒントが表示されます。

構文　ライザのツールヒントの定義

```
"series": [
{
   "series": number,
   "group": number, // Optional
   "tooltip": "string"}
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。"all" を指定すると、すべてのシリーズにツールヒントが割り当てられます。"reset" を指定すると、デフォルト値が削除されます。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"group": number`

0 (ゼロ) を基準とするグループ番号です (オプション)。グループ番号を指定しない場合、指定したプロパティがシリーズ内のすべてのグループに適用されます。

`"tooltip": "string"`

マウスポインタをライザ上に置いたときに表示する文字列、または文字列を返す関数を指定します。ツールヒントオブジェクトに null 値を使用して、特定シリーズのツールヒントを非表示にすることができます。

htmlTooltip を true に設定した場合、特別な文字列である "auto" を使用することで、ツールヒントにシリーズラベル、グループラベル、値を自動的に表示することができます。表示されるテキストは、グラフタイプごとに異なります。

コールバック関数は、値、シリーズ ID、グループ ID の 3 つの引数を指定して呼び出します。以下はその例です。

```
tooltip: function(value, series, group) {
 return this.title.text + ", value: " + value + ", s: " + series + ",
g: " + group;
}
```

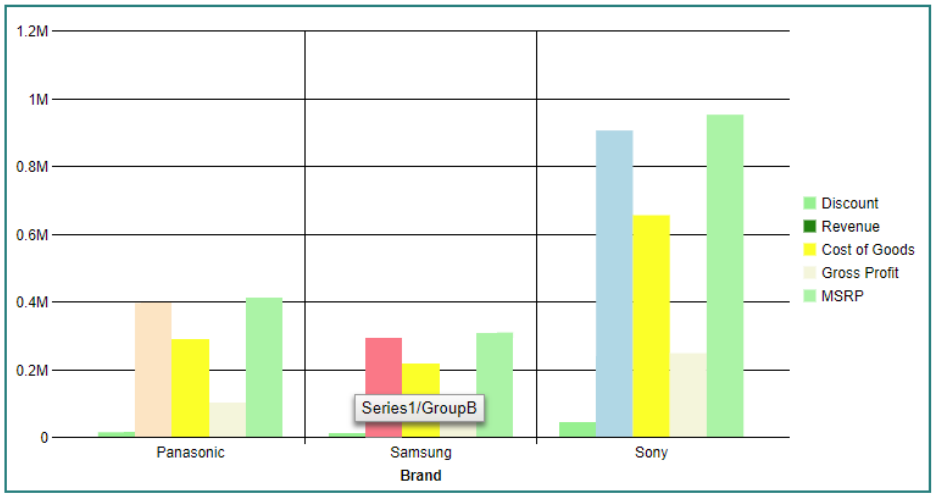
関数の定義でテキストを引用符で囲む場合は、一重引用符 (') と二重引用符 (") の両方がサポートされます。

例　シリーズのツールヒントの定義

次のリクエストは、シリーズ 1 のグループ 1 にツールヒントを割り当てます。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US REVENUE_US COGS_US GROSS_PROFIT_US MSRP_US
BY BRAND
WHERE BRAND EQ 'Sony' OR 'Samsung' OR 'Panasonic'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "group": 0, "color": "bisque"},
   {"series": 1, "group": 1, "color": "red", "tooltip": "Series1/GroupB"},
   {"series": 1, "group": 2, "color": "lightblue"},
   {"series": 2, "color": "yellow"},
   {"series": 3, "color": "beige"}]
*END
ENDSTYLE
END
```

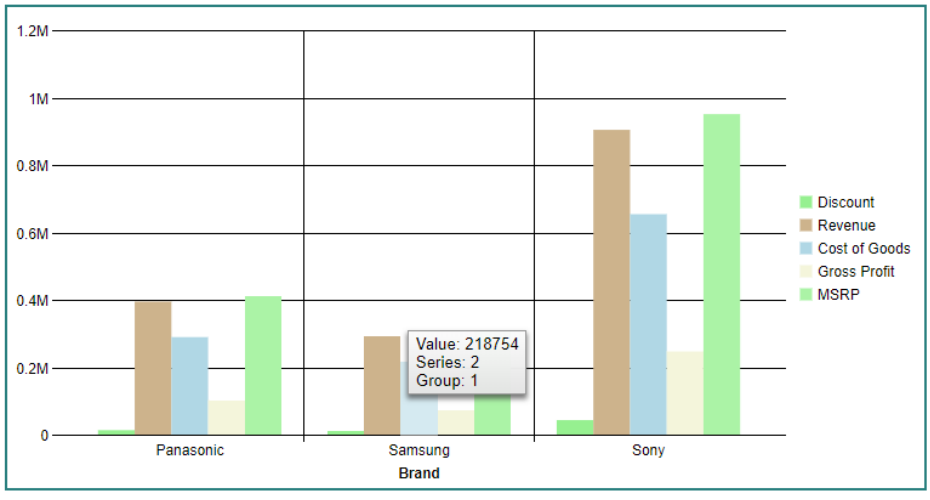
ツールヒントは、指定したライザ上にマウスポインタを置いたときに表示されます。



次のリクエストは、コールバック関数を使用してツールヒントを生成します。コールバック関数を使用するためにデフォルトツールヒントをすべて削除するには、シリーズの "reset" を使用するか、各シリーズのツールヒントを undefined に設定します。関数には HTML タグを使用することもできます。この例では、
 タグを使用してツールヒント内に改行を生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US REVENUE_US COGS_US GROSS_PROFIT_US MSRP_US
BY BRAND
WHERE BRAND EQ 'Sony' OR 'Samsung' OR 'Panasonic'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
*GRAPH_JS_FINAL
"border": {"width": 2, "color": "teal"},
"series": [
   {"series": "reset", "tooltip": function(v, s, g) {return "Value: " + v
+"<br>Series: " + s +"<br>Group: " + g;}},
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "tan"},
   {"series": 2, "color": "lightblue"},
   {"series": 3, "color": "beige"}]
*END
ENDSTYLE
END
```

出力結果では、コールバック関数から返されたカスタムツールヒントが表示されています。



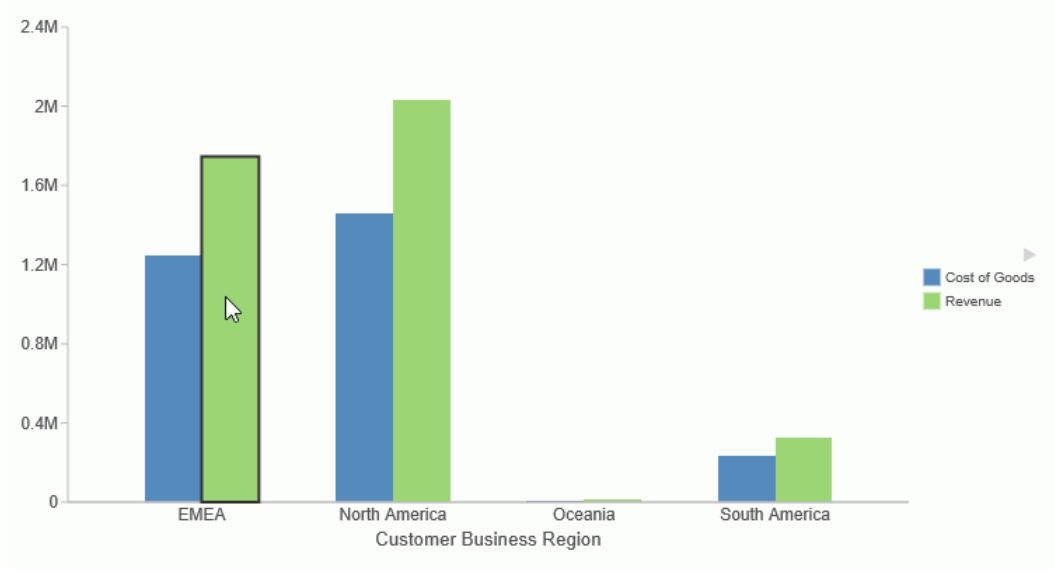
注意：1 つまたはすべてのシリーズでツールヒントが定義されている場合、htmlToolTip プロパティを使用して、任意のグラフツールヒントに HTML ベース (div) スタイルのツールヒントを定義することができます。

例　シリーズのツールヒントの非表示

次のリクエストは、シリーズ 1 のツールヒントを非表示にします。

```
GRAPH FILE wf_retail_lite
SUM COGS_US REVENUE_US
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"series": [ {"series": 1, "tooltip": null} ],
*END
ENDSTYLE
END
```

出力結果は次のとおりです。シリーズ 1 のライザ上にマウスポインタが置かれていますが、ツールヒントは表示されません。



構文　　ツールヒントのサブメニューの定義

ツールヒントを、エントリと子エントリを含むサブメニューの配列として定義することができます。ネストされたサブメニュー (各子エントリにもそのサブメニューが含まれる) がサポートされます。htmlToolTip プロパティを使用して、これらのメニューのスタイルを設定することができます。

```
"series":[
 {"series": number,
   "tooltip": [
      {
    "entry": "Submenu1",
    "children": [
                "Child1",
                "Child2"
                ]
      },
   {
    "entry": "Submenu2",
    "children": [
                 {
                 "entry": "Nested Submenu",
                 "children": [
                             "NestedChildl",
                             "NestedChild2"
                             ]
                }
                  ]
    }

       ]
  }]
```

説明

`"series": number`

ツールヒントのシリーズ ID を指定します。

`"entry": "Submenu1", "entry": "Submenu2" ...`

ツールヒントのサブメニューのテキストを定義します。ツールヒント構成要素の組み合わせを使用することができます (例、リテラル値とテンプレートマクロ)。

`"children": [ "Child1", "Child2" ... ]`

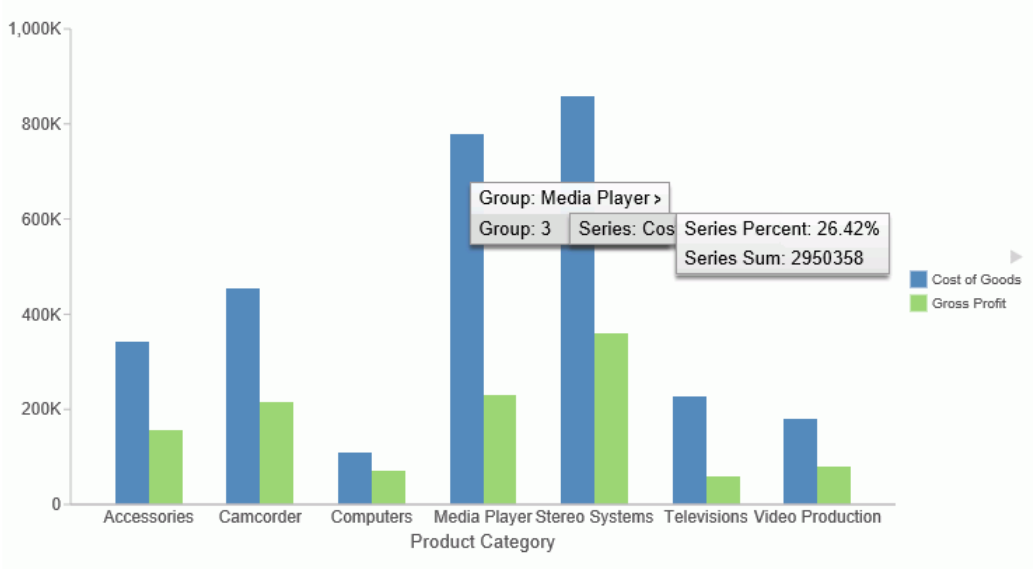
サブメニューの子エントリのテキストを定義します。子にもサブメニューを追加することができます。ツールヒント構成要素の組み合わせを使用することができます (例、リテラル値とテンプレートマクロ)。

例　ツールヒントのサブメニューの定義

次のリクエストは、シリーズ 0 に 2 つのサブメニューを追加します。2 つ目のサブメニューはネストされたサブメニューです。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"series":[
 {"series": "0",
   "tooltip": [
      {
          "entry": "Group: {{group_label}}",
          "children": [
                "Series Label: {{series_label}}",
                "Series ID: {{series_id}}"
             ]
      },
   {
    "entry": "Group: {{group_id}}",
    "children": [
                 {
                 "entry": "Series: {{series_label}}",
                 "children": [
                          "Series Percent: {{series_percent}}",
                    "Series Sum: {{series_sum}}"
                             ]
                }
                  ]
    }

       ]
  }
 ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



バブルグラフおよび散布図でのシリーズ傾向線の描画

trendline プロパティを使用して、バブルグラフおよび散布図で特定のシリーズの傾向線および数式ラベルを描画することができます。

注意：グローバルな trendline プロパティを使用して、すべてのシリーズの傾向線を描画することもできます。詳細は、867 ページの 「バブルグラフおよび散布図での傾向線の描画 (trendline)」 を参照してください。

構文　バブルグラフおよび散布図でのシリーズ傾向線の描画

```
"series": [
{
  "series": number,
      "trendline": {
         "enabled": boolean,
         "mode": "string",
         "order": number,
         "lineStyle": {
             "width": number,
             "color": "string",
             "dash": "string"
                   },
         "equationLabel": {
            "visible": boolean,
            "font": "string",
            "color": "string",
            "mode": "string"
         }
      }
}
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"enabled": boolean`

傾向線を有効または無効にします。有効な値には、次のものがあります。

- true - 傾向線を有効にします。
- false - 傾向線を無効にします。これがデフォルト値です。

`"mode": "string"`

傾向線モードを定義する文字列です。傾向線モードは、傾向線の生成に使用する数式のタイプです。有効な値には、次のものがあります。

- "exponential"
- "geometric"
- "hyperbolic"
- "linear" - 傾向線を表示する場合、これがデフォルト値です。
- "logarithmic"

- "logQuadratic"
- "modExponential"
- "modHyperbolic"
- "polynomial"
- "quadratic"
- "rational"
- undefined - 傾向線を表示しない場合、これがデフォルト値です。傾向線を表示するが、"mode" が undefined に設定されている場合は、"linear" モードが使用されます。

`"order": number`
多項式モードに適用されます。多項式の次数を指定します。デフォルト値は 3 です。

`"lineStyle":`
傾向線のプロパティを定義します。

`"width": number`
傾向線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`
傾向線の色を定義する色指定の文字列です。デフォルト値は undefined です。この場合、傾向線色にシリーズ色を使用します。

`"dash": "string"`
傾向線の破線スタイルを定義する文字列です。破線長さ、破線間隔の順にピクセル数で指定します。たとえば、「"1 1"」と指定すると点線が生成されます。デフォルト値は「""」です (実線)。

`"equationLabel"`
傾向線の数式ラベルのプロパティを定義します。

`"visible": boolean`
数式ラベルの表示を制御します。有効な値には、次のものがあります。

- true - 数式ラベルを表示します。
- false - 数式ラベルを表示しません。これがデフォルト値です。

`"font": "string"`

数式ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "8pt Sans-Serif" です。

```
"color": "string"
```

色名または数値指定の文字列で定義する色です。 デフォルト値は "black" です。

```
"mode": "string"
```

数式ラベルのモードを定義する文字列です。有効な値には、次のものがあります。

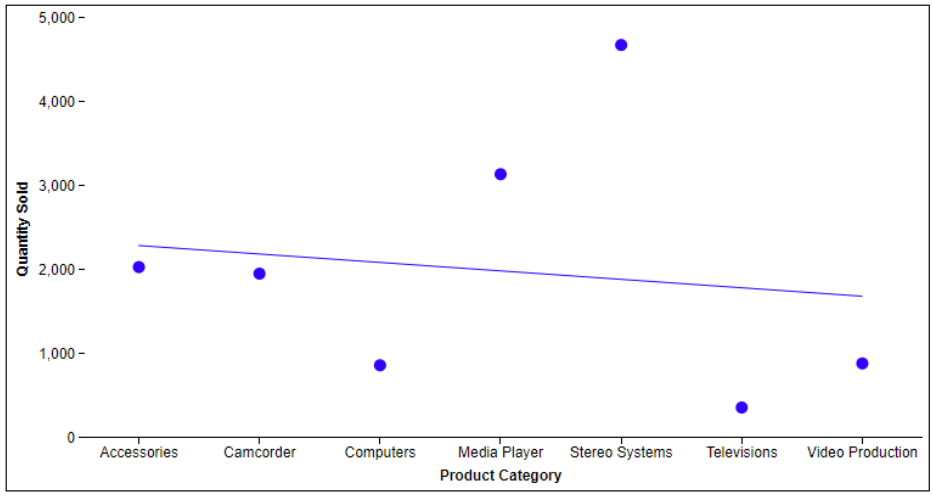
- "equation" - 生成された傾向線のタイプに最も適した形式で数式を表示します。直線の傾向線の場合、傾きと切片で表現する形式で数式が表示されます。これがデフォルト値です。
- "r" - 相関係数を表示します。

例　シリーズ傾向線の描画

次のリクエストは、散布図を生成し、シリーズ 0 の傾向線を直線で描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"series": [
   {"series": 0, "color": "blue", "marker": {"size": 10},
      "trendline": {"enabled": true, "mode": "linear"
      }}]
*END
ENDSTYLE
END
```

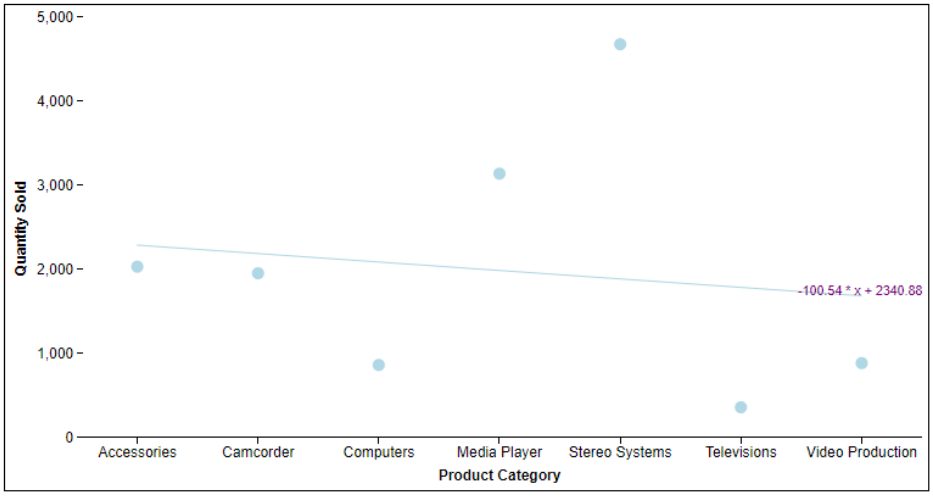
出力結果は次のとおりです。



次のリクエストは、紫色の数式ラベルを追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"series": [
   {"series": 0, "color": "lightblue", "marker": {"size": 10},
      "trendline": {
      "enabled": true, "mode": "linear", "equationLabel": {
         "visible": true, "color": "purple"}}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

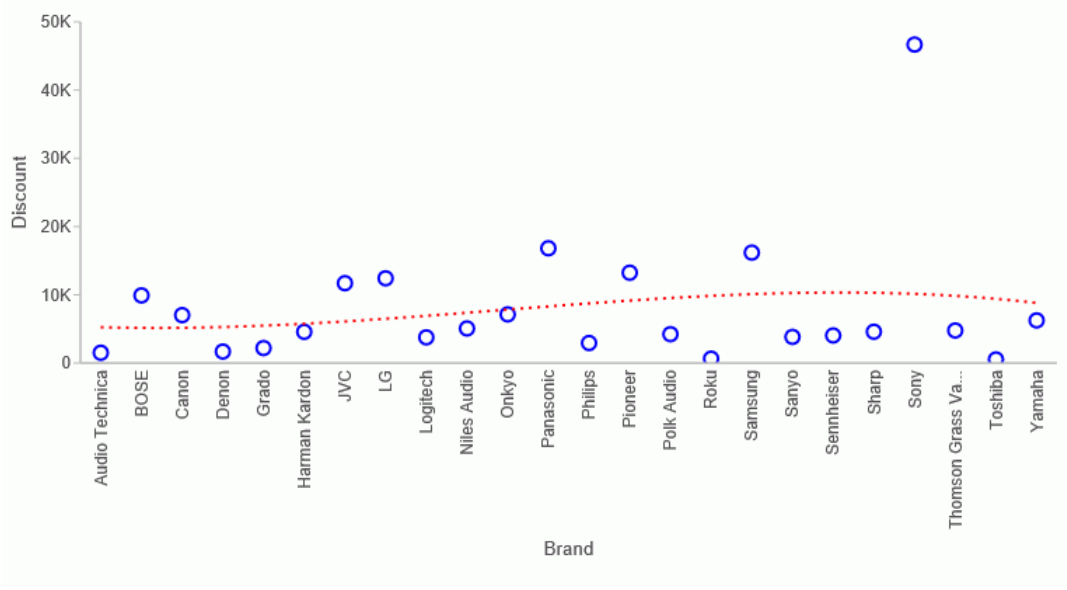


例　シリーズ傾向線のスタイル設定

次のリクエストは、幅 2 ピクセルの赤色の破線で傾向線のスタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US
BY BRAND
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=BRAND, BUCKET=x-axis,$
TYPE=DATA, COLUMN=DISCOUNT_US , BUCKET=y-axis,$
*GRAPH_JS
"series": [
{"series": 0, "color": "blue", "trendline": {"enabled": true,
   "mode": "polynomial", "order": 3,
   "lineStyle": {"width": 2, "color": "red", "dash": "2 4"},
}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



棒グラフ、折れ線グラフ、面グラフでのシリーズ傾向線の描画

trendline プロパティを使用して、棒グラフ、折れ線グラフ、面グラフで特定のシリーズの傾向線および数式ラベルを描画することができます。

構文　　棒グラフ、折れ線グラフ、面グラフでのシリーズ傾向線の描画

```
"series": [
{
  "series": number,
      "trendline": {
         "enabled": boolean,
         "mode": "string",
         "order": number,
         "lineStyle": {
             "width": number,
             "color": "string",
             "dash": "string"
                   },
         "equationLabel": {
            "visible": boolean,
            "font": "string",
            "color": "string"
         }
      }
}
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"enabled": boolean`

傾向線を有効または無効にします。有効な値には、次のものがあります。

- true - 傾向線を有効にします。
- false - 傾向線を無効にします。これがデフォルト値です。

`"mode": "string"`

傾向線モードを定義する文字列です。傾向線モードは、傾向線の生成に使用する数式のタイプです。有効な値には、次のものがあります。

- "min"
- "max"
- "mean"
- "stddev"
- "ucl" - 管理図の上方管理限界線です。

- "lcl" - 管理図の下方管理限界線です。

"lineStyle":
傾向線のプロパティを定義します。

"width": number
傾向線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"
傾向線の色を定義する色指定の文字列です。デフォルト値は undefined です。この場合、傾向線色にシリーズ色を使用します。

"dash": "string"
傾向線の破線スタイルを定義する文字列です。破線長さ、破線間隔の順にピクセル数で指定します。たとえば、「"1 1"」と指定すると点線が生成されます。デフォルト値は「""」です (実線)。

"equationLabel"
傾向線の数式ラベルのプロパティを定義します。

"visible": boolean
数式ラベルの表示を制御します。有効な値には、次のものがあります。

- true - 数式ラベルを表示します。
- false - 数式ラベルを表示しません。これがデフォルト値です。

"font": "string"

数式ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "8pt Sans-Serif" です。

"color": "string"

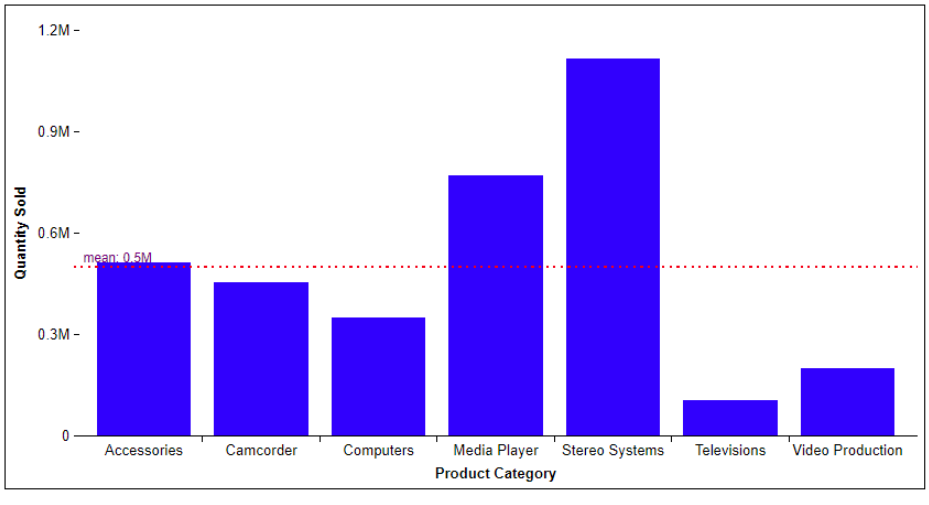
色名または数値指定の文字列で定義する色です。 デフォルト値は "black" です。

例　棒グラフでのシリーズ傾向線の描画およびスタイル設定

次のリクエストは棒グラフを生成し、シリーズ 0 の平均値に赤色の破線の傾向線および紫色の数式ラベルを追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"series": [
{"series": 0, "color": "blue", "marker": {"size": 10},
"trendline": {"enabled": true, "mode": "mean",
"lineStyle": {"width": 2, "color": "red", "dash": "2 4"},
"equationLabel": {"visible": true, "color": "purple"}
}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



特定シリーズの表示の制御

visible プロパティは、特定のシリーズの表示を制御します。

構文　特定シリーズの表示の制御

```
"series":
[
    (
      "series": number,
      "visible": boolean,
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"visible": boolean`

指定したシリーズの表示を制御します。有効な値には、次のものがあります。

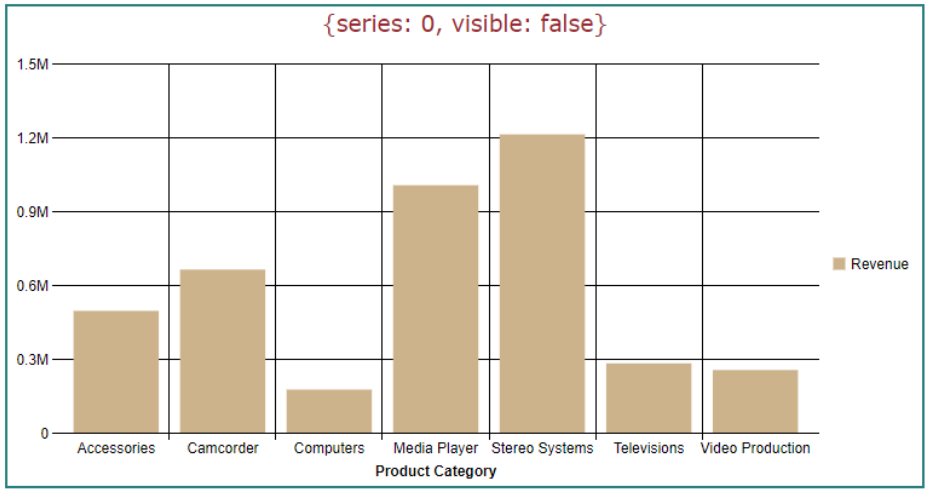
- true - シリーズを表示します。これがデフォルト値です。
- false - シリーズを表示しません。

例　特定シリーズの表示の制御

次のリクエストは、縦棒グラフを生成し、シリーズ 0 を非表示にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": true},
"title": {"visible": true, "text": "{series: 0, visible: false}",
   "font": "14pt Verdana", "color": "brown"},
"border": {"width": 2, "color": "teal"},
"series": [
   {"series": 0, "color": "rgb(204,255,255)", "label": "SERIES ZERO",
"visible": false},
   {"series": 1, "color": "tan", "label": "SERIES ONE"}]
*END
ENDSTYLE
END
```

出力結果では、シリーズ 0 のライザは表示されず、凡例にもシリーズ 0 のエントリは表示されていません。



軸へのシリーズの割り当て

yAxisAssignment プロパティは、軸にシリーズを割り当てます。yAxisAssignment を 2 に設定すると、Y2 軸ラベルがグラフに追加されます。y2axis オブジェクトを制御するプロパティについての詳細は、315 ページの 「 軸のプロパティ 」 を参照してください。

注意 : 属性の割り当てにサブスクリプト値を使用して、シリーズを軸に割り当てることもできます。たとえば、n3 列を Y2 軸に割り当てます。

```
type=data, column=n3, bucket=y-axis(2)
```

グラフ属性カテゴリへのフィールドの割り当てについての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

構文　軸へのシリーズの割り当て

```
"series":
[
    {
      "series": number,
      "yaxisAssignment": number,
   }
]
```

説明

`"series": number`

0 (ゼロ) を基準とするシリーズ番号です。選択したシリーズがグラフに存在しない場合、そのプロパティは無視されます。

`"yaxisAssignment": number`

指定したシリーズを Y 軸または Y2 軸に割り当てます。有効な値には、次のものがあります。

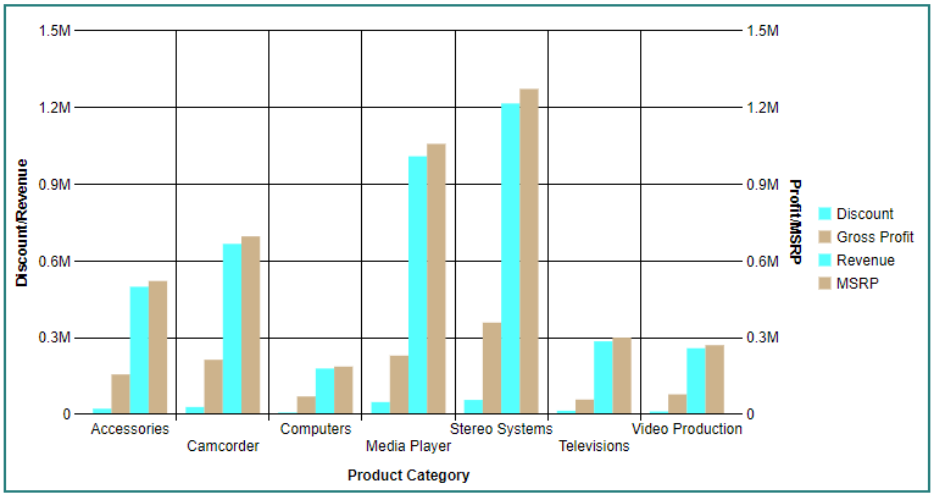
- 1 - シリーズを Y 軸に割り当てます。これがデフォルト値です。
- 2 - シリーズを Y2 軸に割り当てます。

例 軸へのシリーズの割り当て

次のリクエストは、Y 軸および Y2 軸で構成された縦棒グラフを生成し、それぞれの軸にシリーズを割り当てます。

```
GRAPH FILE WF_RETAIL_LITE
SUM DISCOUNT_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"title": {"text": "yAxisAssignment", "font": "14pt Sans-Serif", "color":
"teal"},
"yaxis": {"title": {"visible": true, "text": "Discount/Revenue"}},
"y2axis": {"title": {"visible": true, "text": "Profit/MSRP"}},
"series": [
   {"series": 0, "color": "cyan", "yAxisAssignment": 1},
   {"series": 1, "color": "tan", "yAxisAssignment": 2},
   {"series": 2, "color": "cyan", "yAxisAssignment": 1},
   {"series": 3, "color": "tan", "yAxisAssignment": 2}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



10

グラフ固有のプロパティ

ここでは、各グラフタイプに固有のプロパティについて説明します。任意のプロパティに対する「undefined」という値は、そのプロパティが無視されることを意味します。

トピックス

- 棒グラフ、折れ線グラフ、面グラフのプロパティ (blaProperties)
- ボックスプロットおよび統計ボックスプロットのプロパティ (boxPlotProperties)
- ブレットグラフのプロパティ (bulletProperties)
- データグリッドのプロパティ (dataGridProperties)
- じょうごグラフのプロパティ (funnelProperties)
- メータグラフのプロパティ (gaugeProperties)
- ヒストグラムのプロパティ (histogramProperties)
- マトリックスグラフのプロパティ (matrixProperties)
- パラボックスのプロパティ (paraboxProperties)
- 円グラフのプロパティ (pieProperties)
- 極グラフのプロパティ (polarProperties)
- タグクラウドのプロパティ (tagloudProperties)
- 3D グラフのプロパティ (threedProperties)
- ツリーマップのプロパティ (treemapProperties)
- 滝型グラフのプロパティ (waterfallProperties)

棒グラフ、折れ線グラフ、面グラフのプロパティ (blaProperties)

次のプロパティは、棒グラフ、折れ線グラフ、面グラフの基本属性のデフォルト値を示しています。「undefined」という値は、そのプロパティが無視されることを意味します。

```
"blaProperties": {
"seriesLayout": "sideBySide",
"orientation": "horizontal",
"lineConnection": "linear",
"lineFillEffect": "undefined",
"barGroupGapWidth": 0.2,
"barEffect": undefined,
"stackTotalLabel": {
    "visible": false,
    "font": "10pt Sans-Serif",
    "color": "black",
    "numberFormat": "auto"
},
"missingDataMode": "none",
"missingDataLineStyle": undefined
"splitY": true,
"comboCharts": {
    "barSeriesLayout": undefined,
    "lineSeriesLayout": undefined,
    "areaSeriesLayout": undefined
    }
}
```

また、軸のプロパティの groupFit を使用してライザ幅を制御したり、sort プロパティを使用して積み上げ棒グラフのライザを数値軸値の昇順または降順でソートしたりできます。詳細は、315 ページの 「 軸のプロパティ 」 を参照してください。

棒グラフ、折れ線グラフ、面グラフのシリーズレイアウトの制御

seriesLayout プロパティは、棒グラフ、折れ線グラフ、面グラフのライザの配置方法を制御します。

注意：スタイルセクションの JSON ブロックに seriesLayout プロパティを含める代わりに、WebFOCUS スタイルシートの CHART-SERIES-LAYOUT パラメータを使用して各レイアウトの値を指定することもできます。詳細は、193 ページの 「 棒グラフ、折れ線グラフ、面グラフのグラフレイアウト指定 」 を参照してください。

構文　棒グラフ、折れ線グラフ、面グラフのシリーズレイアウトの制御

```
"blaProperties": {
   "seriesLayout": "string"
              }
```

説明

```
"seriesLayout": "string"
```

ライザの配置方法を定義します。有効な値には、次のものがあります。

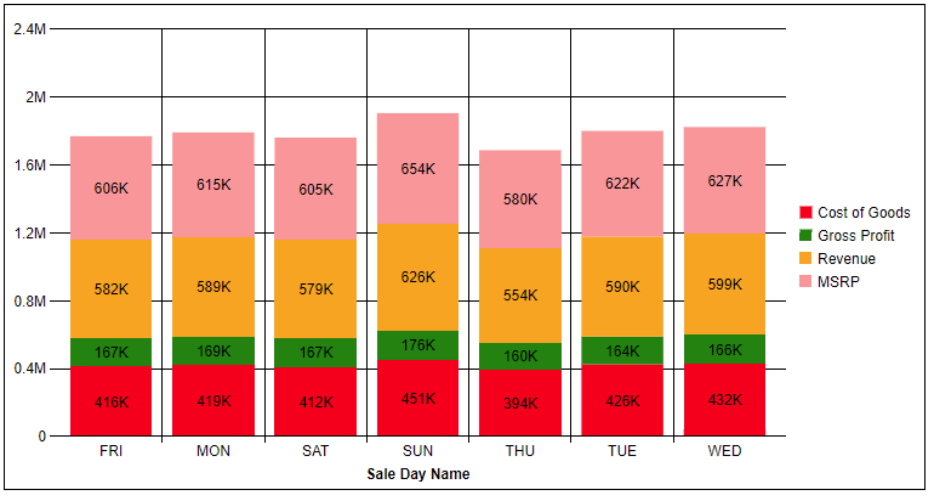
- "stacked" - ライザを別のライザの上に積み重ねます。各ライザの長さは、データ値を表します。
- "absolute" - 縦グラフの場合、各ライザを同一の X 軸値の位置に描画します。横グラフの場合、各ライザを同一の Y 軸値の位置に描画します。各ライザに異なる幅を設定すると、グラフ上でライザを区別することができます。
- "sideBySide" (棒グラフのみ) - 縦棒グラフの場合、ライザを別のライザの横に並べて配置します。横棒グラフの場合、ライザを別のライザの上に並べて配置します。デフォルト値は "sideBySide" です。
- "percent" - ライザを別のライザの上に積み重ねます。各ライザの長さは、そのライザの合計に対するパーセント値を表し、積み上げの合計が 100% になります。

例 seriesLayout プロパティの設定

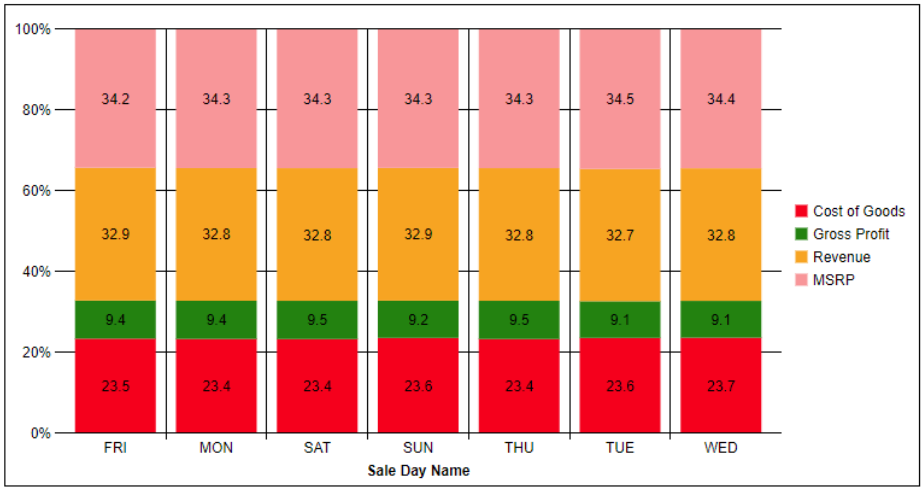
次のリクエストは、seriesLayout プロパティを "stacked" に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US MSRP_US
BY TIME_DAYNAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {"seriesLayout": "stacked"},
"series": [{"series": "all"}],
"dataLabels": {"visible": true, "position": "center"}
*END
ENDSTYLE
END
```

出力結果では、ライザが別のライザの上に積み重ねられ、各ライザにそのデータ値が表示されています。



seriesLayout プロパティを "percent" に変更すると、積み上げグラフが生成され、各ライザの長さが合計に対するパーセント値を表し、各グループの合計が 100% になります。



棒グラフ、折れ線グラフ、面グラフのグラフ方向の指定

orientation プロパティは、棒グラフ、折れ線グラフ、面グラフの方向を指定します。

注意：スタイルセクションの JSON ブロックに orientation プロパティを含める代わりに、WebFOCUS スタイルシートの CHART-ORIENTATION パラメータを使用して、各方向の値を指定することもできます。詳細は、191 ページの 「 グラフ方向の指定 」 を参照してください。

構文　グラフ方向の指定

```
"blaProperties": {
   "orientation": "string"
   }
```

説明

`"orientation": "string"`

有効な値には、次のものがあります。

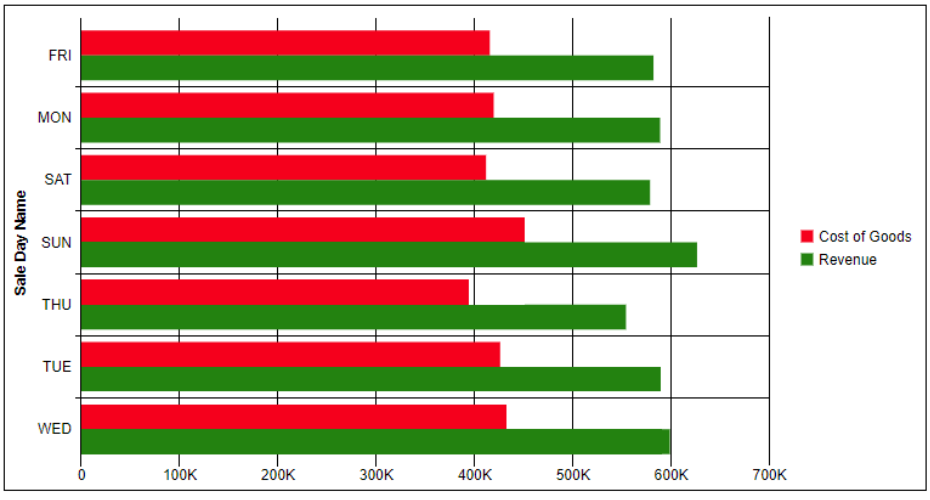
- "horizontal" - 横方向のグラフを描画します (左側に X 軸、下側に Y 軸)。これがデフォルト値です。
- "vertical" - 縦方向のグラフを描画します (下側に X 軸、左側に Y 軸)。

例　棒グラフのグラフ方向の制御

デフォルト設定では、グラフエンジンは縦棒グラフを作成します。次のリクエストは、グラフ方向を横方向に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY TIME_DAYNAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {"orientation": "horizontal"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



棒グラフ、折れ線グラフ、面グラフのデータポイント接続方法の制御

lineConnection プロパティは、折れ線グラフ、面グラフ、複合グラフでのデータポイントの接続方法を制御します。この複合グラフは、comboCharts プロパティ (537 ページの 「 複合グラフのプロパティの指定 」) とシリーズ固有の riserShape プロパティを使用して複数の面ライザおよび線ライザを組み合わせたグラフです。

構文　棒グラフ、折れ線グラフ、面グラフの lineConnection プロパティの設定

```
"blaProperties": {
   "lineConnection": "string"
   }
```

説明

`"lineConnection": "string"`

線の接続方法を定義します。有効な値には、次のものがあります。

- "linear" - 直線セグメントを使用してマーカーを接続します。これがデフォルト値です。
- "curved" - 曲線セグメントを使用してマーカーを接続します。
- "stepBefore" - 横方向と縦方向のみの直線セグメントを使用してマーカーを接続します。前のマーカーと次のマーカーを接続する線セグメントは、次のマーカーの Y 座標 (縦グラフ) または X 座標 (横グラフ) がセグメントの終点になるように描画されます。

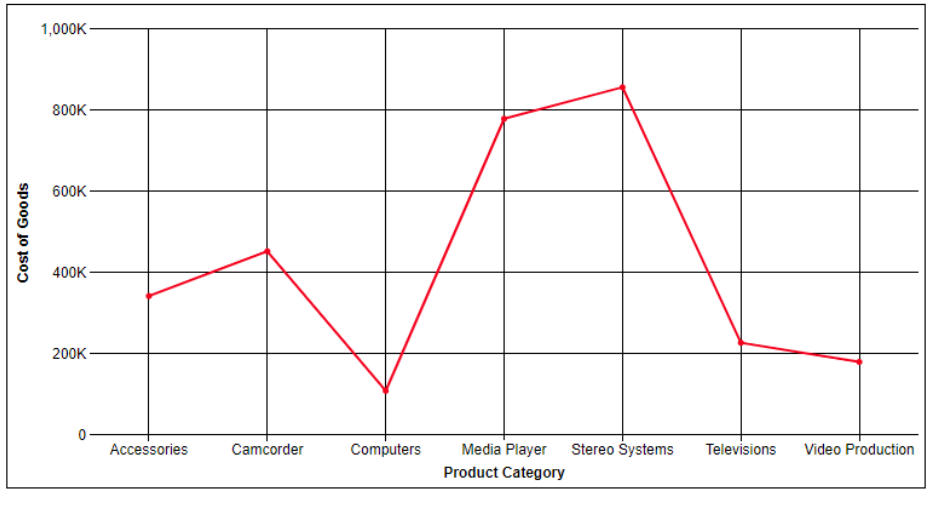
- "stepBetween" - 横方向と縦方向のみの直線セグメントを使用してマーカーを接続します。前のマーカーと次のマーカーを接続する線セグメントは、次のマーカーの Y 座標 (縦グラフ) または X 座標 (横グラフ) がセグメントの中点になるように描画されます。
- "stepAfter" - 横方向と縦方向のみの直線セグメントを使用してマーカーを接続します。前のマーカーと次のマーカーを接続する線セグメントは、次のマーカーの Y 座標 (縦グラフ) または X 座標 (横グラフ) がセグメントの始点になるように描画されます。

例　折れ線グラフの lineConnection プロパティの設定

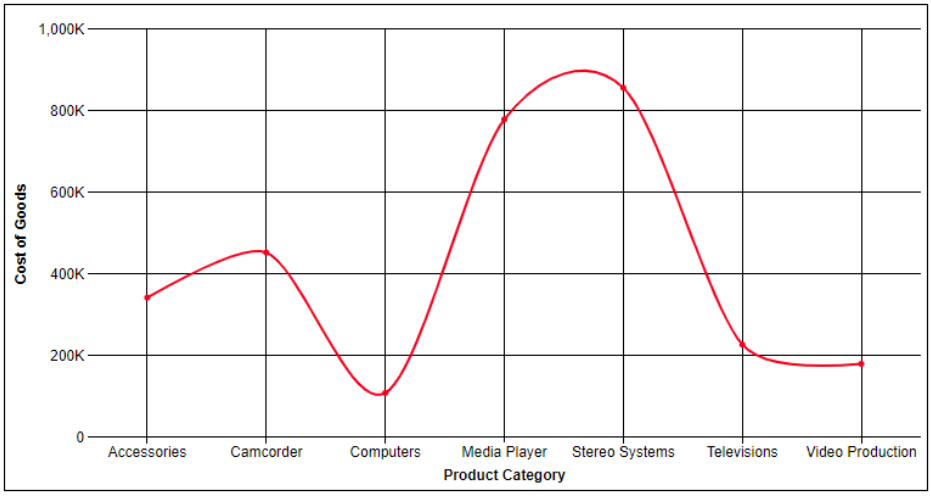
次のリクエストは、縦棒グラフを作成し、lineConnection プロパティをデフォルト値の "linear" に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {
   "lineConnection": "linear"
}
*END
ENDSTYLE
END
```

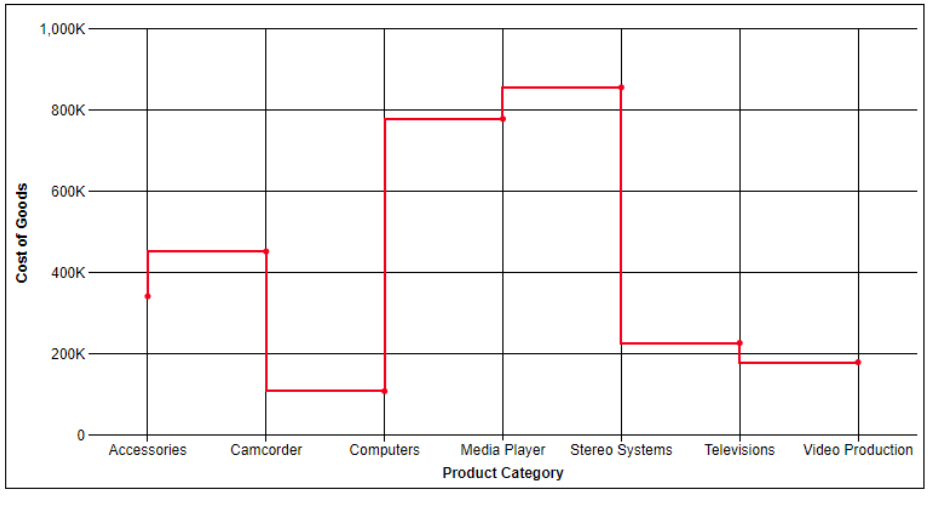
出力結果は次のとおりです。



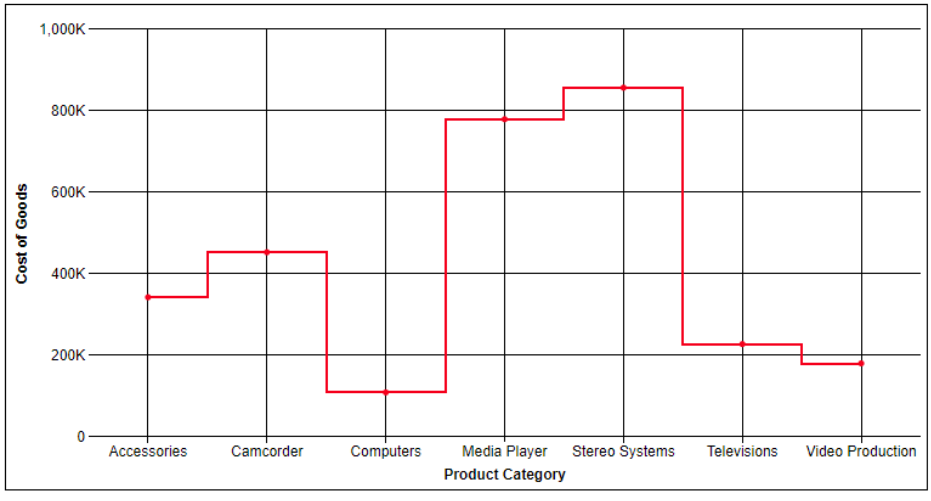
lineConnection プロパティを "curved" に変更すると、下図のグラフが生成されます。



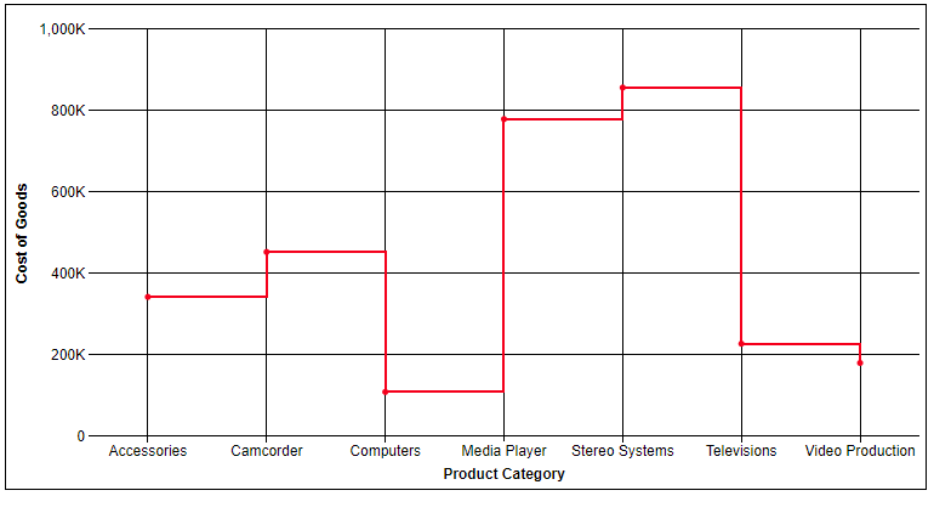
lineConnection プロパティを "stepBefore" に変更すると、下図のグラフが生成されます。このグラフでは、すべての線セグメントが縦方向と横方向のみの直線で表され、マーカーは線セグメントの終点に表示されています。



lineConnection プロパティを "stepBetween" に変更すると、下図のグラフが生成されます。このグラフでは、すべての線セグメントが縦方向と横方向のみの直線で表され、マーカーは線セグメントの中点に表示されています。



lineConnection プロパティを "stepAfter" に変更すると、下図のグラフが生成されます。このグラフでは、すべての線セグメントが縦方向と横方向のみの直線で表され、マーカーは線セグメントの始点に表示されています。



棒グラフ、折れ線グラフ、面グラフの塗りつぶし効果の制御

lineFillEffect プロパティは、積み上げ折れ線グラフまたはパーセント折れ線グラフで、線ライザとベースライン間の領域および 2 つの異なるライザ間の領域を特定の色で塗りつぶします。

構文　棒グラフ、折れ線グラフ、面グラフの lineFillEffect プロパティの設定

```
"blaProperties": {
   "lineFillEffect": "string"
   }
```

説明

`lineFillEffect: "string"`

塗りつぶし効果の色を指定します。有効な値には、次のものがあります。

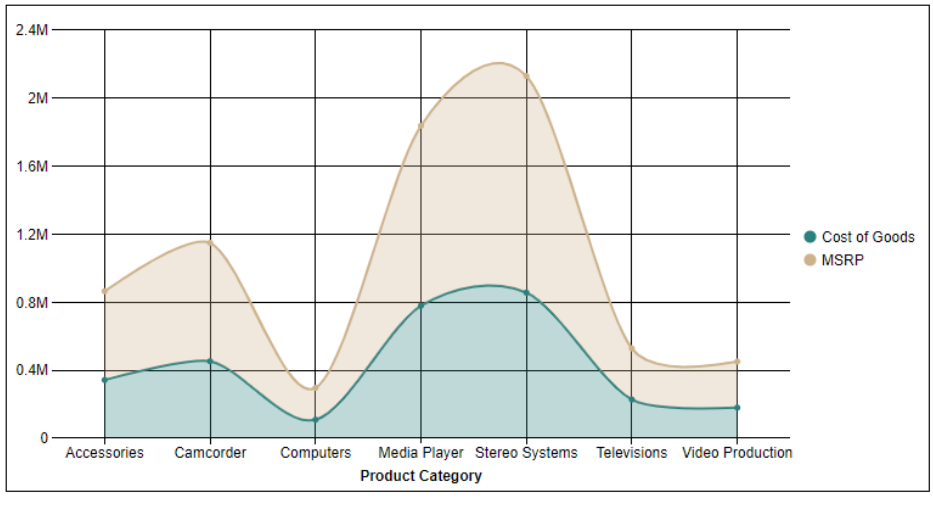
- undefined (無効) - 塗りつぶし効果を使用しません。これがデフォルト値です。
- "seriesAuto" - 塗りつぶし効果にやや薄いシリーズ色を使用します。
- "seriesLighten" - 塗りつぶし効果に薄いシリーズ色を使用します。
- "seriesLightenOpaque" - seriesAuto 塗りつぶし効果より薄い色、 seriesLighten 塗りつぶし効果より透明度の低い色を使用します。
- 色名または数値指定の文字列で定義する色
- 文字列で定義するグラデーション
- パーセント文字列 ("-100%" から "100%")。パーセント文字列で指定した不透明度がシリーズ色に適用されます。

例　折れ線グラフの lineFillEffect プロパティの設定

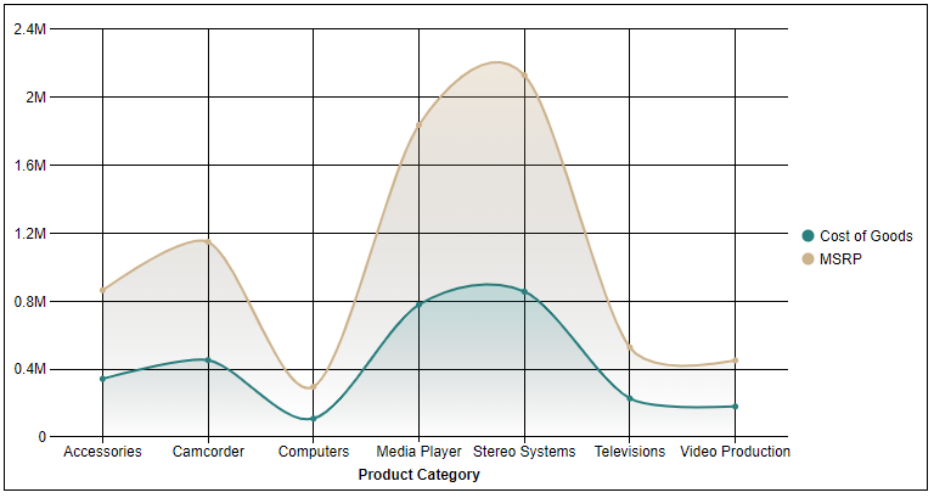
次のリクエストは、縦折れ線グラフを生成し、2 つのシリーズを表示します。シリーズ 0 の線は青緑色、シリーズ 1 の線は黄褐色です。塗りつぶし効果は seriesAuto です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {
   "seriesLayout": "stacked", "lineConnection": "curved",
   "lineFillEffect": "seriesAuto"},
"series": [
   {"series": 0, "color": "teal"},
   {"series": 1, "color": "tan"}],
"legend": {"visible": true}
*END
ENDSTYLE
END
```

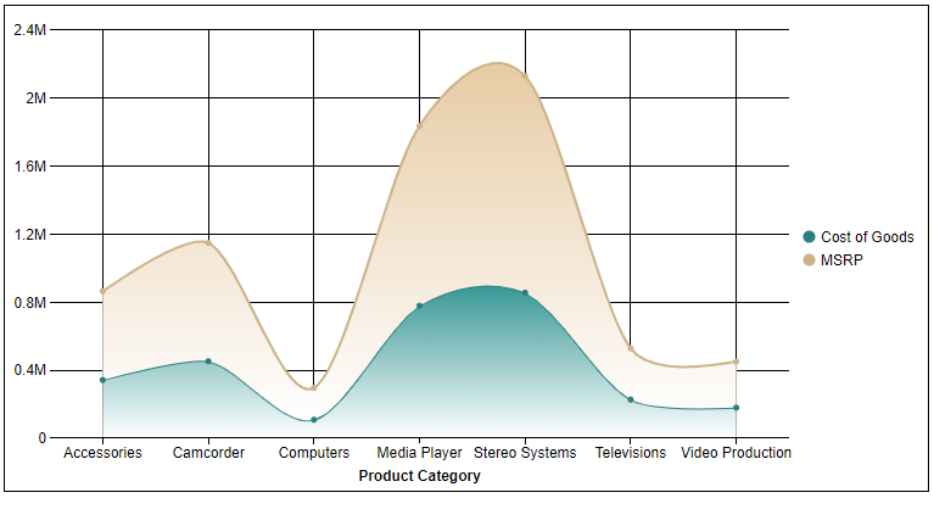
下図のように、塗りつぶし効果には、やや透明なシリーズ色が使用されます。この色は、グラフエンジンによって自動的に生成されます。



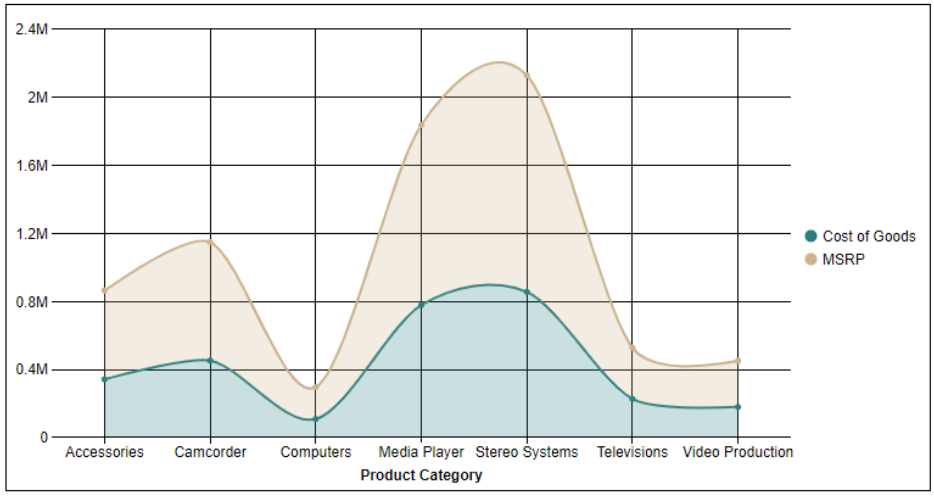
lineFillEffect プロパティを "seriesLighten" に変更すると、下図のグラフが生成されます。このグラフの塗りつぶし効果には、seriesAuto 塗りつぶし効果より薄い色が使用されます。



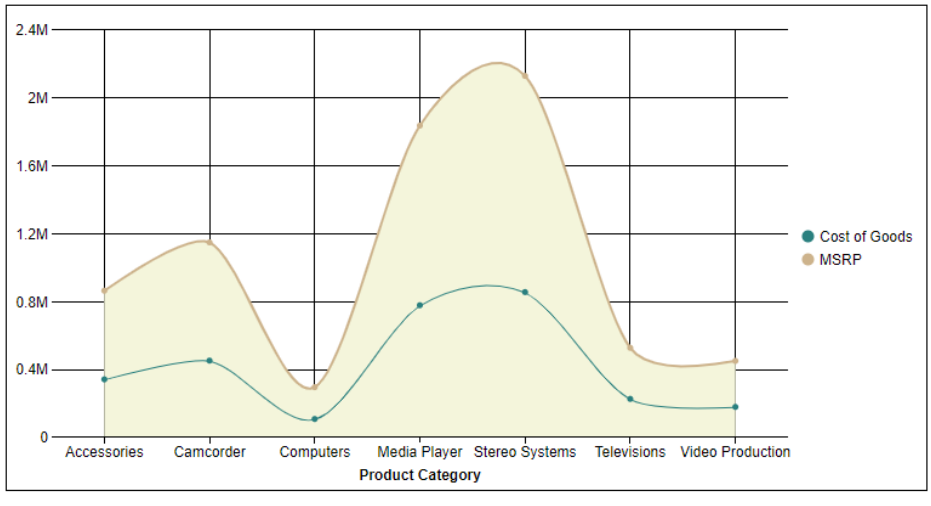
lineFillEffect プロパティを "seriesLightenOpaque" に変更すると、下図のグラフが生成されます。このグラフの塗りつぶし効果には、seriesAuto 塗りつぶし効果より薄い色、seriesLighten 塗りつぶし効果より透明度の低い色が使用されます。



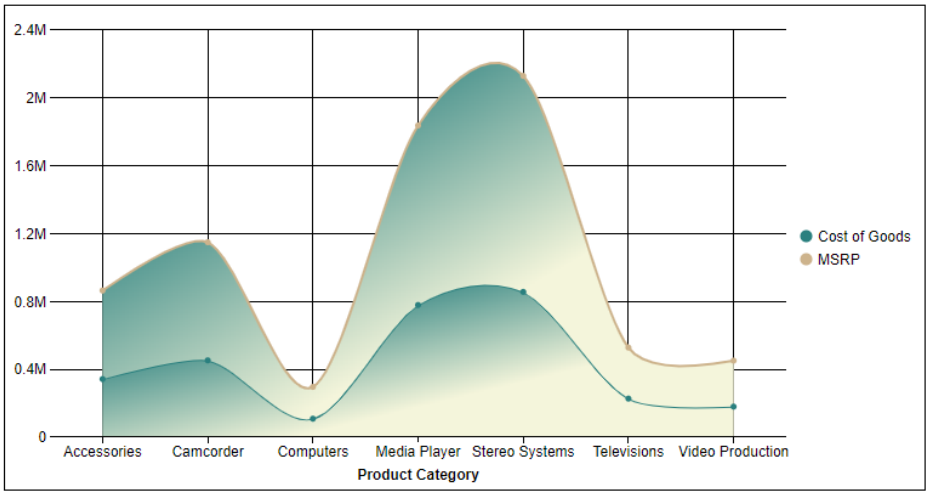
lineFillEffect プロパティを "25%" に変更すると、下図のグラフが生成されます。このグラフの塗りつぶし効果には、不透明度が 25% のシリーズ色が使用されます。



lineFillEffect プロパティを "beige" に変更すると、下図のグラフが生成されます。このグラフの塗りつぶし効果には、ベージュが使用されます。



lineFillEffect プロパティを "linear-gradient(0,0,100%,100%, 20% teal, 65% beige)" に変更すると、下図のグラフが生成されます。このグラフの塗りつぶし効果には、青緑色からベージュに変化する線状グラデーションが使用されます。



棒グラフ、折れ線グラフ、面グラフのライザグループ間隔の制御

barGroupGapWidth プロパティは、ライザグループ間の間隔を制御します。

構文　棒グラフ、折れ線グラフ、面グラフのライザグループ間隔の制御

```
"blaProperties": {
   "barGroupGapWidth": number   }
```

説明

`"barGroupGapWidth": number`

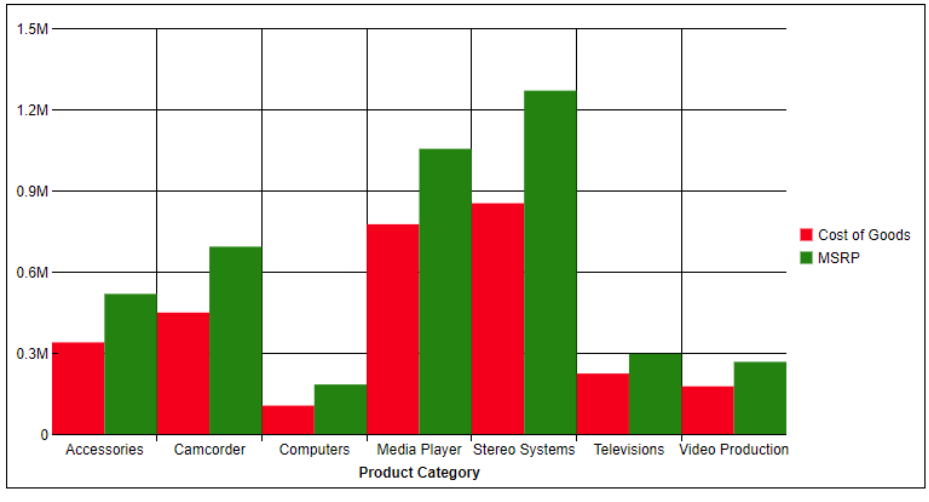
ライザのグループ間の間隔を定義する 0.0 から 1.0 までの数値です。この数値は、各グループで利用可能な幅全体に対する割合を示します。デフォルト値は 0.2 です。

例　棒グラフのライザグループ間隔の指定

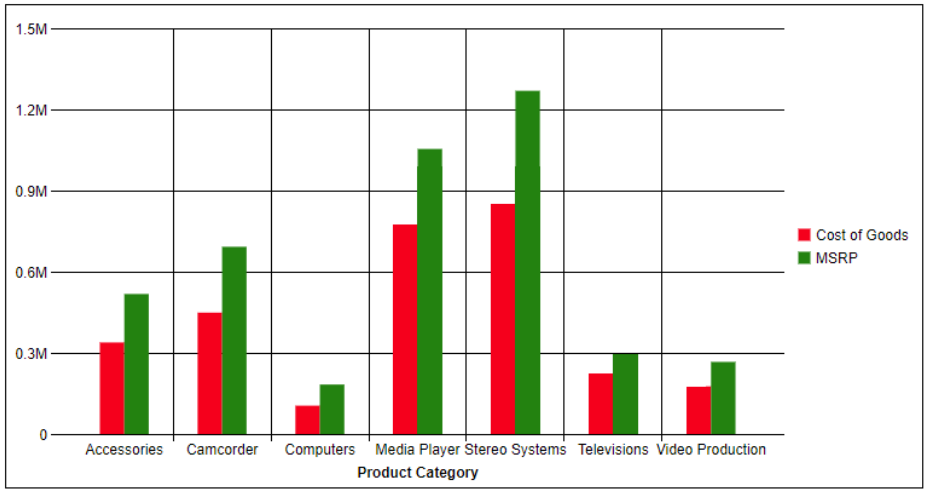
次のリクエストは、ライザグループの間隔を 0.0 (ゼロ) に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {"barGroupGapWidth": 0.0}
*END
ENDSTYLE
END
```

出力されたグラフでは、ライザグループ間 (製品区分別のライザ) に隙間はありません。



barGroupGapWidth プロパティを 0.5 に変更すると、下図のグラフが生成されます。



棒ライザの角の丸みの制御

barEffect プロパティを使用して、常に両端の角に丸みを適用した棒ライザを生成したり、棒が X 軸上から開始する場合は、もう一方の端のみの角に丸みを適用した棒ライザを生成することができます。

構文　棒ライザの角の丸みの制御

棒グラフの blaProperties ブロックに、barEffect プロパティを追加します。

```
"blaProperties": {
    "barEffect": "string",
}
```

説明

```
barEffect: 'string'
```

棒ライザの角の形状を指定します。次のいずれかの値です。

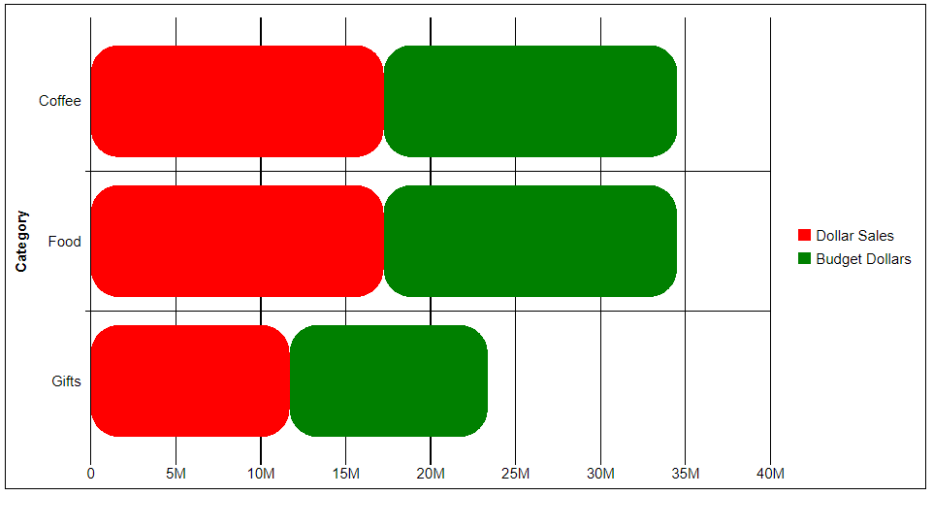
- "halfRound" - X 軸から開始される側の棒の角には丸みを適用せず、それ以外の角には丸みを適用します。
- "fullRound" - すべての棒の角に丸みを適用します。

例　棒ライザすべての角に丸みを適用

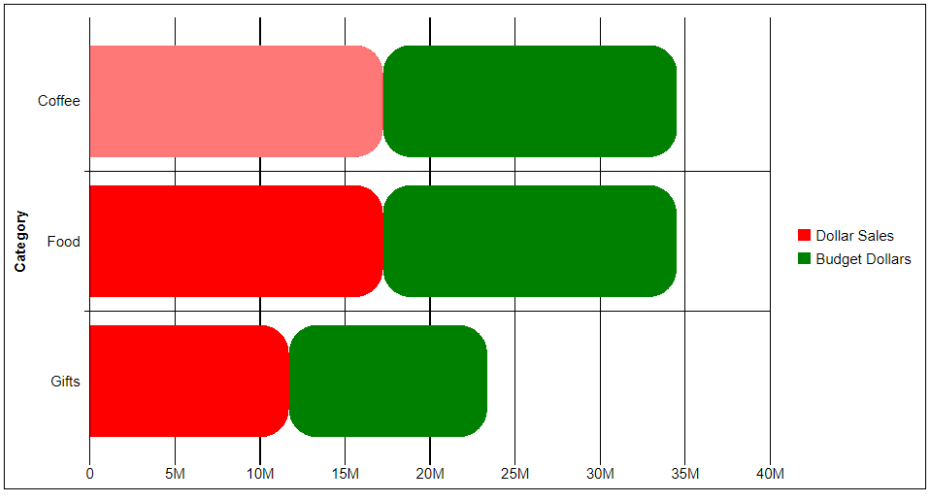
次のリクエストは、すべての棒ライザの角に丸みを適用します。

```
GRAPH FILE ibisamp/ggsales
SUM DOLLARS BUDDOLLARS
BY CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS
"blaProperties": {
    "barEffect": "fullRound",
    "orientation": "horizontal",
    "seriesLayout": "stacked"
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



barEffect プロパティを "halfRound "に変更すると、下図のように、軸から開始される側の棒の角に丸みを適用しないグラフが生成されます。



棒グラフ、折れ線グラフ、面グラフの複数 Y 軸表示の制御

splitY プロパティは、2 軸グラフ (グラフの左側と右側に Y 軸を配置) または分割軸グラフ (複数の Y 軸を上下に積み上げて配置) として複数の Y 軸を表示するかどうかを制御します。

構文　　棒グラフ、折れ線グラフ、面グラフの複数 Y 軸表示の制御

```
"blaProperties": {"splitY": boolean}
```

説明

`"splitY": boolean`

Y 軸を 2 軸にするか軸分割にするかを決定します。有効な値には、次のものがあります。

- true - 軸分割で Y 軸を描画します。これがデフォルト値です。
- false - 2 軸で Y 軸を描画します。

注意：メジャーがすべて同一色で表示される分割 Y 軸グラフを生成するには、次のスタイルシート宣言を使用します。

```
type=report, chart-color-measures=off,$
```

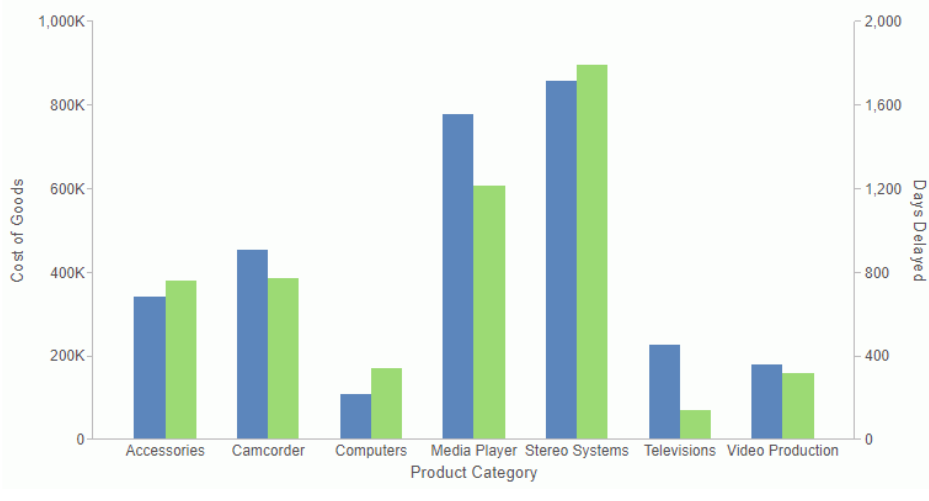
chart-color-measures のデフォルト値は on です。この場合、メジャーごとに異なる色を生成します。

例　複数 Y 軸表示の制御

次のリクエストは、y1-axis と y2-axis に 1 つずつメジャーを割り当て、2 軸グラフとして表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DAYSDELAYED
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=daysdelayed, bucket=y-axis(2),$
*GRAPH_JS
"blaProperties": {"splitY": false},
"legend": {"visible": false}
*END
ENDSTYLE
END
```

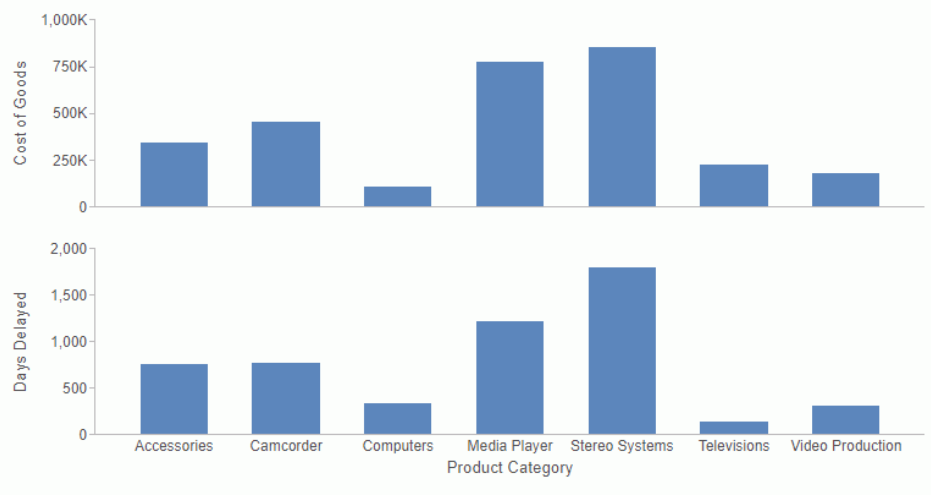
出力結果は次のとおりです。



次のリクエストでは、分割 Y 軸を生成し、chart-color-measures を off に設定します。この場合、両方のメジャーが同一色で表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US DAYSDELAYED
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=daysdelayed, bucket=y-axis(2),$
type=report, chart-color-measures=off,$
*GRAPH_JS
"blaProperties": {"splitY": true},
"legend": {"visible": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



積み上げ棒グラフ、折れ線グラフ、面グラフの合計ラベルの制御

stackTotalLabel プロパティは、積み上げ棒グラフ、積み上げ折れ線グラフ、積み上げ面グラフ (seriesLayout='stacked') の合計ラベルの外観を制御します。

構文　積み上げ合計の生成

```
"blaProperties": {
   "stackTotalLabel": {
      "visible": boolean,
      "font": "string",
      "color": "string",
      "numberFormat": numformat   },
}
```

説明

`"visible": boolean`

積み上げ合計ラベルの表示を定義します。有効な値には、次のものがあります。

- true - 合計ラベルを表示します。
- false - 合計ラベルを表示しません。これがデフォルト値です。

`"font": "string"`

積み上げ合計ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

積み上げ合計ラベルの色を定義する文字列です。デフォルト値は "black" です。

`"numberFormat": numformat`

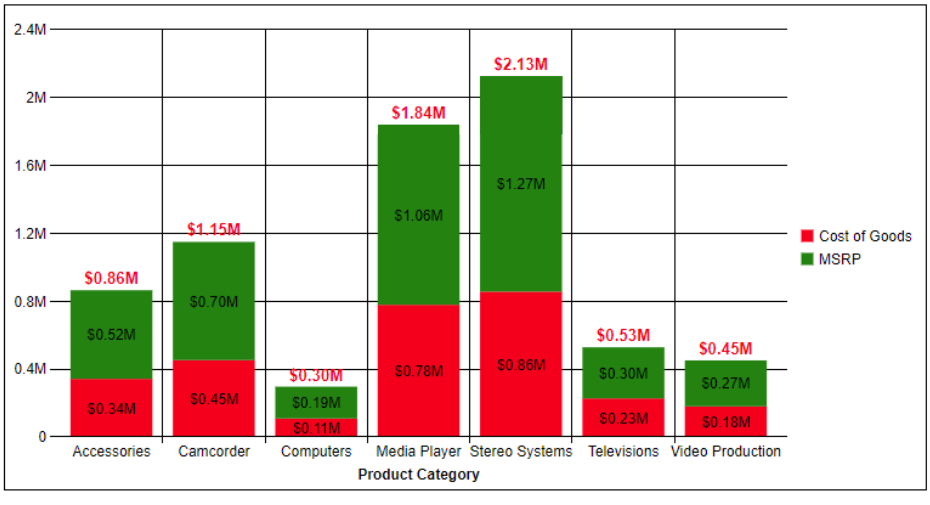
ラベルの数値フォーマットを定義します。数値フォーマットは、JSON オブジェクト、フォーマット文字列、ユーザ定義関数のいずれかで指定することができます。デフォルト値は "auto" です。また、autoNumberFormats を使用して、積み上げ合計ラベルのすべてに数値フォーマットを適用することもできます。数値フォーマットについての詳細は、86 ページの 「 数値のフォーマット設定 」 を参照してください。

例　積み上げ棒グラフの合計ラベルの生成

次のリクエストは、積み上げ棒グラフを生成し、各積み上げ棒ライザの上側に積み上げ合計を赤色で表示します。series オブジェクトでデータラベルの表示を有効にし、dataLabels オブジェクトでデータテキストの表示を有効にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=report, chart-series-layout=stacked,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"blaProperties": {"stackTotalLabel": {
   "visible": true, "font": "10pt Serif, bold", "color": "red",
   "numberFormat": {"mode": "currency", "grouping": "M"}},
"dataLabels": {"visible": true,  "position": "center",
   "numberFormat": {"mode": "currency", "grouping": "M"}},
"legend": {"visible": true}
*END
ENDSTYLE
END
```

出力結果では、データラベルが棒ライザの内側に、積み上げ合計が棒ライザの上側に表示されています。



棒、折れ線、面グラフでの特定データラベルの表示

すべてのデータラベルを表示する、またはデータラベルを表示しない代わりに、1 つまたは複数のシリーズに対して、以下のデータラベルを 1 つまたは複数表示するよう指定することができます。

- **first** シリーズで最初の表示可 (ミッシング以外) データポイントのデータラベルを表示します。
- **last** シリーズで最後の表示可 (ミッシング以外) データポイントのデータラベルを表示します。
- **high** シリーズが割り当てられた Y 軸で計測されるシリーズで最高値のデータポイントのデータラベルを表示します。同一最高値として複数のポイントが存在する場合、すべてが表示されます。
- **low** シリーズが割り当てられた Y 軸で計測されるシリーズで最低値のデータポイントのデータラベルを表示します。同一最低値として複数のポイントが存在する場合、すべてが表示されます。

構文　棒、折れ線、面グラフでの特定データラベルの表示

特定のデータラベルを表示するには、シリーズの dataLabels プロパティに dataLabelMask 配列を追加します。

```
"series":     [
        {"series": number,
            "dataLabels": {
                "visible": true,
                "dataLabelMask": [ "string", "string" ]
             }

        }

    ]
```

説明

`"series": number`

dataLabelsMask の適用先シリーズを数値または文字列 "all" で指定します。

```
"dataLabelMask": [ "string", "string" ]
```

表示するデータラベルを指定する 1 から 4 の文字値です。有効な値には、次のものがあります。

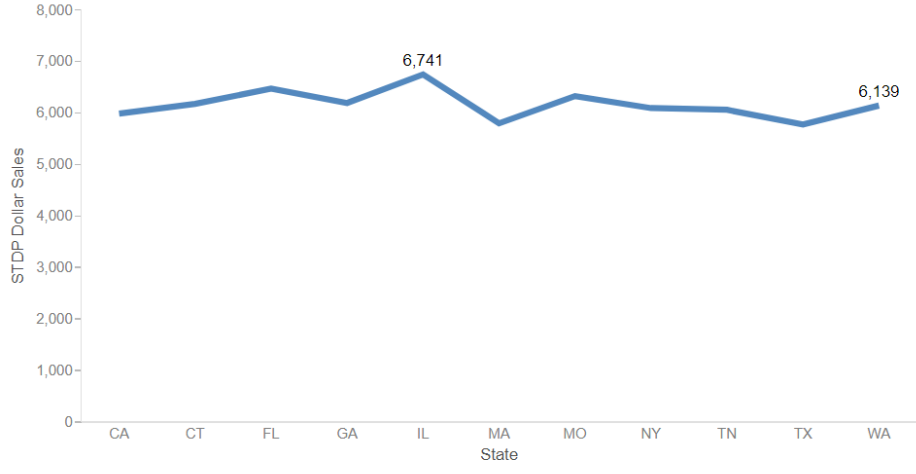
- **first** シリーズで最初の表示可 (ミッシング以外) データポイントのデータラベルを表示します。
- **last** シリーズで最後の表示可 (ミッシング以外) データポイントのデータラベルを表示します。
- **high** シリーズが割り当てられた Y 軸で計測されるシリーズで最高値のデータポイントのデータラベルを表示します。同一最高値として複数のポイントが存在する場合、すべてが表示されます。
- **low** シリーズが割り当てられた Y 軸で計測されるシリーズで最低値のデータポイントのデータラベルを表示します。同一最低値として複数のポイントが存在する場合、すべてが表示されます。

例 シリーズ最高値および最終データラベルの表示

次のリクエストは、シリーズ 0 (ゼロ) の最高値および最後のデータポイントのデータラベルを表示します。

```
GRAPH FILE ibisamp/ggsaLes
SUM STDP.DOLLARS
BY ST
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/WFC/Global/Themes/Standard/Default/theme.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"series":     [
        {"series": 0,
           "dataLabels": {
                "visible": true,
                "dataLabelMask": [ "high", "last" ],
                "font": "10pt Sans-Serif"
            }
        }
    ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



複合グラフのプロパティの指定

comboCharts プロパティを "series#": "riserShape" プロパティとともに使用して、複合グラフを定義します。複合グラフは、棒ライザ、面ライザ、線ライザを任意に組み合わせて作成されるグラフです。棒グラフ、折れ線グラフ、面グラフを生成する際に、シリーズ固有の riserShape プロパティを使用して、各シリーズのライザ形状を棒、線、面のいずれかとして定義することができます。詳細は、482 ページの 「 棒グラフ、折れ線グラフ、面グラフのシリーズライザ形状の定義 」 を参照してください。これらのプロパティは、ライザのレイアウト (stacked、absolute、percent、sideBySide) をシリーズごとに制御します。

構文　　ライザ形状の組み合わせ指定

```
"blaProperties": {
   "comboCharts": {
      "barSeriesLayout": "string",
      "lineSeriesLayout": "string",
      "areaSeriesLayout": "string",
   }
}
```

説明

`"barSeriesLayout": "string"`

棒ライザ ("series#": "markerShape": "bar") に割り当てられたシリーズのライザレイアウトを定義します。有効な値は、"absolute"、"percent"、"sideBySide"、"stacked" のいずれかです。デフォルト値は undefined です。シリーズのレイアウトについての詳細は、514 ページの「 棒グラフ、折れ線グラフ、面グラフのシリーズレイアウトの制御 」 を参照してください。

`"lineSeriesLayout": "string"`

線ライザ ("series#": "markerShape": "line") に割り当てられたシリーズのライザレイアウトを定義します。デフォルト値は undefined です。有効な値は、"absolute"、"percent"、"stacked" のいずれかです。シリーズのレイアウトについての詳細は、514 ページの「 棒グラフ、折れ線グラフ、面グラフのシリーズレイアウトの制御 」 を参照してください。

`"areaSeriesLayout": "string"`

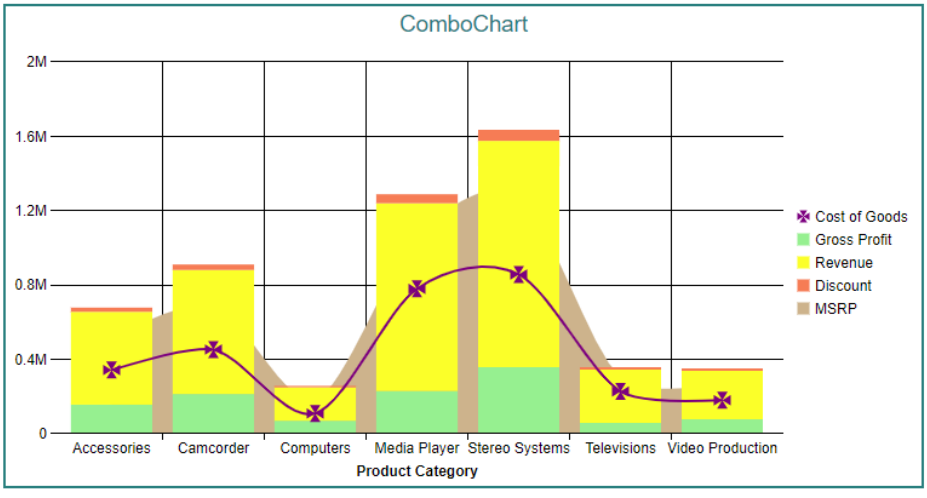
面ライザ ("series#": "markerShape": "area") に割り当てられたシリーズのライザレイアウトを定義します。デフォルト値は undefined です。有効な値は、"absolute"、"percent"、"stacked" のいずれかです。シリーズのレイアウトについての詳細は、514 ページの「 棒グラフ、折れ線グラフ、面グラフのシリーズレイアウトの制御 」 を参照してください。

例　複合グラフのシリーズレイアウトの制御

次のリクエストは、1 つの線ライザ (マーカー形状はパイレーツクロス)、1 つの面ライザ、および 3 つの積み上げ棒ライザで構成された複合グラフを作成します。series オブジェクトで、各シリーズに異なるライザタイプを割り当てます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
type=data, column=n6, bucket=y-axis,$
*GRAPH_JS
"legend": {"visible": true},
"border": {"width": 2, "color": "teal"},
"title": {"visible": true, "text": "ComboChart", "font": "14pt Sans-Serif",
"color": "teal"},
"blaProperties": {
   "lineConnection": "curved",
   "comboCharts": {
      "barSeriesLayout": "stacked",
      "lineSeriesLayout": "absolute",
      "areaSeriesLayout": "undefined"}},
"series": [
   {"series": 0, "color": "purple", "riserShape": "line", "border":
{"width": 2},
      "marker": {"shape": "pirateCross", "size": 14, "visible": true}},
   {"series": 1, "color": "lightgreen", "riserShape": "bar"},
   {"series": 2, "color": "yellow", "riserShape": "bar"},
   {"series": 3, "color": "coral", "riserShape": "bar"},
   {"series": 4, "color": "tan", "riserShape": "area"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



注意：comboChart プロパティと "series#": "riserShape" プロパティの両方を使用して複合グラフを定義した場合、最初に面ライザが背面に、次に棒ライザ、最後に線ライザが前面に描画されます。

棒グラフ、折れ線グラフ、面グラフでの Null 値表示の制御

グラフデータを生成する場合、Null (ミッシング) データをグラフエンジンに送信する方法を、Reporting Server に指示する必要があります。この指示を省略すると、Null 値は常に値 0.0 で置き換えられます。これを実行するには、コマンド「ON GRAPH SET VZERO OFF」を発行します。さらに、showNullGroups プロパティおよび missingDataMode プロパティを使用して、グラフエンジンでの Null 値の表示方法を制御することができます。

グラフでは次のような状況が一般的であり、これらはグラフエンジンで制御することができます。

- 棒グラフで、X 軸グループにライザが含まれるものと含まれないものが存在することが考えられます。この場合、"showNullGroups":true を使用してすべてのグループ (データが存在しないグループを含む) を表示することも、"showNullGroups":false を使用して空のグループを除外し、ライザをまとめて表示することもできます。
- 次の 3 つの状況は、折れ線グラフおよび面グラフで、特に時系列を含む場合に一般的です。これらの状況では、"showNullGroups" は true です。
  - **折れ線シリーズ上のミッシングデータを間隔として表示**　この表示方法を実装するには、"missingDataMode":"none" を使用します。

- **間隔を表示する代わりに、実データポイント間に補間線を表示**　この表示方法を実装するには、"missingDataMode":"linear" を使用します。
- **間隔を表示する代わりに、ミッシングデータポイントを 0.0 にマッピング**　この方法は、VZERO ON 設定と同等で、"missingDataMode":"zero" を使用して表示します。

構文　棒グラフ、折れ線グラフ、面グラフでの Null 値表示の制御

注意：ミッシングデータの表示方法をグラフエンジンに指示するには、リクエストに ON GRAPH SET VZERO OFF コマンドを含める必要があります。

グラフから Null 値を除外するには、showNullGroups プロパティを false に設定します。

```
"showNullGroups": false
```

グループ全体が Null 値の場合、グループ全体がグラフから除外され、グループラベルは表示されません。グループ全体が Null ではない場合、Null データは、これらのミッシングデータが存在するライザまたは折れ線内の間隔として表示されます。

Null 値を表示し、かつその表示方法を制御するには、showNullGroups プロパティを true に設定した上で、blaProperties: missingDataMode 値を設定します。

```
"showNullGroups": true,
"blaProperties": {
                 "missingDataMode": "string"
                 }
```

説明

```
"missingDataMode": "string"
```

Null データの表示方法を定義します。有効な値には、次のものがあります。

- **"none"**　棒グラフ、折れ線グラフ、面グラフで、Null 値を間隔として表示します。グループ全体が Null 値の場合でも、グループラベルが表示されます。これがデフォルト値です。
- **"zero"**　グラフを表示する際に、Null 値は 0.0 として処理されます。
- **"linear"**　グラフエンジンによって存在する値間に補間線が描画され、破線で表示されます。

この線にフォーマットを設定するには、missingDataLineStyle プロパティを使用します。以下はその例です。

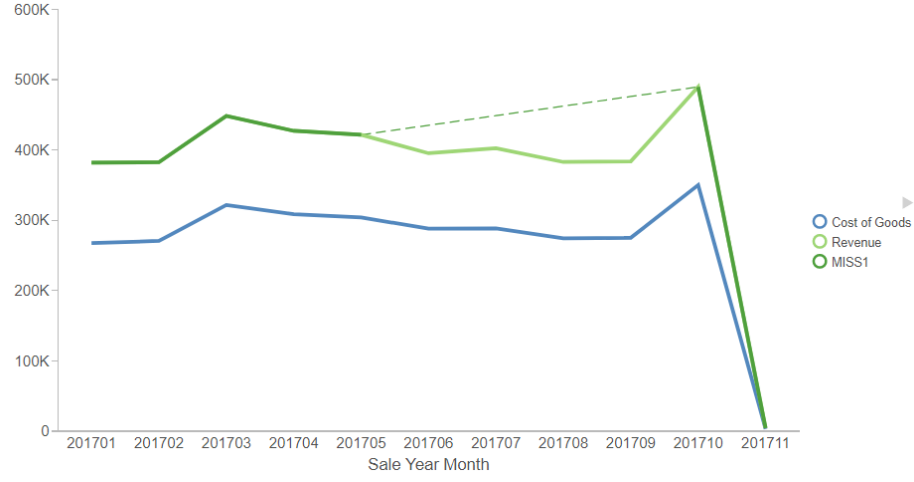
```
"blaProperties": {
         "missingDataMode": "linear",
         "missingDataLineStyle": {"color":"red", "dash": "1 1"}
}
```

例　Null 値の表示を制御

次のリクエストは、年月の組み合わせのいくつかがミッシングとなる一時項目 (DEFINE) を定義し、showNullGroups を true、missingDataMode を linear に設定します。

```
DEFINE FILE WF_RETAIL_LITE
MISS1/D12.2 MISSING ON =
            IF TIME_YEARMTH GE '201706' AND TIME_YEARMTH LT '201710'
            THEN MISSING ELSE REVENUE_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM
COGS_US
REVENUE_US
MISS1
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VZERO OFF
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
*GRAPH_JS
"showNullGroups": true,
        "blaProperties": {
                 "missingDataMode": "linear"
        }
*END
ENDSTYLE
END
```

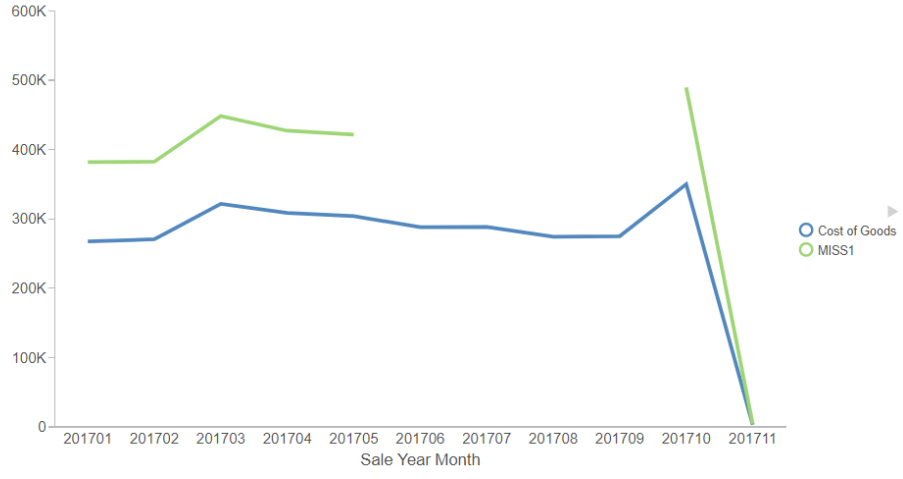
出力結果は次のとおりです。破線は、Null データを示しています。



次のリクエストは、年月の組み合わせのいくつかがミッシングとなる一時項目 (DEFINE) を定義し、showNullGroups を true、missingDataMode を none に設定します。

```
DEFINE FILE WF_RETAIL_LITE
MISS1/D12.2 MISSING ON =
            IF TIME_YEARMTH GE '201706' AND TIME_YEARMTH LT '201710'
            THEN MISSING ELSE REVENUE_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM
COGS_US
MISS1
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VZERO OFF
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS
"showNullGroups": true,
        "blaProperties": {
                "missingDataMode": "none"

        }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。折れ線内の間隔は、Null データを示しています。



次のリクエストは、年月の組み合わせのいくつかがミッシングとなる一時項目 (DEFINE) を定義し、showNullGroups を true、missingDataMode を zero に設定します。

```
DEFINE FILE WF_RETAIL_LITE
MISS1/D12.2 MISSING ON = IF TIME_YEARMTH EQ '201706'
            THEN MISSING ELSE REVENUE_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM
COGS_US
REVENUE_US
MISS1
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VZERO OFF
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N4, BUCKET=y-axis, $
*GRAPH_JS
"showNullGroups": true,
        "blaProperties": {
                "missingDataMode": "zero"
        }
*END
ENDSTYLE
END
```

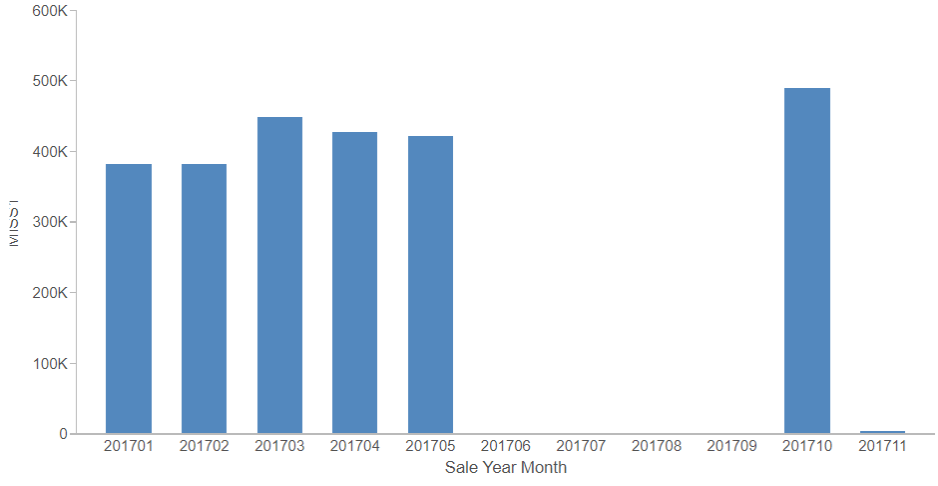
出力結果は次のとおりです。0 (ゼロ) まで下降し、さらに上昇する破線は、Null データを示しています。

例　棒グラフでの Null グループの表示および除外

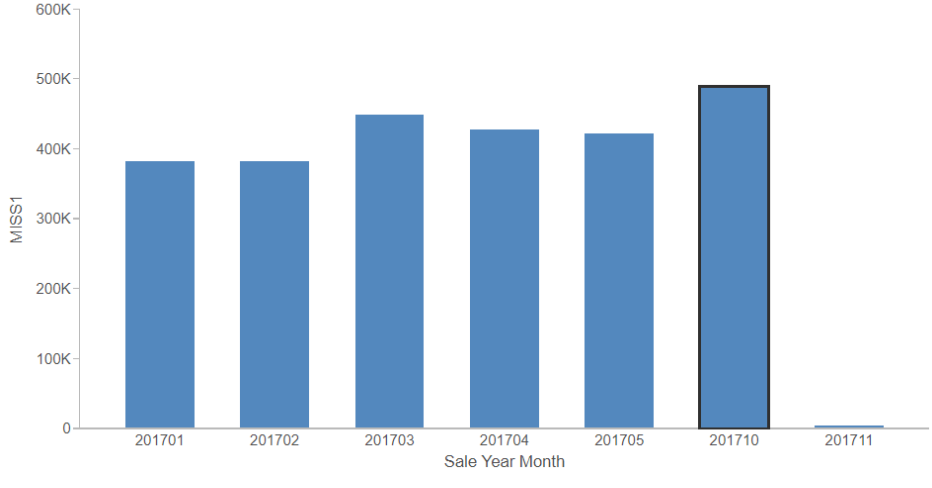
次のリクエストは、棒グラフに Null グループを作成し、"showNullGroups": true を使用してこれらを表示します。

```
DEFINE FILE WF_RETAIL_LITE
MISS1/D12.2 MISSING ON =
           IF TIME_YEARMTH GE '201706' AND TIME_YEARMTH LT '201710'
           THEN MISSING ELSE REVENUE_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM
MISS1
BY TIME_YEARMTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VZERO OFF
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS
"showNullGroups": true
*END
ENDSTYLE
END
```

出力結果は次のとおりです。グループラベルが表示され、ライザ間に間隔があります。



"showNullGroups": false に設定すると、次のグラフが表示されます。ここでは、空のグループラベルは除外され、ライザ間の間隔もありません。



ボックスプロットおよび統計ボックスプロットのプロパティ (boxPlotProperties)

これらのプロパティは、ボックスプロットまたは統計ボックスプロットの全体的な外観を制御します。

次のコードセグメントは、プロパティおよびデフォルト値を示しています。

```
"boxPlotProperties": {
   "hatWidth": "50%",
   "drawHatAsBox": false,
   "notch": {
          "width": undefined,
          "multiplier": 1.58
   "medianLine": {
      "width": 1,
      "color": "black",
      "dash": ""
   },
        "connectorLine": {
                        "width": 1,
                        "color": "black",
                        "dash": ""
   },
   "meanMarker": {
      "visible": false,
      "color": "orange",
      "border": {
         "width": 1,
         "color": undefined,
         "dash:" undefined
            },
      "size": 8,
      "shape": "circle",
      "rotation": 0,
      "tooltip": undefined
      }

}
```

ボックスプロットのハット幅の指定

hatWidth プロパティは、ボックスプロットの上部と下部に描画されるハットの幅を制御します。

構文　ボックスプロットのハット幅の設定

```
boxPlotProperties: {
   "hatWidth": width
```

説明

`"hatWidth:width`

ハットの幅をピクセル単位で定義する数値、またはパーセント記号を含み二重引用符 (") で囲まれたパーセント値を表す文字列 ("0%"..."100%") です。デフォルト値は "50%" です。

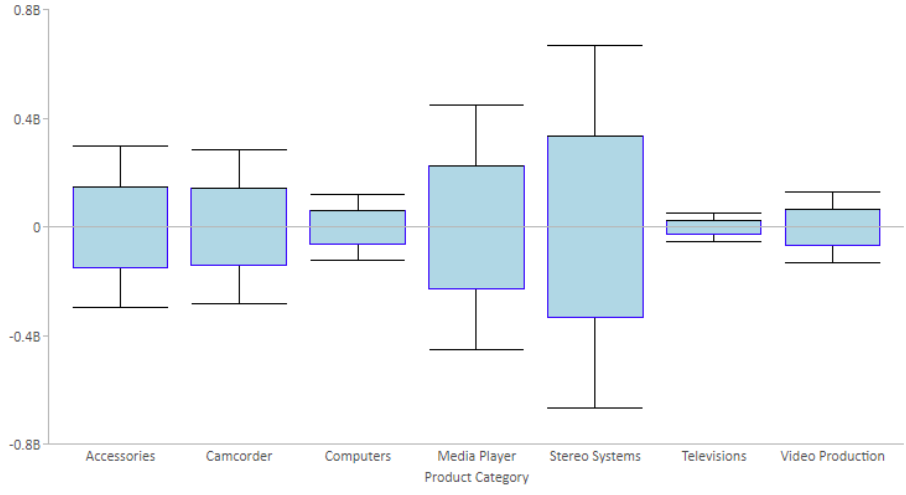
注意：ハットを箱型として描画する場合、ハットの色およびフォーマットは series および border プロパティで制御します。ハットを箱型として描画しない場合、ハットの色およびフォーマットは connectorLine プロパティで制御します。ハット全体を削除するには、border プロパティで幅を 0 (ゼロ) に設定します (hatWidth: "0%" または hatWidth: 0 と同一)。

例　ボックスプロットのハット幅の指定

次のリクエストは、ボックス幅に対するハット幅の割合を 100% に設定します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=diff2, bucket=min,$
type=data, column=diff1, bucket=lower,$
type=data, column=c4, bucket=median,$
type=data, column=diff3, bucket=upper,$
type=data, column=diff4, bucket=max,$
*GRAPH_JS
"boxPlotProperties": {
      "drawHatAsBox": false,
      "hatWidth": "100%"
      },
"series": [
 {"series": 0, "color": "lightblue", "border": {
    "width": 1, "color": "blue"}
    }]
*END
ENDSTYLE
END
```

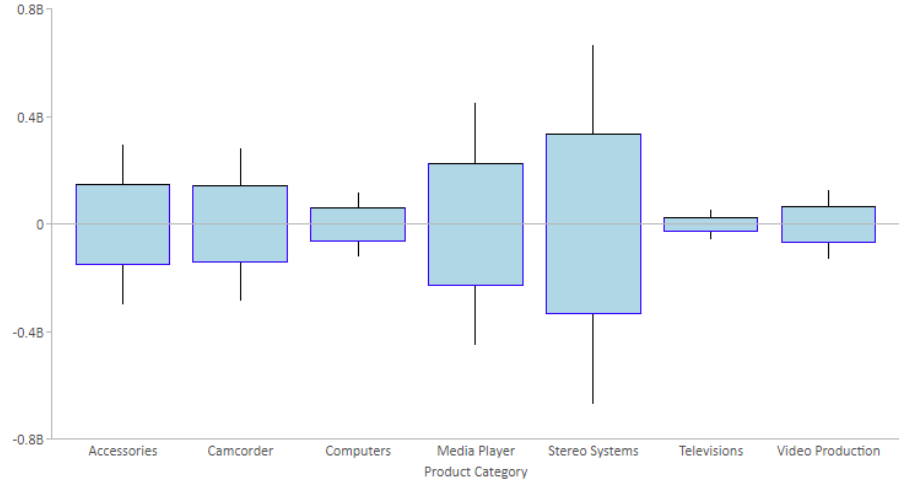
出力結果は次のとおりです。



hatwidth プロパティを 0 (ゼロ) に設定すると、ハットは削除されますが、接続線は保持されます。

```
"boxPlotProperties": {
   "hatWidth": 0},
```

出力結果は次のとおりです。



ボックスプロットのハットの外観制御

drawHatAsBox プロパティは、ボックスプロットのハット (上部および下部) の外観を制御します。

構文　ボックスプロットのハットの外観制御

```
"boxPlotProperties": {
   "drawHatAsBox": boolean   },
```

説明

`"drawHatAsBox:boolean`

有効な値には、次のものがあります。

- true - ボックスプロットのハットを箱型で描画します。これがデフォルト値です。
- false - 通常のハットを描画します。

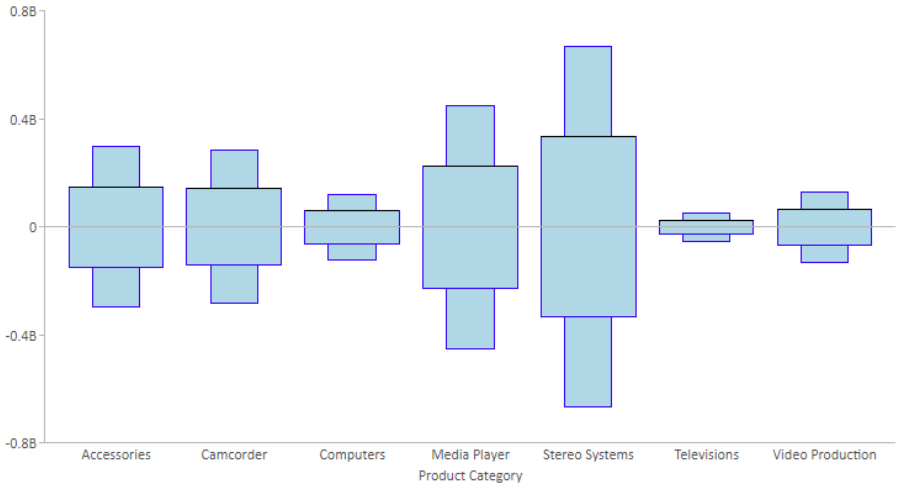
注意：ハットを箱型として描画する場合、ハットの色およびフォーマットは series および border プロパティで制御します。 ハットを箱型として描画しない場合、ハットの色およびフォーマットは connectorLine プロパティで制御します。

例　　ボックスプロットでの箱型ハットの描画

次のリクエストは、ボックスプロットのハットを箱型で描画します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=diff2, bucket=min,$
type=data, column=diff1, bucket=lower,$
type=data, column=c4, bucket=median,$
type=data, column=diff3, bucket=upper,$
type=data, column=diff4, bucket=max,$
*GRAPH_JS
"boxPlotProperties": {
      "drawHatAsBox": true,
      },
"series": [
 {"series": 0, "color": "lightblue", "border": {
    "width": 1, "color": "blue"}
    }]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ボックスプロットのノッチの外観制御

notch プロパティは、中央値線にオプションで追加するボックスのインデントを制御します。

構文　ボックスプロットのノッチの外観制御

```
"boxPlotProperties": {
          "notch": {
            "width": "string",
            "multiplier": number
        },
```

説明

`"notch"`
　中央値線にオプションで追加するボックスのインデントのプロパティを定義します。

　`"width": "string"`
　　ボックス幅のパーセントとしてノッチの切り込みを定義するパーセント文字列 ("100%" 未満) です。デフォルト値は、ノッチを描画しない undefined です。

"multiplier": number
中央値ポイントと第 1 四分位および第 3 四分位ポイント間に描画するノッチの幅を定義するのに使用する要素です。デフォルト値は、1.58 です。ノッチ幅の計算に使用する数式は、次のとおりです。

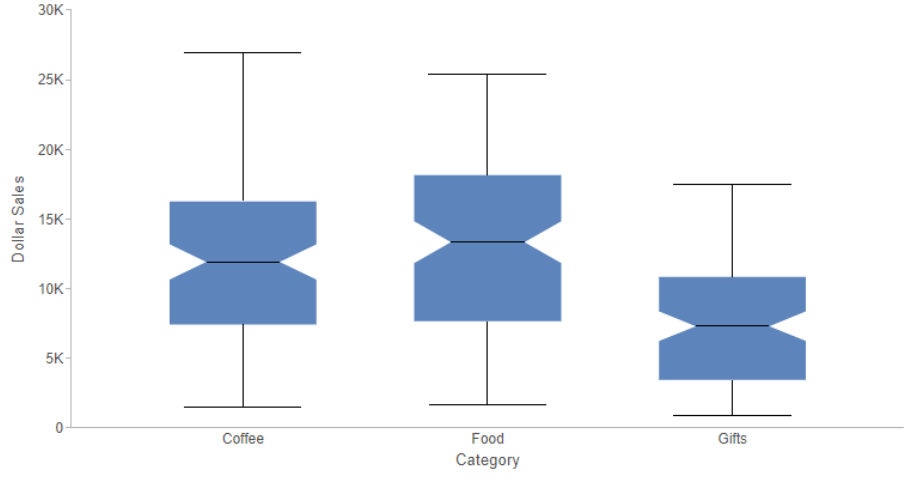
```
+/- notch.multiplier * (box upper value - box lower value) /
sqrt(data point count)
```

例　統計ボックスプロットのノッチのフォーマット設定

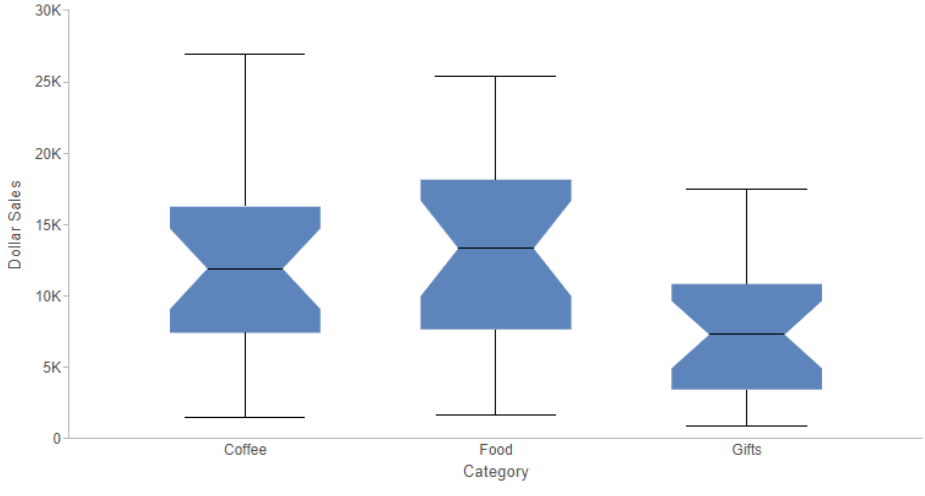
次のリクエストは、統計ボックスプロットを生成し、ハット幅をボックス幅の 80% に設定し、中央値線にノッチを 50% のインデントで設定します。

```
GRAPH FILE GGSALES
PRINT DOLLARS
BY CATEGORY
IF ST EQ 'NY'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STAT-BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=n1, bucket=x-axis, $
type=data, column=n2, bucket=y-axis, $
*GRAPH_JS
"boxPlotProperties":
 {
  "hatWidth": "80%",
  "drawHatAsBox": false,
  "notch": {
         "width": "50%",
         "multiplier": 1.58
      }
    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



下図のように、ノッチの multiplier を 3.5 にすると、ノッチの幅が広くなります。



ボックスプロットの中央値線の外観制御

medianLine プロパティは、ボックスプロットの中央値線の外観を制御します。

構文　ボックスプロットの中央値線の外観制御

```
"boxPlotProperties": {
            "medianLine": {
                        "width": number,
      "color": "string",
      "dash": "string"
   }
},
```

説明

`"width": number`

中央値線の幅をピクセル単位で指定します。デフォルト値は 1 です。

`"color": "string"`

キーワードまたは数値指定の文字列で、中央値線の色を定義します。デフォルト値は "black" です。

`"dash": "string"`

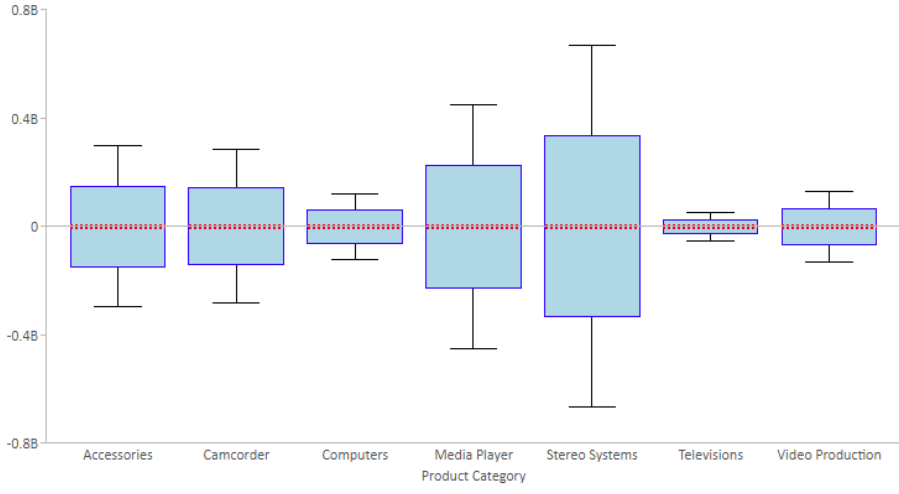
中央値線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　ボックスプロットの中央値線の制御

次のリクエストは、4 ピクセル幅の赤色の中央値線を生成し、破線長さを 2 ピクセル、破線間隔を 2 ピクセルに設定します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=diff2, bucket=min,$
type=data, column=diff1, bucket=lower,$
type=data, column=MDN.COGS_US, bucket=median,$
type=data, column=diff3, bucket=upper,$
type=data, column=diff4, bucket=max,$
*GRAPH_JS
"boxPlotProperties": {"drawHatAsBox": false, "medianLine": {
   "width": 4, "color": "red", "dash": "2 2"},
"series": [
 {"series": 0, "color": "lightblue", "border": {
    "width": 1, "color": "blue"}
   }]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ボックスプロットの接続線の外観制御

connectorLine プロパティは、ボックスプロットの接続線とハットの外観を制御します。接続線は、drawHatAsBox が false の場合にのみ表示されます。

構文　ボックスプロットの接続線の外観制御

```
"boxPlotProperties": {
   "connectorLine": {
      "width": number,
      "color": "string",
      "dash": "string"
   }
},
```

説明

`"width": number`

接続線の幅をピクセル単位で指定します。デフォルト値は 1 です。

`"color": "string"`

キーワードまたは数値指定の文字列で、接続線の色を定義します。デフォルト値は "black" です。

"dash": "string"

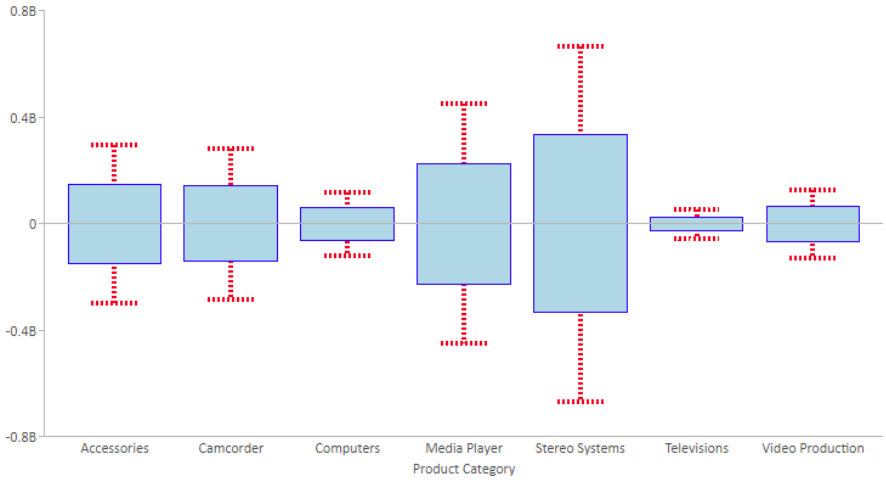
接続線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　ボックスプロットの接続線の制御

次のリクエストは、4 ピクセル幅の赤色の接続線を生成し、破線長さを 2 ピクセル、破線間隔を 2 ピクセルに設定します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=product_category, bucket=x-axis,$
type=data, column=diff2, bucket=min,$
type=data, column=diff1, bucket=lower,$
type=data, column=MDN.COGS_US, bucket=median,$
type=data, column=diff3, bucket=upper,$
type=data, column=diff4, bucket=max,$
*GRAPH_JS
"boxPlotProperties": {"drawHatAsBox": false, "connectorLine": {
   "width": 4, "color": "red", "dash": "2 2"},
"series": [
 {"series": 0, "color": "lightblue", "border": {
    "width": 1, "color": "blue"}
   }]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ボックスプロットの平均値マーカーの外観制御

ボックスプロットの平均値ポイントに表示するマーカーを構成することができます。

構文　　ボックスプロットの平均値マーカーの外観制御

```
"boxPlotProperties": {
    "meanMarker": {
      "visible": boolean,
      "color": "string",
      "border": {
      "width": number,
          "color": "string",
          "dash": "string"
            },
      "size": number,
      "shape": "string",
      "rotation": number,
      "tooltip": "string"
        }
    },
```

説明

`"meanMarker"`
ボックスプロットの平均値ポイントに描画するマーカーのプロパティを定義します。

`"visible": boolean`
平均値マーカーを描画するかどうかを定義します。有効な値には、次のものがあります。

- true - 平均値マーカーを描画します。
- false - 平均値マーカーを描画しません。これがデフォルト値です。

`"color": "string"`
平均値マーカーの色を定義する色指定の文字列です。デフォルト値は、"orange" です。

`"border"`
平均値マーカー境界線のプロパティを定義します。

`"width": number`
平均値マーカー境界線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`
平均値マーカー境界線の色を定義する色指定の文字列です。デフォルト値は undefined です。

`"dash": "string"`
平均値マーカー境界線の破線プロパティを定義する文字列です。破線のピクセルサイズ、破線間隔のピクセル数の順で入力します。デフォルト値は undefined です。

`"size": number`
平均値マーカーのサイズです (ピクセル単位)。デフォルト値は 8 です。

`"shape": "string"`
平均値マーカーの形状を定義します。標準的なマーカー形状はすべてサポートされます。デフォルト値は "circle" です。

`"rotation": number`
平均値マーカーの回転角度を定義する数値です。デフォルト値は 0 (ゼロ) です。

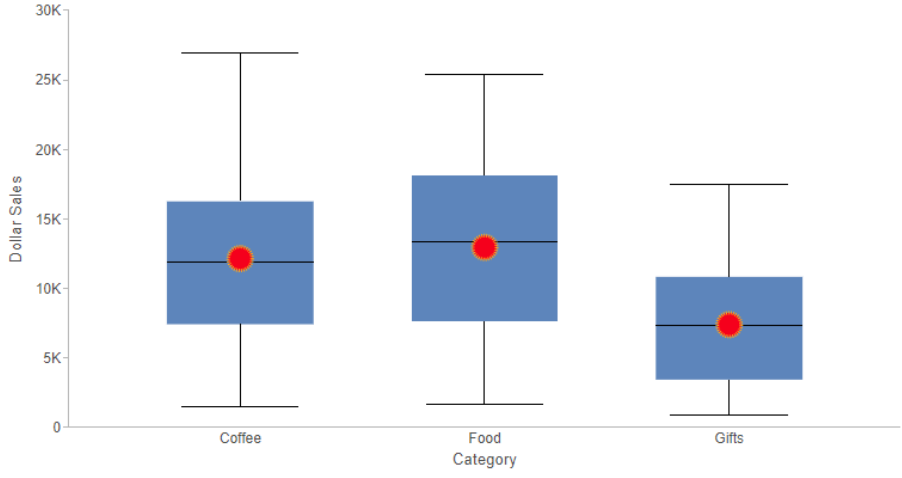
`"tooltip": "string"`
平均値マーカーのツールヒントを定義します。デフォルト値は undefined です。

例　統計ボックスプロットの平均値マーカーの外観制御

次のリクエストは、統計ボックスプロットを生成し、平均値マーカーを赤色の丸、境界線をオレンジ色の破線で設定します。

```
GRAPH FILE GGSALES
PRINT DOLLARS
BY CATEGORY
IF ST EQ 'NY'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STAT-BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=n1, bucket=x-axis, $
type=data, column=n2, bucket=y-axis, $
*GRAPH_JS
"boxPlotProperties": {
    "hatWidth": "80%",
    "drawHatAsBox": false,
    "meanMarker":
    {
      "visible": true,
      "color": "red",
      "border": {
         "width": 3,
         "color": "orange",
         "dash": "1 1"
              },
     "size": 20,
     "shape": "circle",
     "rotation": 0,
     "tooltip": undefined
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ブレットグラフのプロパティ (bulletProperties)

drawFirstValueAsBar プロパティは、ブレットグラフの全体的な外観を制御します。

構文　　ブレットグラフの外観の制御

```
"bulletProperties": {
   "drawFirstValueAsBar": boolean},
```

説明

`"drawFirstValueAsBar": boolean`

有効な値には、次のものがあります。

- true - 最初のデータ値を棒として描画します。これがデフォルト値です。
- false - 最初のデータ値をマーカーとして描画します。

例　ブレットグラフの外観の制御

次のリクエストは、drawFirstValueAsBar プロパティのデフォルト値 (true) を使用してブレットグラフを作成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MSRP_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Accessories' OR 'Computers' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET VAXIS 80
ON GRAPH SET LOOKGRAPH CUSTOM
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"chartType": "bullet",
"border": {"width": 1, "color": "navy"},
"bulletProperties": {"drawFirstValueAsBar": true},
"dataLabels": {"visible": true, "font": "6pt"},
"yaxis": {"colorBands": [
    {"start": 0, "stop": 80000000, "color": "silver"},
    {"start": 80000000, "stop": 150000000, "color": "lightgrey"},
    {"start": 150000000, "stop": 200000000, "color": "whitesmoke"}]},
"series": [
    {"series": "all", "showDataValues": true},
    {"series": 0, "group": 0, "color": "yellow",
        "marker": {"size": 15, "position": "bottom", "shape": "triangle"}},
    {"series": 0, "group": 1, "color": "red",
        "marker": {"size": 15, "position": "top", "shape": "triangle"}},
    {"series": 0, "group": 2, "color": "blue",
        "marker": {"size": 15, "position": "middle", "shape": "triangle"}}]
*END
ENDSTYLE
END
```

出力されたグラフでは、最初のデータ値が黄色の棒で表示され、その他のデータ値は赤色と青色の三角形で表示されています。

drawFirstValueAsBar の値を false に変更すると、黄色の棒が三角形で表示されたグラフが生成されます。

データグリッドのプロパティ (dataGridProperties)

データグリッドは、表形式レポートのように、データを表形式で表示します。グリッドは、複数の行、列、メジャーで構成することができます。データグリッドは、WebFOCUS TABLE FILE コマンドで生成される表形式レポートと異なり、PCHOLD FORMAT JSCHART を使用した GRAPH FILE リクエストで生成されます。

データグリッドは、グラフ属性構文を使用した場合のみ生成することができます。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

注意：HTML5 ビジュアライゼーションの [リスト] ビジュアルでは、テーマ (WebFOCUS スタイルシート) の変更はサポートされません。現時点では、[グリッド] ビジュアルで WebFOCUS TABLE フォーマット構文がサポートされないためです。

次のコードセグメントは、サポートされるプロパティおよびデフォルト値を示しています。

```
"dataGridProperties": {
    "altRowFill": "undefined",
    "cell": {
        "layout": {
             "width": {
                   "rule": "maxSize",
                   "value": 300},
    "cellPadding": {
      "left": 5,
      "top": 5,
      "right": 5,
      "bottom": 5},
            "height": {
                 "rule": "maxSize",
                 "value": 500}
            },

        "innerGridLines": {
            "horizontal": {
                "width": 1,
                "color": "#E0E0E0",
                "dash": ""},
            "vertical": {
                "width": 1,
                "color": "#EEEEEE",
                "dash": ""}},
        "labels": {
            "font": "10pt Sans-Serif",
            "color": "rgb(20, 20, 20)",
            "numberFormat": "auto",
            "align": "right",
            "valign": "top"},
        "tooltip": "undefined",
        "formatList": []
        },
```

```
"rowHeader": {
    "fill": "#FAFAFA",
    "border": {
        "width": 1,
        "color": "#D0D0D0",
        "dash": ""
        },
    "innerGridLines": {
        "horizontal": {
            "width": 1,
            "color": "#D8D8D8",
            "dash": ""},
        "vertical": {
            "width": 1,
            "color": "#E0E0E0",
            "dash": ""}
        },
    "dividerLine": {
        "width": 1,
        "color": "#D0D0D0",
        "dash": ""
        },
    "title": {
        "text": [],
        "font": "bold 10pt Sans-Serif",
        "color": "rgb(20, 20, 20)",
        "align": "left",
        "valign": "top",
        "tooltip": "undefined",
        "dividerLine": {
            "width": 1,
            "color": "#D0D0D0",
            "dash": ""}
        },
    "labels": {
        "font": "10pt Sans-Serif",
        "color": "rgb(20, 20, 20)",
        "align": "left",
        "valign": "top",
        "mergeMatching": true,
        "content": []}
        },
```

```
"colHeader": {
   "sorting": {
       "enabled": false},
   "resize": {
       "enabled": false},
   "fill": "#FAFAFA",
   "border": {
       "width": 1,
       "color": "#D0D0D0",
       "dash": ""},
   "innerGridLines": {
       "horizontal": {
           "width": 1,
           "color": "#E0E0E0",
           "dash": ""},
       "vertical": {
           "width": 1,
           "color": "#ECECEC",
           "dash": ""}
       },
   "dividerLine": {
       "width": 1,
       "color": "#D0D0D0",
       "dash": ""},
   "title": {
       "text": [],
       "font": "bold 10pt Sans-Serif",
       "color": "rgb(20, 20, 20)",
       "align": "right",
       "valign": "top",
       "tooltip": "undefined",
           "dividerLine": {
               "width": 1,
               "color": "#EEEEEE",
               "dash": ""}
       },
```

```
        "labels": {
            "font": "10pt Sans-Serif",
            "color": "rgb(20, 20, 20)",
            "align": "center",
            "valign": "top",
            "mergeMatching": true,
            "content": []},
        "lastLabels": {
            "font": "bold 10pt Sans-Serif",
            "color": "rgb(20, 20, 20)",
            "align": "right",
            "valign": "center"}
        },
    "columnTotals": {
        "visible": false,
        "calculation": "sum",
        "headerLabel": {
            "text": "TOTAL",
            "font": "bold 10pt Sans-Serif",
            "color": "rgb(20, 20, 20)",
            "align": "left",
            "valign": "center"},
        "cellLabel": {
            "font": "bold 10pt Sans-Serif",
            "color": "rgb(20, 20, 20)",
            "align": "right",
            "valign": "top"},
        "border": {
            "width": 1,
            "color": "#D0D0D0",
            "dash": ""},
        "dividerLine": {
            "width": 2,
            "color": "#D0D0D0",
            "dash": ""}
        },
```

```
        "scroll": {
            "enabled": true,
            "freezeRowHeader": true,
            "freezeColHeader": true,
            "size": 15,
            "color": "rgb(240, 240, 240)",
            "handle": {
                "color": "#C2C2C2",
                "hoverColor": "#A8A8A8",
                "border": {
                    "width": 0,
                    "color": "transparent",
                    "dash": ""}
                }
            }
        }
    }
```

データグリッド内の 1 行おきに異なる背景色の定義

dataGridProperties:altRowFill プロパティを使用して、グリッド内の連続する行に対して 1 行おきに異なる背景色を定義することができます。1 つの色を定義すると、その色とデフォルト設定の塗りつぶし色が交互に表示されます。

構文　データグリッド内の 1 行おきに異なる背景色の定義

```
dataGridProperties: {
 "altRowFill": "string"
                          }
```

説明

`"altRowFill": "string"`

データグリッド内の行を 1 行おきに異なる背景色で塗りつぶします。

次のいずれかの値です。

- undefined - これがデフォルト値です。
- 単一色を定義する文字列 (例、"red")
- 色の配列を定義する文字列 (例、"red","green")

例　データグリッド内の 1 行おきに異なる背景色の定義

次のリクエストは、データグリッド内の行の背景色として交互に異なる色の配列を定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {
    "altRowFill": ["pink", "yellow", "cyan"]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region	EMEA		North America		Oceania	
Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Pro
Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.00	$1,173.
Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92		
DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02		
Flat Panel TV	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.
Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.00	$325.
Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.00	$204.
Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.00	$283.
Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20		
Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68		
Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.
Speaker Kits	$144,389.00	$46,298.54	$163,451.00	$52,624.66		
Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.00	$110.
Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.00	$101.

データグリッドのセルのフォーマット設定

セル、グリッド線、ラベルのレイアウトのフォーマット設定が可能です。

構文　データグリッドのセルレイアウトのフォーマット設定

dataGridProperties:cell:layout プロパティで、グリッド内のデータセルの幅と高さを設定することができます。

```
"dataGridProperties": {
    "cell": {
        "layout":
    "width": width,
    "cellPadding": {
       "left": number,
       "top": number,
       "right": number,
       "bottom": number}
    "height": height
      }
          }
 }
```

説明

`"width": width`

グリッド内のセルの幅を定義します。定義しなかった場合、グリッドのサイズは利用可能なスペースをすべて使用するように設定されます。次のいずれかの値を指定します。

- ピクセル数
- 以下の形式のルールおよび値の組み合わせ

  `{"rule": "string", "value": number}`

  説明

  `"rule": "string"`

  次のいずれかです。

  - **"exactSize"** - *n* によって指定された正確なサイズに各セルのサイズを設定します。
  - **"minSize"** - *n* によって指定されたサイズ以上に各セルのサイズを決定します。
  - **"maxSize"** - *n* によって指定されたサイズ以下に各セルのサイズを決定します。

  デフォルト値は {"rule": "maxSize", "value": 300} です。

  `"value": number`

  ピクセル数です。

"cellPadding"
セル内のマージンを定義します。

"top": number
セルの上マージンをピクセル単位で指定します。デフォルト値は 5 です。

"left": number
セルの左マージンをピクセル単位で指定します。デフォルト値は 5 です。

"height": height

グリッド内のセルの高さを定義します。定義しなかった場合、グリッドのサイズは利用可能なスペースをすべて使用するように設定されます。次のいずれかの値を指定します。

- ピクセル数
- 以下の形式のルールおよび値の組み合わせ

  {"rule": "string", "value": number}

  説明

  "rule": "string"

  次のいずれかです。

  - **"exactSize"** - *n* によって指定された正確なサイズに各セルのサイズを設定します。
  - **"minSize"** - *n* によって指定されたサイズ以上に各セルのサイズを決定します。
  - **"maxSize"** - *n* によって指定されたサイズ以下に各セルのサイズを決定します。

  デフォルト値は {"rule": "maxSize", "value": 500} です。

  "value": number

  ピクセル数です。

例　データグリッドのセルレイアウトのフォーマット設定

次のリクエストは、セルの幅を 100 ピクセル、セルの高さを 20 ピクセルに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
dataGridProperties: {
cell:{
  layout:{
     width: 100,
     height: 20
           }
    }
                                 }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。列見出しのいくつかは、100 ピクセルの幅に収まらず、末尾が切り捨てられた状態で表示されています。これは、列の幅を大きくするか、minSize を 100 に設定 (width: {"rule": "minSize", "value":100}) することで解決できます。

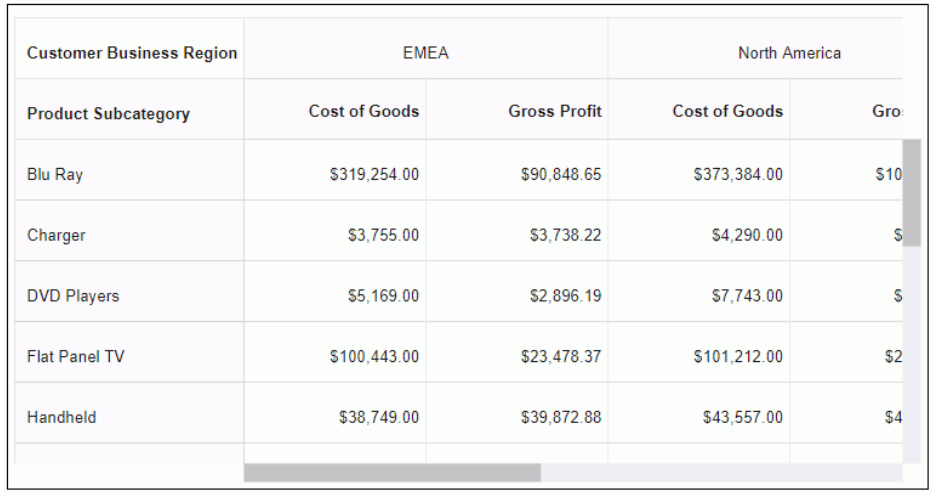
Customer Business Region	EMEA		North America		Oceania	
Product Subcategory	Cost of Goo...	Gross Profit	Cost of Goo...	Gross Profit	Cost of Goo...	Gross I
Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.00	$1,1
Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92		
DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02		
Flat Panel TV	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-
Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.00	$3
Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.00	$2
Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.00	$2
Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20		
Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68		
Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$3
Speaker Kits	$144,389.00	$46,298.54	$163,451.00	$52,624.66		
Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.00	$1
Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.00	$1
Universal Remote Controls	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,398.00	$6
Video Editing	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$3
iPod Docking Station	$43,842.00	$25,662.47	$55,383.00	$33,111.58	$891.00	$6

例　データグリッドのセルパディングのフォーマット設定

次のリクエストは、データグリッドを生成し、セルの最小幅を 150 ピクセル、各セルのパディングを上側 20 ピクセル、右側 5 ピクセル、下側 15 ピクセル、左側 10 ピクセルに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties":
 {
  "cell":{
     "layout":
             {
              "width": {"rule": "minSize", "value": 150},
              "cellPadding": {"top": 20,
                              "right": 5,
                              "bottom": 15,
                              "left": 10}
            }
         }
 }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region	EMEA		North America	
Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gro
Blu Ray	$319,254.00	$90,848.65	$373,384.00	$10
Charger	$3,755.00	$3,738.22	$4,290.00	$
DVD Players	$5,169.00	$2,896.19	$7,743.00	$
Flat Panel TV	$100,443.00	$23,478.37	$101,212.00	$2
Handheld	$38,749.00	$39,872.88	$43,557.00	$4

構文　データグリッドの内罫線のフォーマット設定

内罫線とは、グリッド内のセル間の区切り線、およびグリッドの右端 (横罫線の場合) または下端 (縦罫線の場合) の線のことです。

```
"dataGridProperties":
 "cell":{
  "innerGridLines": {
   "horizontal": {
    "width": number,
    "color": "string",
    "dash": "string"
             },
   "vertical": {
    "width": number,
    "color": "string",
    "dash": "string"
              }
                }
      }
                          }
```

説明

`"width": number`

横罫線の幅を定義するピクセル数です。デフォルト値は 1 です。

`"color": "string"`

横罫線の色を定義する文字列です。デフォルト値は "#E0E0E0" です。

"dash": "string"

横罫線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

破線を定義する文字列では、複数の破線スタイルを指定することができます。たとえば、「"2 4 4 2"」と指定すると、内罫線の破線スタイルが交互に入れ替わります。

"width": number

縦罫線の幅を定義するピクセル数です。デフォルト値は 1 です。

"color": "string"

縦罫線の色を定義する文字列です。デフォルト値は "#EEEEEE" です。

"dash": "string"

縦罫線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

破線を定義する文字列では、複数の破線スタイルを指定することができます。たとえば、「"2 4 4 2"」と指定すると、内罫線の破線スタイルが交互に入れ替わります。

例　データグリッドの内罫線のフォーマット設定

次のリクエストは、横罫線を赤色で幅 2 ポイント、縦罫線を緑色で幅 3 ポイントに指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"cell": {
    "innerGridLines": {
        "horizontal": {"width": 2, "color": "red", "dash": ""},
        "vertical": {"width": 3, "color": "green", "dash": ""}
    }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region	EMEA		North America		Oceania	
Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Pro
Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.00	$1,173.
Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92		
DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02		
Flat Panel TV	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.
Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.00	$325.
Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.00	$204.
Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.00	$283.
Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20		
Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68		
Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.
Speaker Kits	$144,389.00	$46,298.54	$163,451.00	$52,624.66		
Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.00	$110.
Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.00	$101.

構文　データグリッドのセルラベルのフォーマット設定

"dataGridProperties": "cell": "labels" プロパティは、グリッドセルの値およびツールヒントのフォーマットを指定します。デフォルト設定のツールヒントを上書きするには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックを使用する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

```
"dataGridProperties": {
"cell": {
"labels": {
   "font": "string",
   "color": "string",
   "numberFormat": "string",
   "align": "string",
   "valign": "string"
  },
  "tooltip": "string",
  "formatList": [
   {"row": number, "col": number, "repeat": number,
      "color": "string",
     "labels": {"font": "string", "color": "string"}} ...
   ]
 }
 }
 }
```

説明

`"font": "string"`

セルラベルのフォント属性を定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

セルラベルの色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

`"numberFormat": "string"`

セルラベルの数値フォーマットを指定します。デフォルト値は "auto" です。数値フォーマットについての詳細は、61 ページの 「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

`"align": "string"`

セルラベルの横方向の位置揃えを定義する文字列です。デフォルト値は "right" です。

`"valign": "string"`

セルラベルの縦方向の位置揃えを定義する文字列です。デフォルト値は "top" です。

"tooltip": "string"

セルラベルのカスタムツールヒントを定義します。カスタムツールヒントを定義するには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

"formatList"

特定のセルに、他のセルとは異なるフォーマットを設定する場合のフォーマットの配列です。formatList では、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの「 TIBCO WebFOCUS グラフ構文 」を参照してください。配列の各要素に対して以下のプロパティを使用することができます。

"row": number

行番号です。

"col": number

列番号です。

"repeat": number

増分値です。最初の行や列にフォーマットを適用後、増分値が追加され、後置のセルにも同じフォーマットが適用されます。

"color": "string"

指定したセルの塗りつぶし色を定義する文字列です。

"font": "string"

指定したセルのフォント属性を定義する文字列です。

"color": "string"

指定したセルのラベル色を定義する文字列です。

例　データグリッドのセルラベルのフォーマット設定

次のリクエストは、セルラベルの大部分を青色に設定します。ただし、行番号 2 (3 行目)、列番号 1 (2 列目) を赤色、12 ポイント、Arial フォントに、また塗りつぶしの色を水色に設定します。その後、行番号と列番号が 2 増えるごとに同じフォーマットが繰り返し適用されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"cell": {
    "labels": {"color": "blue"},
    "formatList": [
        {"row": 2, "col": 1, "repeat": 2, "color": "aqua",
            "labels": {"font": "12pt arial", "color": "red"}}]
    }}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。行番号および列番号はゼロ (0) から始まるため、行番号 2 は 3 行目となり、列番号 1 は 2 列目となります。

Customer Business Region	EMEA		North America		Oceania	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	143987	67490.99	171306	78629.1	1861	831.82
Camcorder	187000	93235.28	242967	106788.15	552	435.85
Computers	47616	30624.81	53329	34514.34	491	316.96
Media Player	328401	96506.47	386681	114870.53	4328	1275.82
Stereo Systems	361756	152933.93	412036	173093.52	1427	920.49
Televisions	100443	23478.37	101212	27670.05	365	-65.6
Video Production	78722	33419.72	89489	39758.23	589	328.99

データグリッドの行見出しのフォーマット設定

"dataGridProperties": "rowHeader" を使用して、行見出しの塗りつぶし色、境界線、内罫線、区切り線、タイトル、ラベルのフォーマットを設定することができます。

構文　データグリッドの行見出し塗りつぶし色の設定

```
"dataGridProperties": {
"rowHeader": {
  "fill": "string",
         }
                    }
```

説明

`"fill": "string"`

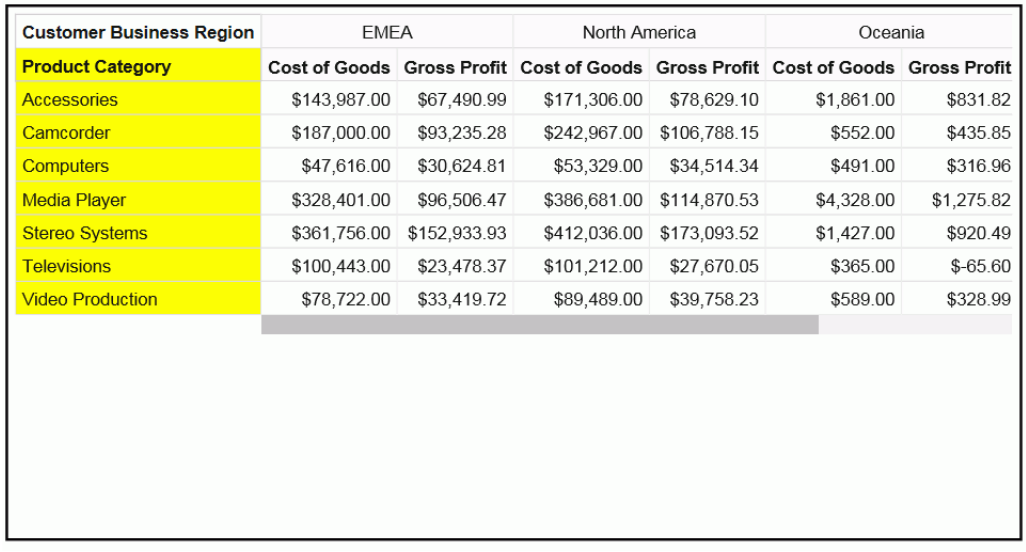
行見出しのセルの塗りつぶし色を定義する文字列です。デフォルト値は "#FAFAFA" です。

例　データグリッドの行見出し塗りつぶし色の設定

次のリクエストは、データグリッドの行見出し塗りつぶし色を黄色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"rowHeader": {"fill": "yellow"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region	EMEA		North America		Oceania	
Product Category	**Cost of Goods**	**Gross Profit**	**Cost of Goods**	**Gross Profit**	**Cost of Goods**	**Gross Profit**
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99

構文　　データグリッドの行見出し境界線のフォーマット設定

```
"dataGridProperties": {
    "rowHeader": {
     "border": {
   "width": number,
   "color": "string",
   "dash": "string"
  },
  }
}
```

説明

`"width": number`

行見出し境界線の幅をピクセル単位で定義します。デフォルト値は `1` です。

`"color": "string"`

行見出し境界線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

"dash": "string"

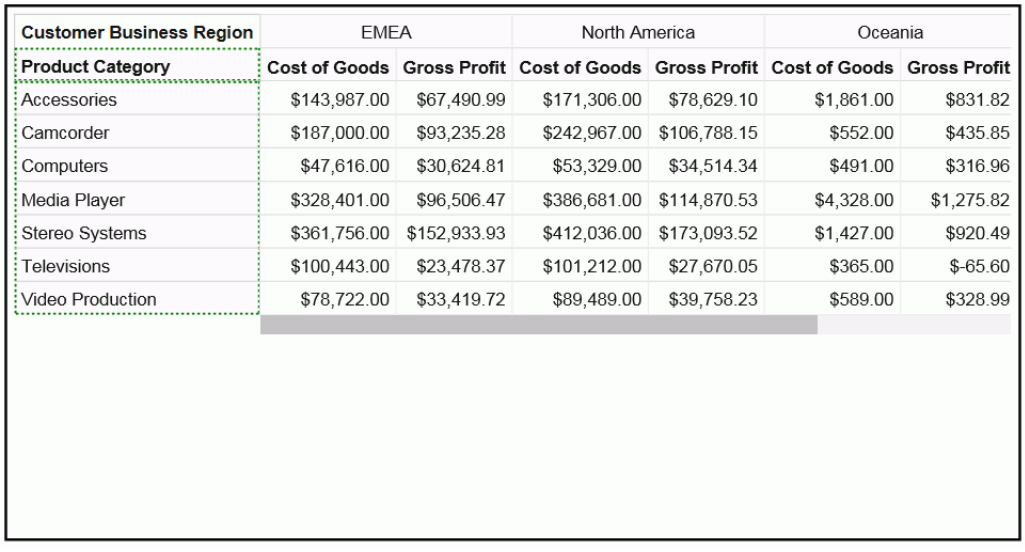
行見出し境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの行見出し境界線のフォーマット設定

次のリクエストは、行見出し境界線を幅 3 ピクセルの緑色に設定し、破線長さ 2 ピクセル、破線間隔 2 ピクセルで破線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"rowHeader": {
    "border": {"width": 3, "color": "green", "dash": "2 2"}
    }}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region	EMEA		North America		Oceania	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99

構文　　データグリッドの行見出し内罫線のフォーマット設定

行見出し内罫線とは、行見出し内の行列間の線のことです。

```
"dataGridProperties":{
"rowHeader": {
    "innerGridLines": {
   "horizontal": {
    "width": number,
    "color": "string",
    "dash": "string"
              },
   "vertical": {
    "width": number,
    "color": "string",
    "dash": "string"
            }
                    }
          }
                                        }
```

説明

`"width": number`

　行見出し内の横罫線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

　行見出し内の横罫線の色を定義する文字列です。デフォルト値は "#D8D8D8" です。

"dash": "string"

行見出し内の横罫線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"width": number

行見出し内の縦罫線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

行見出し内の縦罫線の色を定義する文字列です。デフォルト値は "#E0E0E0" です。

"dash": "string"

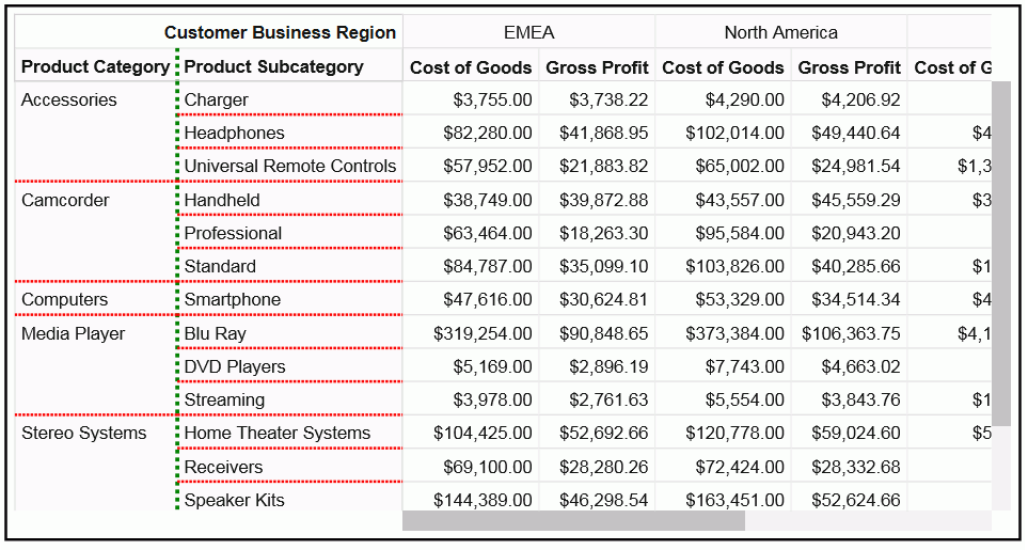
行見出し内の縦罫線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの行見出し内罫線のフォーマット設定

次のリクエストは、行見出し内の横罫線を赤色の幅 2 ピクセルに設定し、破線長さ 2 ピクセル、破線間隔 1 ピクセルで破線スタイルを設定します。行見出し内の縦罫線は緑色の幅 3 ピクセルに設定し、破線長さ 3 ピクセル、破線間隔 3 ピクセルで破線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"rowHeader": {
    "innerGridLines": {
        "horizontal": {"width": 2, "color": "red", "dash": "2 1"},
        "vertical": {"width": 3, "color": "green", "dash": "3 3"}
        }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region		EMEA		North America		
Product Category	**Product Subcategory**	**Cost of Goods**	**Gross Profit**	**Cost of Goods**	**Gross Profit**	**Cost of G**
Accessories	Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92	
	Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$4
	Universal Remote Controls	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,3
Camcorder	Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$3
	Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20	
	Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$1
Computers	Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$4
Media Player	Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,1
	DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02	
	Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$1
Stereo Systems	Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$5
	Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68	
	Speaker Kits	$144,389.00	$46,298.54	$163,451.00	$52,624.66	

構文　データグリッドの行見出し区切り線のフォーマット設定

データグリッドの行見出し区切り線とは、行見出しとグリッド本文のセルとの境目にある線のことです。

```
"dataGridProperties":{
"rowHeader": {
    "dividerLine": {
   "width": number,
   "color": "string",
   "dash": "string"
                },
          }
                    }
```

説明

`"width": number`

行見出し区切り線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

行見出し区切り線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

"dash": "string"

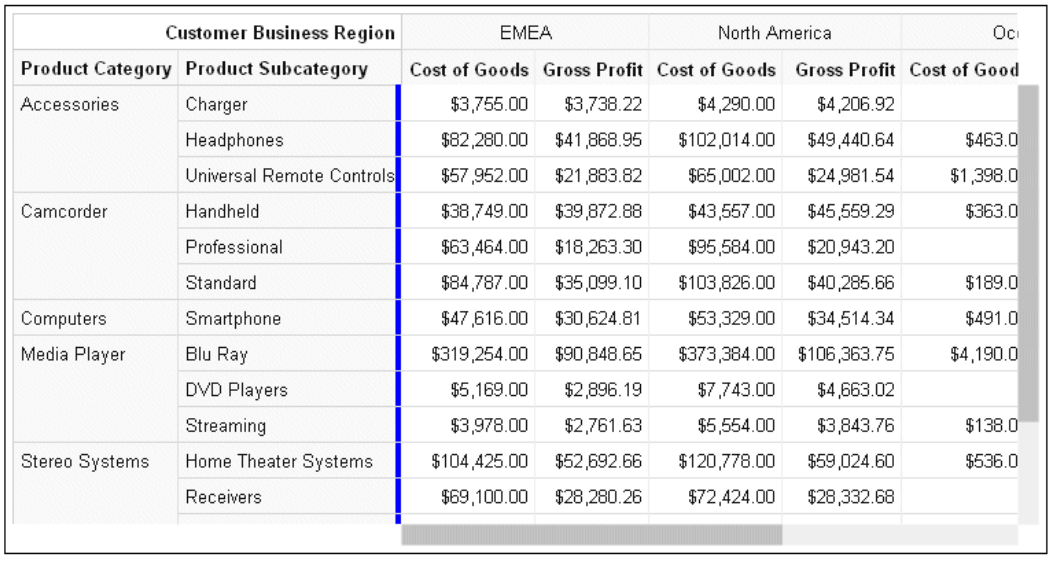
行見出し区切り線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの行見出し区切り線のフォーマット設定

次のリクエストは、行見出し区切り線を幅 8 ピクセルの青色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"rowHeader": {
    "dividerLine": {"width": 8, "color": "blue"}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region		EMEA		North America		Oc
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Good
Accessories	Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92	
	Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.0
	Universal Remote Controls	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,398.0
Camcorder	Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.0
	Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20	
	Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.0
Computers	Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.0
Media Player	Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.0
	DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02	
	Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.0
Stereo Systems	Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.0
	Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68	

構文　データグリッドの行見出しタイトルのフォーマット設定

デフォルト設定では、行見出しタイトルはマスターファイルまたはリクエスト内で定義されます。デフォルト設定のタイトルおよびツールヒントを上書きするには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

```
"dataGridProperties":{
"rowHeader": {
   "title": {
   "text": ["string",...],
   "font": "string",
   "color": "string",
   "align": "string",
   "valign": "string",
   "tooltip": "string",
   "dividerLine": {
    "width": number,
    "color": "string",
    "dash": "string"
   }
  },
    }
 }
}
```

説明

`"text": ["string",...]`

行見出しタイトルのテキストを定義します。単一文字列または文字列配列で定義します。ネストされた行見出しタイトルはサポートされません。デフォルト設定の行見出しタイトルは、リクエストの AS 名またはマスターファイルのフィールド名かタイトルのいずれかです。行見出しタイトルを変更するには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

`"font": "string"`

行見出しタイトルのフォント属性を定義する文字列です。デフォルト値は "bold 10pt Sans-Serif" です。

`"color": "string"`

行見出しタイトルの色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

`"align": "string"`

行見出しタイトルの横方向の位置揃えを定義する文字列です。デフォルト値は "left" です。

`"valign": "string"`

行見出しタイトルの縦方向の位置揃えを定義する文字列です。デフォルト値は "top" です。

`"tooltip": "string"`

行見出しタイトルのツールヒントを定義します。デフォルト値は undefined です。 行見出しタイトルを変更するには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。コールバック関数を使用したカスタムツールヒントの作成についての詳細は、61 ページの 「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

`"width": number`

行見出しタイトル区切り線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

行見出しタイトル区切り線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

```
"dash": "string"
```

行見出しタイトル区切り線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの行見出しタイトルのフォーマット設定

次のリクエストは、行見出しタイトルのフォーマットを設定します。2 つのタイトルのテキストを紺色の中央揃えに設定し、値を Category と Subcategory に設定します。行見出しタイトルの区切り線は水色、幅 5 ピクセルです。プロパティは、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックに追加する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"rowHeader": {
    "title": {
        "text": ["Category", "Subcategory"],
        "font": "bold 10pt Sans-Serif", "color": "navy",
        "align": "center", "valign": "top",
        "dividerLine": {"width": 5, "color": "aqua"}
        }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region		EMEA		North America		Ocea
Category	Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods
Accessories	Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92	
	Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.00
	Universal Remote Controls	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,398.00
Camcorder	Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.00
	Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20	
	Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.00
Computers	Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00
Media Player	Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.00
	DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02	
	Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.00
Stereo Systems	Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.00
	Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68	

構文　データグリッドの行見出しラベルのフォーマット設定

データグリッドの行見出しラベルは、row 属性カテゴリに割り当てられた各 BY フィールドのソートフィールド値になります。ラベルのフォント、色、横方向および縦方向の位置揃え、コンテンツのフォーマット設定が可能です。また、連続するラベルを同一のコンテンツに統合するか、別個に扱うかを指定することもできます。

```
"dataGridProperties":{
 "rowHeader": {
  "labels": {
    "font": "string",
    "color": "string",
    "align": "string",
    "valign": "string",
    "mergeMatching": boolean,
   "content": ["string",...]
  }
 }
}
```

説明

`"font": "string"`

行見出しラベルのフォント属性を定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

行見出しラベルの色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

`"align": "string"`

行見出しラベルの横方向の位置揃えを定義する文字列です。デフォルト値は "left" です。

`"valign": "string"`

行見出しラベルの縦方向の位置揃えを定義する文字列です。デフォルト値は "top" です。

`"mergeMatching": boolean`

各行の連続した重複ラベルを繰り返し表示するか統合するかを定義します。有効な値には、次のものがあります。

- true - 連続した行見出しラベルの重複を統合します。これがデフォルト値です。
- false - 列見出しラベルの重複を繰り返し表示します。

`"content": ["string",...]`

行見出しラベルを定義します。単一文字列、文字列配列、ネストされたオブジェクト配列 (オブジェクトのリーフが文字列配列の場合) のいずれかで定義することができます。

注意：ラベルを置き換える場合は、実際の行見出しラベル配列の構造に関係なく、定義したとおりのラベル配列が入力データに適用されます。

例　データグリッドの行見出しラベルのフォーマット設定

次のリクエストは、行見出しラベルを赤色にし、その他の行見出しラベルのプロパティにはすべてデフォルト設定を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"rowHeader": {
    "labels": {"color": "red"}
    }}
*END
ENDSTYLE
END
```

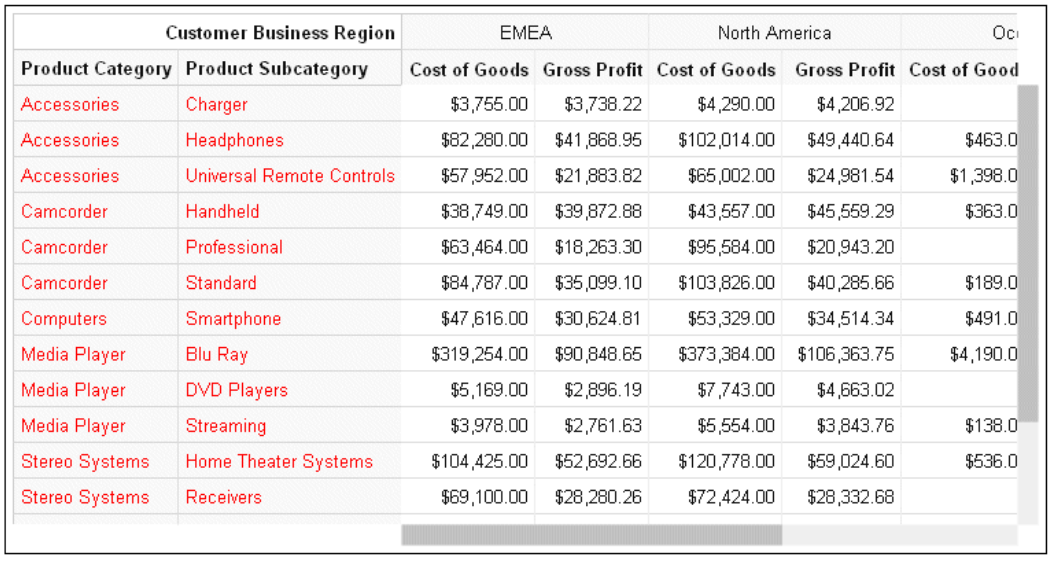
出力結果は次のとおりです。

Customer Business Region		EMEA		North America		Oc
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Good
Accessories	Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92	
	Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.0
	Universal Remote Controls	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,398.0
Camcorder	Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.0
	Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20	
	Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.0
Computers	Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.0
Media Player	Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.0
	DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02	
	Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.0
Stereo Systems	Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.0
	Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68	

次のリクエストでは、行見出しラベルのプロパティに "mergeMatching": false を追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"rowHeader": {
    "labels": {"color": "red", "mergeMatching": false}
    }}
*END
ENDSTYLE
END
```

下図のように、生成されたデータグリッドでは、行見出しラベルが各行に重複して表示されています。

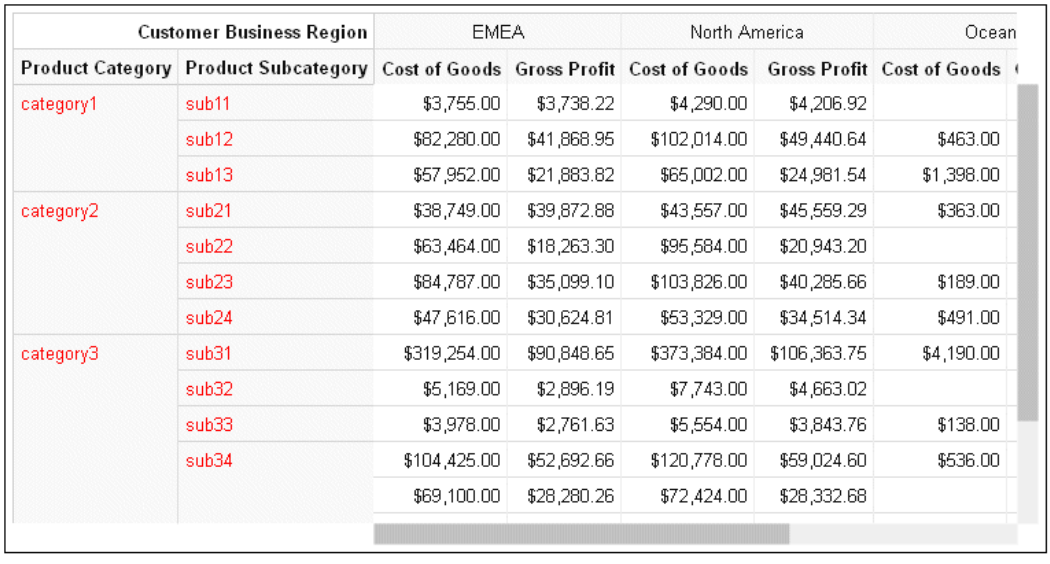


Customer Business Region		EMEA		North America		Oc
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Good
Accessories	Charger	$3,755.00	$3,738.22	$4,290.00	$4,206.92	
Accessories	Headphones	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.0
Accessories	Universal Remote Controls	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,398.0
Camcorder	Handheld	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.0
Camcorder	Professional	$63,464.00	$18,263.30	$95,584.00	$20,943.20	
Camcorder	Standard	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.0
Computers	Smartphone	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.0
Media Player	Blu Ray	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.0
Media Player	DVD Players	$5,169.00	$2,896.19	$7,743.00	$4,663.02	
Media Player	Streaming	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.0
Stereo Systems	Home Theater Systems	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.0
Stereo Systems	Receivers	$69,100.00	$28,280.26	$72,424.00	$28,332.68	

次のリクエストでは、オブジェクト内のラベル配列形式にカスタムラベルを指定します。リクエスト内の配列構造は、デフォルト設定で生成される実際のラベル配列構造とは一致しません。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"rowHeader": {"labels": {
    "font": "10pt Sans-Serif", "color": "red",
    "align": "left", "valign": "top",
    "mergeMatching": true,
    "content": [
        {"category1": ["sub11", "sub12", "sub13"]},
        {"category2": ["sub21", "sub22", "sub23", "sub24"]},
        {"category3": ["sub31", "sub32", "sub33", "sub34"]}
        ]
    }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。データは、リクエスト内の content オブジェクトに記述されたとおりのラベルに割り当てられます。



Customer Business Region		EMEA		North America		Ocean
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods
category1	sub11	$3,755.00	$3,738.22	$4,290.00	$4,206.92	
	sub12	$82,280.00	$41,868.95	$102,014.00	$49,440.64	$463.00
	sub13	$57,952.00	$21,883.82	$65,002.00	$24,981.54	$1,398.00
category2	sub21	$38,749.00	$39,872.88	$43,557.00	$45,559.29	$363.00
	sub22	$63,464.00	$18,263.30	$95,584.00	$20,943.20	
	sub23	$84,787.00	$35,099.10	$103,826.00	$40,285.66	$189.00
	sub24	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00
category3	sub31	$319,254.00	$90,848.65	$373,384.00	$106,363.75	$4,190.00
	sub32	$5,169.00	$2,896.19	$7,743.00	$4,663.02	
	sub33	$3,978.00	$2,761.63	$5,554.00	$3,843.76	$138.00
	sub34	$104,425.00	$52,692.66	$120,778.00	$59,024.60	$536.00
		$69,100.00	$28,280.26	$72,424.00	$28,332.68	

データグリッドの列見出しのフォーマット設定

"dataGridProperties": "colHeader" プロパティを使用して、列見出しのソート、サイズ設定、塗りつぶしの色、境界線、内罫線、区切り線、タイトル、ラベル、最終行ラベルのフォーマットを設定することができます。

列見出しの最終行には、データグリッドのメジャーフィールドの列タイトルが表示されます。この行に対しては、"dataGridProperties": "colHeader": "lastLabels" プロパティを使用して別個にフォーマットを設定する必要があります。

構文　データグリッドの対話的な並べ替えの設定

列見出しの並べ替えを有効にすると、各列見出しのセルの最終行に上下矢印が表示されます。これは、この列が元の順序で表示されていることを示します。矢印をクリックすると、列の順序が昇順、降順、元の順序でグラフの表示が切り替わります。矢印は、ソート順が昇順の場合は上向き、ソート順が降順の場合は下向きに変わります。矢印は、アクティブ列では太字で表示されます。

```
"dataGridProperties":{
"colHeader": {
  "sorting": {
   "enabled": boolean
          }
            }
                     }
```

説明

`"enabled": boolean`

列見出しの並べ替えを有効または無効にします。有効な値には、次のものがあります。

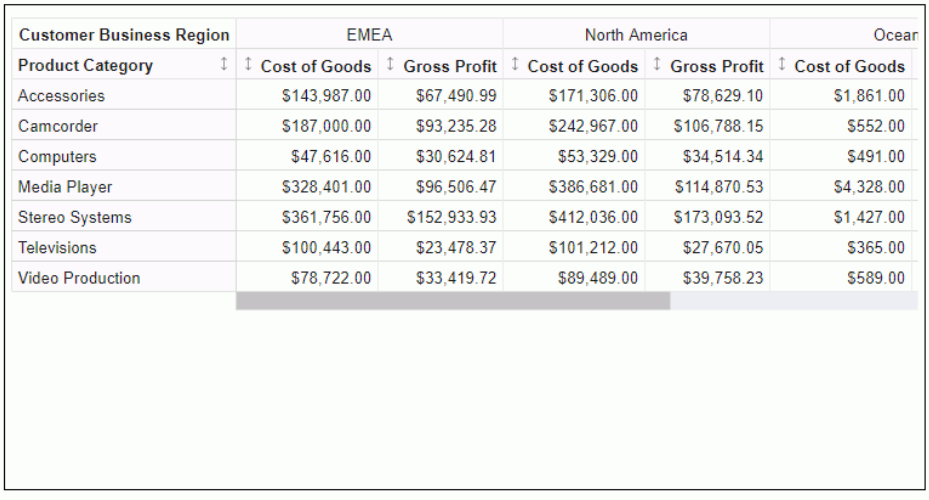
- true - 列に基づくグリッドの対話的な並べ替えを有効にします。
- false - 対話的なグリッドの並べ替えを無効にします。これがデフォルト値です。

例　データグリッドの対話的な並べ替えの有効化

次のリクエストは、データグリッドの対話的な並べ替えを有効にします。

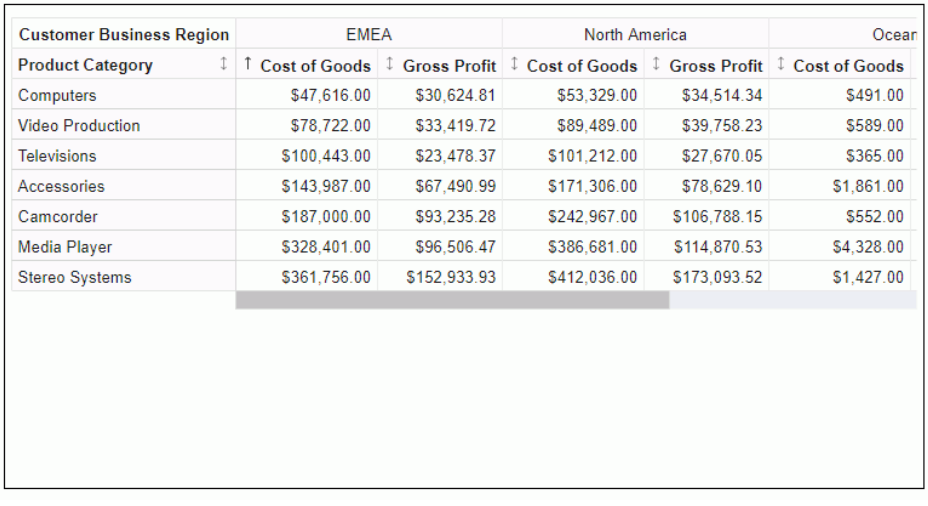
```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"colHeader": {"sorting": {"enabled":
true}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。列見出しの矢印をクリックすると、この列の値の順序が、昇順、降順、元の順序で切り替わります。



Customer Business Region	EMEA		North America		Ocear
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00

下図のデータグリッドでは、EMEA の売上原価列の値が昇順でソートされています。矢印が上向きに変わり、太字になっています。



Customer Business Region	EMEA		North America		Ocear
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00

構文　　データグリッドの対話的な列サイズ調整の有効化

データグリッドの列サイズ調整を有効化した場合、ユーザは列の右端をドラッグすることで列のサイズを調整することができます。

```
"dataGridProperties":{
"colHeader": {
  "resize": {
   "enabled": boolean
          }
            }
                     }
```

説明

`"enabled": boolean`

列のサイズ調整を有効または無効にします。有効な値には、次のものがあります。

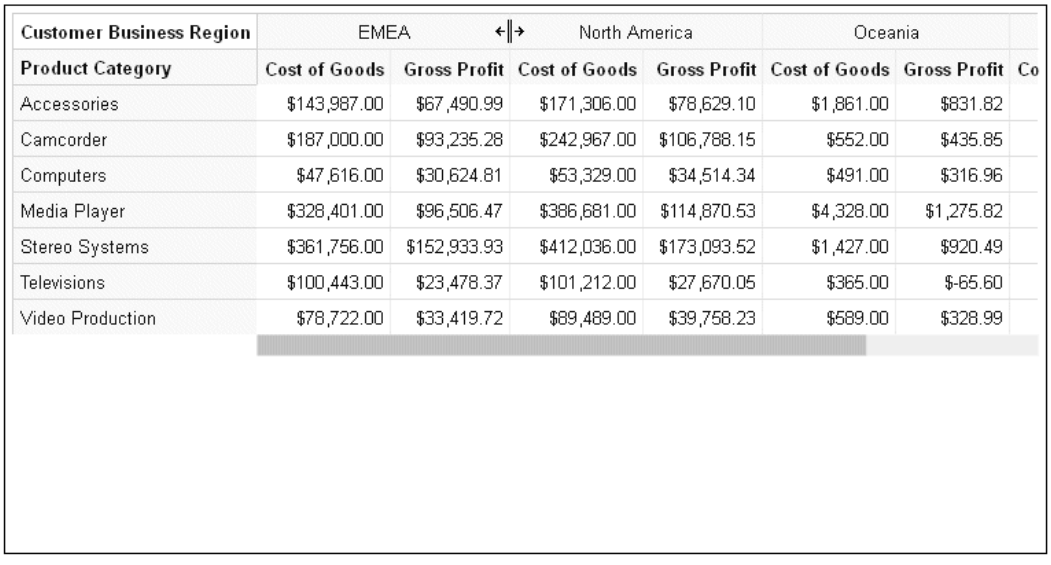
- true - グリッドの列サイズ調整を有効にします。
- false - グリッドの列サイズ調整を無効にします。これがデフォルト値です。

例　データグリッドの対話的な列サイズ調整の有効化

次のリクエストは、列のサイズ調整を有効にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"colHeader": {"resize": {"enabled":
true}}}
*END
ENDSTYLE
END
```

下図のように、列見出しの右端にマウスポインタを置くとサイズ調整ハンドルが表示され、ドラッグしながら列のサイズを調整することができます。



Customer Business Region	EMEA		North America		Oceania		
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Co
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82	
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85	
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96	
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82	
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49	
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60	
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99	

構文　　データグリッドの列見出し塗りつぶし色の設定

```
"dataGridProperties": {
"colHeader": {
  "fill": "string",
         }
                    }
```

説明

`"fill": "string"`

列見出しセルの塗りつぶし色を定義する文字列です。デフォルト値は "#FAFAFA" です。

例　データグリッドの列見出し塗りつぶし色の設定

次のリクエストは、データグリッドの列見出し塗りつぶし色を黄色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"colHeader": {"fill": "yellow"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region	EMEA		North America		Oceania		
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Co
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82	
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85	
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96	
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82	
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49	
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60	
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99	

構文　データグリッドの列見出し境界線のフォーマット設定

```
"dataGridProperties": {
"colHeader": {
  "border": {
   "width": number,
   "color": "string",
   "dash": "string"
  },
  }
}
```

説明

`"width": number`

列見出し境界線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

列見出し境界線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

`"dash": "string"`

列見出し境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの列見出し境界線のフォーマット設定

次のリクエストは、列見出し境界線を幅 3 ピクセルの緑色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"colHeader": {
    "border": {"width": 3, "color": "green"}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region	EMEA		North America		Oceania		South America	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82	$25,723.00	$9,722.49
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85	$22,686.00	$13,590.20
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96	$7,845.00	$5,024.35
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82	$60,183.00	$17,544.74
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49	$81,823.00	$32,403.95
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60	$25,800.00	$7,257.86
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99	$11,740.00	$5,132.90

構文　データグリッドの列見出し内罫線のフォーマット設定

列見出し内罫線とは、列見出し内の行列間の線のことです。

```
"dataGridProperties": {
"colHeader": {
    "innerGridLines": {
   "horizontal": {
    "width": number,
    "color": "string",
    "dash": "string"
              },
   "vertical": {
    "width": number,
    "color": "string",
    "dash": "string"
            }
                    }
          }
                                        }
```

説明

`"width": number`

　列見出し内の横罫線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

　列見出し内の横罫線の色を定義する文字列です。デフォルト値は "#E0E0E0" です。

"dash": "string"

列見出し内の横罫線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"width": number

列見出し内の縦罫線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

列見出し内の縦罫線の色を定義する文字列です。デフォルト値は "#ECECEC" です。

"dash": "string"

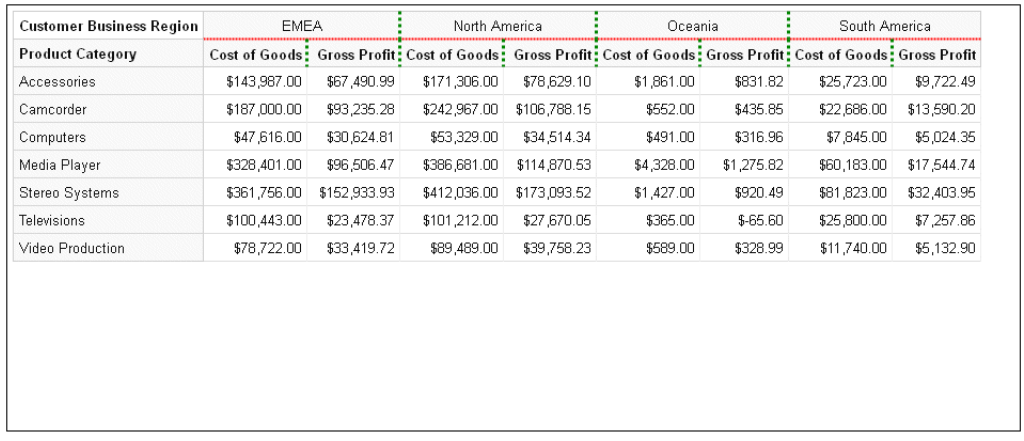
列見出し内の縦罫線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの列見出し内罫線のフォーマット設定

次のリクエストは、列見出し内の横罫線を赤色の幅 2 ピクセルに設定し、破線長さ 2 ピクセル、破線間隔 1 ピクセルで破線スタイルを設定します。行見出し内の縦罫線は緑色の幅 3 ピクセルに設定し、破線長さ 3 ピクセル、破線間隔 3 ピクセルで破線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET HAXIS 950
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"colHeader": {
    "innerGridLines": {
        "horizontal": {"width": 2, "color": "red", "dash": "2 1"},
        "vertical": {"width": 3, "color": "green", "dash": "3 3"}
}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region	EMEA		North America		Oceania		South America	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82	$25,723.00	$9,722.49
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85	$22,686.00	$13,590.20
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96	$7,845.00	$5,024.35
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82	$60,183.00	$17,544.74
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49	$81,823.00	$32,403.95
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60	$25,800.00	$7,257.86
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99	$11,740.00	$5,132.90

構文　データグリッドの列見出し区切り線のフォーマット設定

データグリッドの列見出し区切り線とは、列見出しとグリッド本文のセルとの境目にある線のことです。

```
"dataGridProperties": {
"colHeader": {
    "dividerLine": {
   "width": number,
   "color": "string",
   "dash": "string"
                },
          }
                    }
```

説明

`"width": number`

列見出し区切り線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

列見出し区切り線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

`"dash": "string"`

列見出し区切り線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの列見出し区切り線のフォーマット設定

次のリクエストは、列見出し区切り線を幅 8 ピクセルの青色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS
"dataGridProperties": {"colHeader": {
    "dividerLine": {"width": 8, "color": "blue"}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region	EMEA		North America		Oceania		
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Co
Accessories	$143,987.00	$67,490.99	$171,306.00	$78,629.10	$1,861.00	$831.82	
Camcorder	$187,000.00	$93,235.28	$242,967.00	$106,788.15	$552.00	$435.85	
Computers	$47,616.00	$30,624.81	$53,329.00	$34,514.34	$491.00	$316.96	
Media Player	$328,401.00	$96,506.47	$386,681.00	$114,870.53	$4,328.00	$1,275.82	
Stereo Systems	$361,756.00	$152,933.93	$412,036.00	$173,093.52	$1,427.00	$920.49	
Televisions	$100,443.00	$23,478.37	$101,212.00	$27,670.05	$365.00	$-65.60	
Video Production	$78,722.00	$33,419.72	$89,489.00	$39,758.23	$589.00	$328.99	

構文　データグリッドの列見出しタイトルのフォーマット設定

デフォルト設定では、列見出しタイトルはマスターファイルまたはリクエスト内で定義されます。デフォルト設定のタイトルおよびツールヒントを上書きするには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

```
"dataGridProperties":{
"colHeader": {
   "title": {
   "text": ["string",...],
   "font": "string",
   "color": "string",
   "align": "string",
   "valign": "string",
   "tooltip": "string",
   "dividerLine": {
    "width": number,
    "color": "string",
    "dash": "string"
   }
  },
    }
 }
}
```

説明

`"colHeader":`

データグリッドの列見出しのプロパティを定義します。

`"text": ["string",...]`

列見出しタイトルのテキストを定義します。単一文字列または文字列配列で定義します。ネストされた列見出しタイトルはサポートされません。デフォルト設定の列見出しタイトルは、マスターファイルのフィールド名かタイトル、またはリクエストの AS 名のいずれかです。行見出しタイトルを変更するには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

`"font": "string"`

列見出しタイトルのフォント属性を定義する文字列です。デフォルト値は "bold 10pt Sans-Serif" です。

`"color": "string"`

列見出しタイトルの色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

"align": "string"

列見出しタイトルの横方向の位置揃えを定義する文字列です。デフォルト値は "right" です。

"valign": "string"

列見出しタイトルの縦方向の位置揃えを定義する文字列です。デフォルト値は "top" です。

"tooltip": "string"

列見出しタイトルのツールヒントを定義します。デフォルト値は undefined です。列見出しのツールヒントを変更するには、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックにプロパティを追加する必要があります。詳細は、123 ページの「 TIBCO WebFOCUS グラフ構文 」を参照してください。コールバック関数を使用したカスタムツールヒントの作成についての詳細は、61 ページの「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

"dividerLine":

列見出しタイトル区切り線のプロパティを定義します。

"width": number

列見出しタイトル区切り線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

列見出しタイトル区切り線の色を定義する文字列です。デフォルト値は "#EEEEEE" です。

"dash": "string"

列見出しタイトル区切り線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの列見出しタイトルのフォーマット設定

次のリクエストは、列見出しタイトルのフォーマットを設定します。2 つのタイトルのテキストを紺色の中央揃えに設定し、値を Region と Country に設定します。列見出しタイトルの区切り線は水色の幅 5 ピクセルです。プロパティは、WebFOCUS スタイルシートの *GRAPH_JS_FINAL ブロックに追加する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
BY COUNTRY_NAME
WHERE BUSINESS_REGION EQ 'North America'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=country_name,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"colHeader": {
    "title": {
        "text": ["Region", "Country"],
        "font": "bold 10pt Sans-Serif", "color": "navy",
        "align": "center", "valign": "top",
        "dividerLine": {"width": 5, "color": "aqua"}
        }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Region / Country	North America					
	Canada		Mexico		United States	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$36,904.00	$17,936.26	$9,751.00	$3,733.69	$124,651.00	$56,959.15
Camcorder	$54,681.00	$22,860.20	$8,906.00	$5,828.75	$179,380.00	$78,099.20
Computers	$14,424.00	$9,537.58	$1,488.00	$1,039.94	$37,417.00	$23,936.82
Media Player	$90,911.00	$26,264.88	$19,987.00	$5,942.69	$275,783.00	$82,662.96
Stereo Systems	$89,325.00	$36,347.38	$19,224.00	$9,342.79	$303,487.00	$127,403.35
Televisions	$25,774.00	$6,764.75	$7,590.00	$2,267.43	$67,848.00	$18,637.87
Video Production	$22,097.00	$9,594.85	$4,420.00	$2,693.38	$62,972.00	$27,470.00

構文　データグリッドの列見出しラベルのフォーマット設定

データグリッドの列見出しラベルは、column 属性カテゴリに割り当てられた各 BY フィールドのソートフィールド値になります。ラベルのフォント、色、横方向および縦方向の位置揃え、コンテンツのフォーマット設定が可能です。また、連続するラベルを同一のコンテンツに統合するか、別個に扱うかを指定することができます。

```
"dataGridProperties":{
 "colHeader": {
  "labels": {
    "font": "string",
    "color": "string",
    "align": "string",
    "valign": "string",
    "mergeMatching": boolean,
   "content": ["string",...]
  }
 }
}
```

説明

`"font": "string"`

列見出しラベルのフォント属性を定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

列見出しラベルの色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

`"align": "string"`

列見出しラベルの横方向の位置揃えを定義する文字列です。デフォルト値は "center" です。

`"valign": "string"`

列見出しラベルの縦方向の位置揃えを定義する文字列です。デフォルト値は "top" です。

`"mergeMatching": boolean`

各列の連続した重複ラベルを繰り返し表示するか統合するかを定義します。有効な値には、次のものがあります。

- true - 列見出しラベルの重複を統合します。これがデフォルト値です。
- false - 列見出しラベルの重複を繰り返し表示します。

`"content": ["string",...]`

列見出しラベルを定義します。単一文字列、文字列配列、ネストされたオブジェクト配列 (オブジェクトのリーフが文字列配列の場合) のいずれかで定義することができます。配列では、列見出しの最終行 (メジャーフィールドに列タイトルを含む行) を考慮する必要があります。

注意：ラベルを置き換える場合は、実際の列見出しラベル配列の構造に関係なく、定義したとおりのラベル配列が入力データに適用されます。

例　データグリッドの列見出しラベルのフォーマット設定

次のリクエストは、列見出しラベルを赤色にし、その他の列見出しラベルのプロパティにはすべてデフォルト設定を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
BY COUNTRY_NAME
WHERE BUSINESS_REGION EQ 'North America'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=country_name,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"colHeader": {"labels": {"color": "red"}}}

*END
ENDSTYLE
END
```

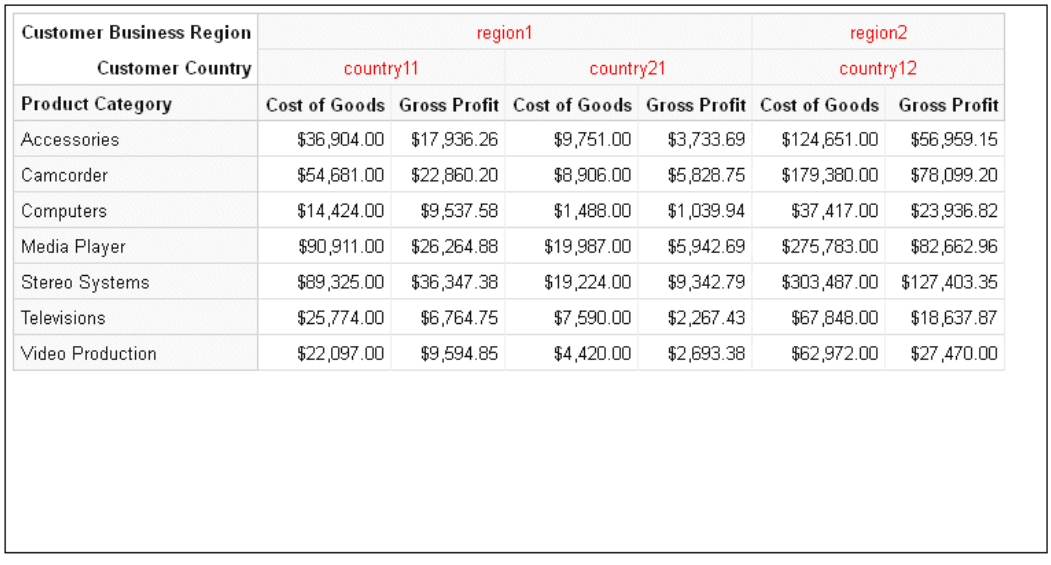
出力結果は次のとおりです。

Customer Business Region	North America					
Customer Country	Canada		Mexico		United States	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$36,904.00	$17,936.26	$9,751.00	$3,733.69	$124,651.00	$56,959.15
Camcorder	$54,681.00	$22,860.20	$8,906.00	$5,828.75	$179,380.00	$78,099.20
Computers	$14,424.00	$9,537.58	$1,488.00	$1,039.94	$37,417.00	$23,936.82
Media Player	$90,911.00	$26,264.88	$19,987.00	$5,942.69	$275,783.00	$82,662.96
Stereo Systems	$89,325.00	$36,347.38	$19,224.00	$9,342.79	$303,487.00	$127,403.35
Televisions	$25,774.00	$6,764.75	$7,590.00	$2,267.43	$67,848.00	$18,637.87
Video Production	$22,097.00	$9,594.85	$4,420.00	$2,693.38	$62,972.00	$27,470.00

次のリクエストでは、オブジェクト内のラベル配列形式にカスタムラベルを指定します。配列とオブジェクトを正確に記述するには、列見出しの最終行を考慮する必要があります。ここにはメジャーフィールドの列タイトルが含まれます。この例では、リクエスト内の配列構造は、デフォルト設定の実際のラベル配列構造とは一致しません。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
BY COUNTRY_NAME
WHERE BUSINESS_REGION EQ 'North America'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=country_name,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"colHeader": {
    "labels": {"color": "red",
        "content": [{"region1": [{"country11": ["Cost of Goods","Gross
Profit"]},
                {"country21": ["Cost of Goods", "Gross Profit"]}],
            "region2": [{"country12": ["Cost of Goods", "Gross Profit"]}]
        }]
    }}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。データは、リクエスト内の content オブジェクトに記述されたとおりのラベルに割り当てられます。



Customer Business Region	region1				region2	
Customer Country	country11		country21		country12	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$36,904.00	$17,936.26	$9,751.00	$3,733.69	$124,651.00	$56,959.15
Camcorder	$54,681.00	$22,860.20	$8,906.00	$5,828.75	$179,380.00	$78,099.20
Computers	$14,424.00	$9,537.58	$1,488.00	$1,039.94	$37,417.00	$23,936.82
Media Player	$90,911.00	$26,264.88	$19,987.00	$5,942.69	$275,783.00	$82,662.96
Stereo Systems	$89,325.00	$36,347.38	$19,224.00	$9,342.79	$303,487.00	$127,403.35
Televisions	$25,774.00	$6,764.75	$7,590.00	$2,267.43	$67,848.00	$18,637.87
Video Production	$22,097.00	$9,594.85	$4,420.00	$2,693.38	$62,972.00	$27,470.00

構文　データグリッドの列見出しラベル最終行のフォーマット設定

列見出しラベルの最終行は、データグリッドのメジャーフィールドの列タイトルに対応します。"dataGridProperties": "colHeader": "lastLabels" プロパティを使用して、これらのラベルにフォーマットを設定することができます。

```
"dataGridProperties": {
 "colHeader": {
  "lastLabels": {
    "font": "string",
    "color": "string",
    "align": "string",
    "valign": "string",
        }
           }
               }
```

説明

`"font": "string"`

列見出しラベル最終行のフォント属性を定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

"color": "string"

列見出しラベル最終行の色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

"align": "string"

列見出しラベル最終行の横方向の位置揃えを定義する文字列です。デフォルト値は "right" です。

"valign": "string"

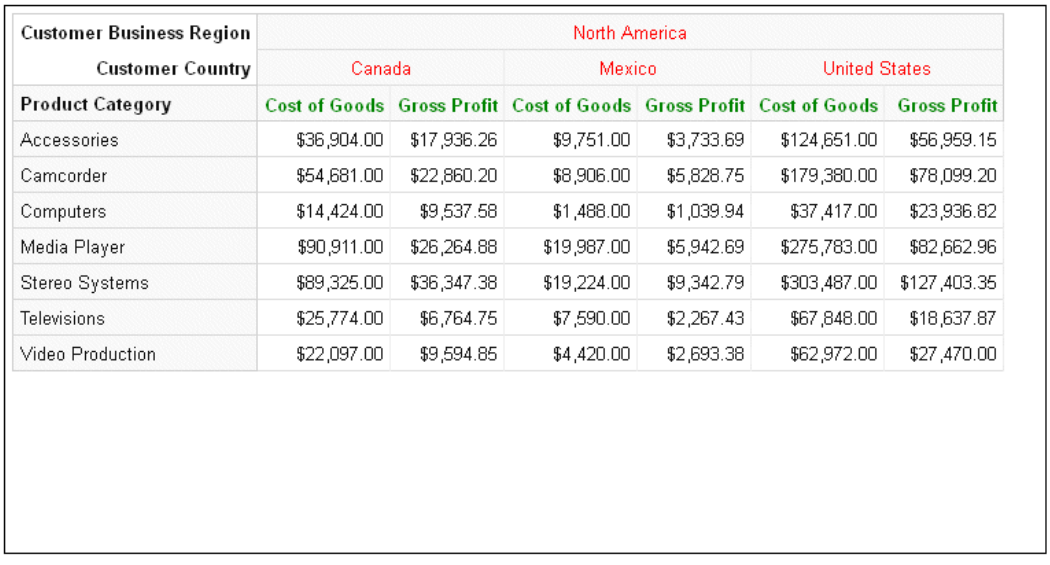
列見出しラベル最終行の縦方向の位置揃えを定義する文字列です。デフォルト値は "center" です。

例　データグリッドの列見出しラベル最終行のフォーマット設定

次のリクエストは、列見出しラベル最終行を緑色、その他の行は赤色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
BY COUNTRY_NAME
WHERE BUSINESS_REGION EQ 'North America'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=country_name,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"colHeader": {
    "labels": {"color": "red"},
    "lastLabels": {"color": "green"}
    }}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



Customer Business Region	North America					
Customer Country	Canada		Mexico		United States	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$36,904.00	$17,936.26	$9,751.00	$3,733.69	$124,651.00	$56,959.15
Camcorder	$54,681.00	$22,860.20	$8,906.00	$5,828.75	$179,380.00	$78,099.20
Computers	$14,424.00	$9,537.58	$1,488.00	$1,039.94	$37,417.00	$23,936.82
Media Player	$90,911.00	$26,264.88	$19,987.00	$5,942.69	$275,783.00	$82,662.96
Stereo Systems	$89,325.00	$36,347.38	$19,224.00	$9,342.79	$303,487.00	$127,403.35
Televisions	$25,774.00	$6,764.75	$7,590.00	$2,267.43	$67,848.00	$18,637.87
Video Production	$22,097.00	$9,594.85	$4,420.00	$2,693.38	$62,972.00	$27,470.00

データグリッドの列合計の生成とフォーマット設定

"dataGridProperties": "columnTotals" を使用して、データグリッド上に列合計を生成し、そのプロパティ、演算、タイトル、区切り線のフォーマットを設定することができます。

構文　　データグリッドの列合計の生成とフォーマット設定

```
"dataGridProperties":
{
"columnTotals": {
  "visible": boolean,
  "calculation": "string",
  "headerLabel": {
   "text": "string",
   "font": "string",
   "color": "string",
   "align": "string",
   "valign": "string"
  },
  "cellLabel": {
   "font": "string",
   "color": "string",
   "align": "string",
   "valign": "string"
  },
  "border": {
   "width": number,
   "color": "string",
   "dash": "string"
  },
  "dividerLine": {
   "width": number,
   "color": "string",
   "dash": "string"
  }
 }
}
```

説明

`"visible": boolean`

データグリッド上の列合計の表示を制御します。有効な値には、次のものがあります。

- true - 列合計を表示します。
- false - 列合計を表示しません。これがデフォルト値です。

`"calculation": "string"`

列合計の演算方法を制御します。有効な値は、次のとおりです。

- "sum" - 各列の値を加算して列合計を生成します。これがデフォルト値です。
- "min" - 各列の最小値を計算して列合計を生成します。
- "max" - 各列の最大値を計算して列合計を生成します。

- "mean" - 各列の値を平均して列合計を生成します。

`"headerLabel":`
列合計の行に適用する見出しラベルのプロパティを定義します。

`"text": "string"`

列合計の行に適用する見出しラベルを定義する文字列です。デフォルト値は "TOTAL" です。

`"font": "string"`

列合計の見出しラベルのフォント属性を定義する文字列です。デフォルト値は "bold 10pt Sans-Serif" です。

`"color": "string"`

列合計の見出しラベルの色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

`"align": "string"`

列合計見出しラベルの横方向の位置揃えを定義する文字列です。デフォルト値は "left" です。

`"valign": "string"`

列合計見出しラベルの縦方向の位置揃えを定義する文字列です。デフォルト値は "center" です。

`"cellLabel":`
列合計行のセルのプロパティを定義します。

`"font": "string"`

列合計行のセルの値に適用するフォント属性を定義する文字列です。デフォルト値は "bold 10pt Sans-Serif" です。

`"color": "string"`

列合計行のセルの値に適用する色を定義する文字列です。デフォルト値は "rgb(20, 20, 20)" です。

`"align": "string"`

列合計行のセルの値に適用する横方向の位置揃えを定義する文字列です。デフォルト値は "right" です。

"valign": "string"

列合計行のセルの値に適用する縦方向の位置揃えを定義する文字列です。デフォルト値は "top" です。

"border":

列合計行下部および両端を囲む境界線のプロパティを定義します。

"width": number

列合計行下部および両端を囲む境界線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

列合計行下部および両端を囲む境界線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

"dash": "string"

列合計行下部および両端を囲む境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"dividerLine"

列合計行上部の区切り線のプロパティを定義します。

"width": number

列合計行上部の区切り線の幅をピクセル単位で定義します。デフォルト値は 2 です。

"color": "string"

列合計行上部の区切り線の色を定義する文字列です。デフォルト値は "#D0D0D0" です。

"dash": "string"

列合計行上部の区切り線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドの列合計行の生成とフォーマット設定

次のリクエストは、演算方法に "mean" を使用して列合計行を生成します。列合計行の見出しラベルは Average、色は赤です。列合計行のセルの計算値は緑色です。区切り線は青色です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
BY COUNTRY_NAME
WHERE BUSINESS_REGION EQ 'North America'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET HAXIS 800
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=country_name,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {"columnTotals": {
    "visible": true, "calculation": "mean",
    "headerLabel": {"text": "Average", "color": "red"},
    "cellLabel": {"color": "green"},
    "dividerLine": {"width": 2, "color": "blue"}
}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

Customer Business Region	North America					
Customer Country	Canada		Mexico		United States	
Product Category	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit
Accessories	$36,904.00	$17,936.26	$9,751.00	$3,733.69	$124,651.00	$56,959.15
Camcorder	$54,681.00	$22,860.20	$8,906.00	$5,828.75	$179,380.00	$78,099.20
Computers	$14,424.00	$9,537.58	$1,488.00	$1,039.94	$37,417.00	$23,936.82
Media Player	$90,911.00	$26,264.88	$19,987.00	$5,942.69	$275,783.00	$82,662.96
Stereo Systems	$89,325.00	$36,347.38	$19,224.00	$9,342.79	$303,487.00	$127,403.35
Televisions	$25,774.00	$6,764.75	$7,590.00	$2,267.43	$67,848.00	$18,637.87
Video Production	$22,097.00	$9,594.85	$4,420.00	$2,693.38	$62,972.00	$27,470.00
Average	$47,730.86	$18,472.27	$10,195.14	$4,406.95	$150,219.71	$59,309.91

データグリッドのスクロールバーの生成とフォーマット設定

デフォルト設定では、グリッド全体が描画領域に入りきらない場合に、データグリッド上にスクロールバーが生成されます。"dataGridProperties": "scroll" プロパティを使用して、スクロールバーの無効化、スクロールバーのフォーマット設定、および列見出しと行見出しの固定を行うことができます。

構文　データグリッドのスクロールバーの生成とフォーマット設定

```
"dataGridProperties": {
    "scroll": {
        "enabled": boolean,
  "freezeRowHeader": boolean,
  "freezeColHeader": boolean,
  "size": number,
  "color": "string",
  "handle": {
   "color": "string",
   "hoverColor": "string",
   "border": {
    "width": number,
    "color": "string",
    "dash": "string"
          }
        }
      }
            }
```

説明

`"enabled": boolean`

データグリッド上のスクロールバーを有効または無効にします。有効な値には、次のものがあります。

- true - スクロールバーを有効にします。これがデフォルト値です。
- false - スクロールバーを無効にします。

`"freezeRowHeader": boolean`

データグリッド上で行見出しを固定するかスクロールするかを定義します。有効な値には、次のものがあります。

- true - 行見出しを固定します。これがデフォルト値です。
- false - 行見出しをスクロールします。

"freezeColHeader": boolean
データグリッド上で列見出しを固定するかスクロールするかを定義します。有効な値には、次のものがあります。

- true - 列見出しを固定します。これがデフォルト値です。
- false - 列見出しをスクロールします。

"size": number

スクロールバーの高さをピクセル単位で定義します。デフォルト値は 15 です。

"color": "string"

スクロールバーの色を定義する文字列です。デフォルト値は "rgb(240, 240, 240)" です。

"handle":
スクロールバーハンドルのプロパティを定義します。

"color": "string"

スクロールバーハンドル (データグリッドをスクロールするためにドラッグする部分) の色を定義する文字列です。デフォルト値は "#C2C2C2" です。

"hoverColor": "string"

マウスポインタを置いたときのスクロールバーの色を定義する文字列です。デフォルト値は "#A8A8A8" です。

"border":
スクロールバーハンドルの境界線のプロパティを定義します。

"width": number

スクロールバーハンドルの境界線の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です。

"color": "string"

スクロールバーハンドルの境界線の色を定義する文字列です。デフォルト値は "transparent" です。

"dash": "string"

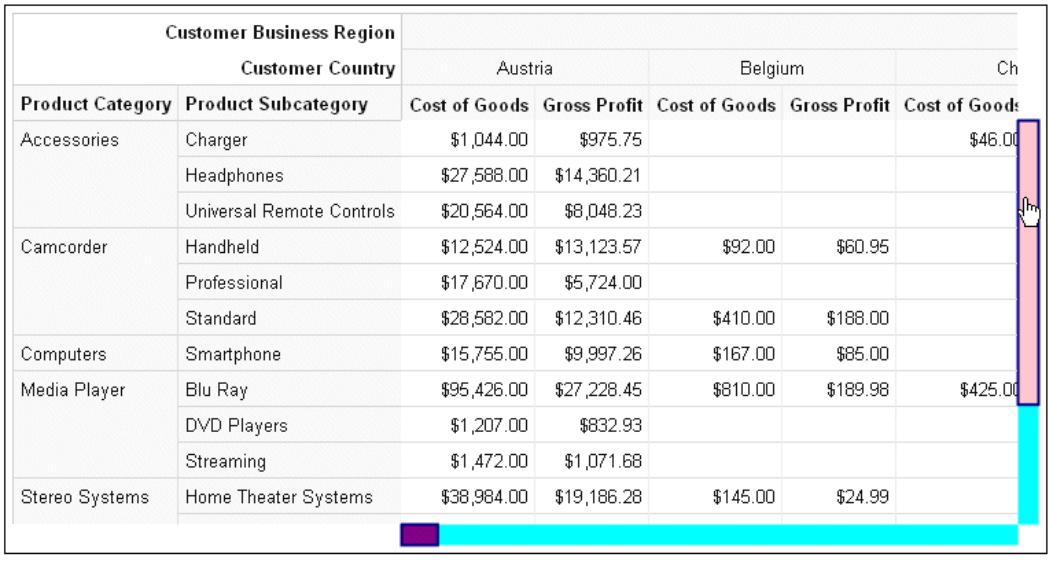
スクロールバーハンドルの境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　データグリッドのスクロールバーのフォーマット設定

次のリクエストは、スクロールバーの色を水色、ハンドルを紫色、境界線を幅 2 ピクセルの紺色、マウスポインタを置いたときの色をピンク色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
BY BUSINESS_REGION
BY COUNTRY_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH DATAGRID
ON GRAPH SET STYLE *
type=data, column=product_category,  bucket=row, $
type=data, column=product_subcateg,  bucket=row, $
type=data, column=business_region,   bucket=column, $
type=data, column=country_name,   bucket=column, $
type=data, column=cogs_us,    bucket=measure, $
type=data, column=gross_profit_us, bucket=measure, $
*GRAPH_JS_FINAL
"dataGridProperties": {
    "scroll": {"color": "aqua",
        "handle": {"color": "purple", "hoverColor": "pink",
            "border": {"width": 2, "color": "navy"}}
        }}
*END
ENDSTYLE
END
```

下図は、出力結果を示しています。ここでは、マウスポインタが縦スクロールバーハンドルの上に置かれています。



Customer Business Region						
Customer Country		Austria		Belgium		Ch
Product Category	Product Subcategory	Cost of Goods	Gross Profit	Cost of Goods	Gross Profit	Cost of Goods
Accessories	Charger	$1,044.00	$975.75			$46.00
	Headphones	$27,588.00	$14,360.21			
	Universal Remote Controls	$20,564.00	$8,048.23			
Camcorder	Handheld	$12,524.00	$13,123.57	$92.00	$60.95	
	Professional	$17,670.00	$5,724.00			
	Standard	$28,582.00	$12,310.46	$410.00	$188.00	
Computers	Smartphone	$15,755.00	$9,997.26	$167.00	$85.00	
Media Player	Blu Ray	$95,426.00	$27,228.45	$810.00	$189.98	$425.00
	DVD Players	$1,207.00	$832.93			
	Streaming	$1,472.00	$1,071.68			
Stereo Systems	Home Theater Systems	$38,984.00	$19,186.28	$145.00	$24.99	

じょうごグラフのプロパティ (funnelProperties)

じょうごグラフは基本的には円グラフですが、一度に表示するデータグループは 1 つです。グループに属する各シリーズは、じょうご内に積み重ねられ、最初のシリーズが最上部に、最後のシリーズが最下部に表示されます。じょうごグラフでは、表示フィールドはグループとして機能し、ソートフィールド値はシリーズとして機能します。

ここでは、じょうごグラフの全体的なレイアウトを制御するプロパティについて説明します。

次のコードセグメントは、デフォルト値を示しています。

```
"funnelProperties": {
    "topWidth": "100%",
    "baseWidth": "20%",
    "riserGap": 0,
    "groupLabel": {
        "visible": false,
        "font": "10pt Sans-Serif",
        "color": "black"
        }
    }
```

じょうごセグメントの色はシリーズ固有のプロパティで制御します。

じょうご上端幅の制御

topWidth プロパティは、じょうごの上端幅を制御します。

構文　じょうご上端幅の制御

```
"funnelProperties": {
    "topWidth": width}
```

説明

`"topWidth": width`

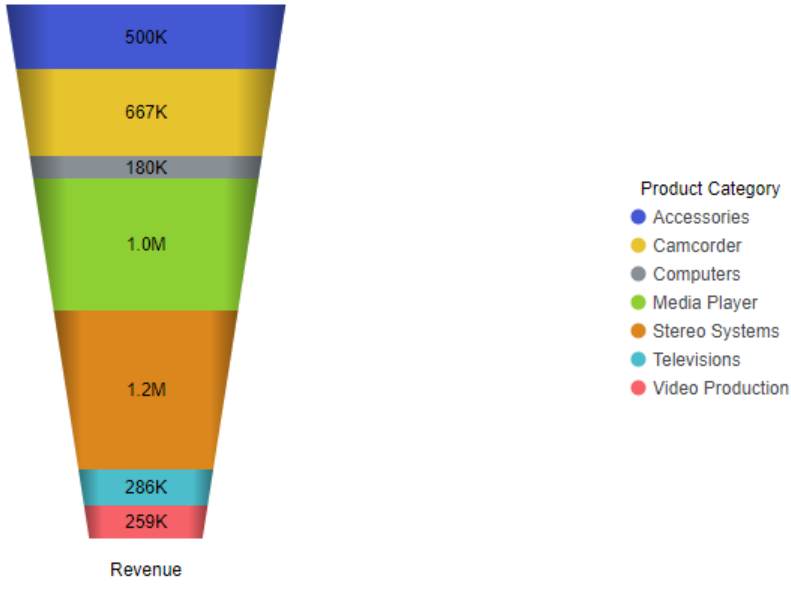
幅を定義する数値をピクセル単位で指定するか、フレーム内の利用可能領域に対する割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます ("0%" から "100%")。デフォルト値は "100%" です。

例　じょうご上端幅の制御

次のリクエストは、じょうごグラフを生成し、上端幅を 50% に縮小します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"legend": {"visible": true},
"funnelProperties": {"topWidth": "50%"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



じょうご下端幅の制御

baseWidth プロパティは、じょうごの下端幅を制御します。

構文　じょうご下端幅の制御

```
"funnelProperties": {
   "baseWidth": width}
```

説明

`"baseWidth": width`

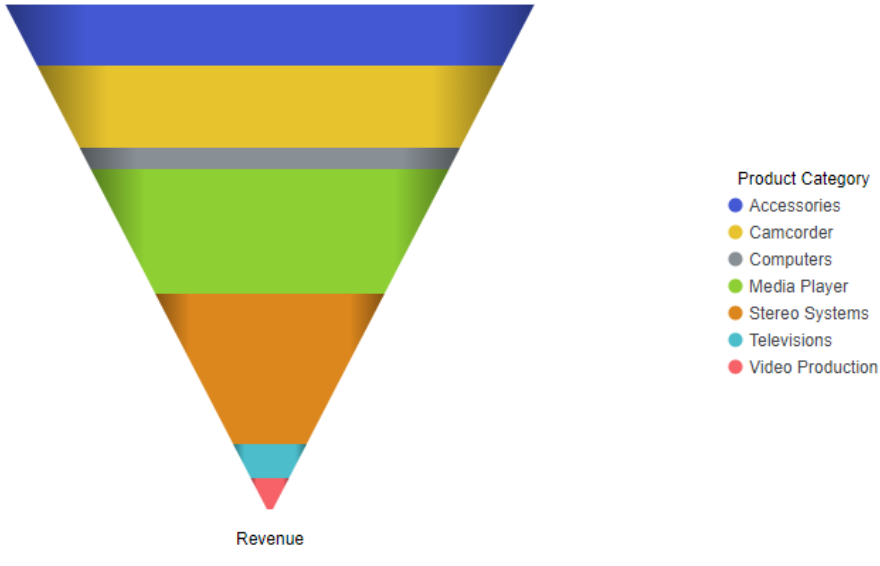
幅を定義する数値をピクセル単位で指定するか、パーセント値を文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます ("0%" から "100%")。デフォルト値は "20%" です。

例　　じょうご下端幅の制御

次のリクエストは、じょうごグラフを生成し、下端幅を 1 ピクセルに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"legend": {"visible": true},
"dataLabels": {"visible": false},
"funnelProperties": {"baseWidth": 1}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



じょうごグラフのレイヤ間隔の制御

riserGap プロパティは、じょうごグラフの各レイヤ間の間隔を制御します。

構文　じょうごグラフのレイヤ間隔の制御

```
"funnelProperties": {
   "riserGap": gap},
```

説明

`"riserGap": gap`

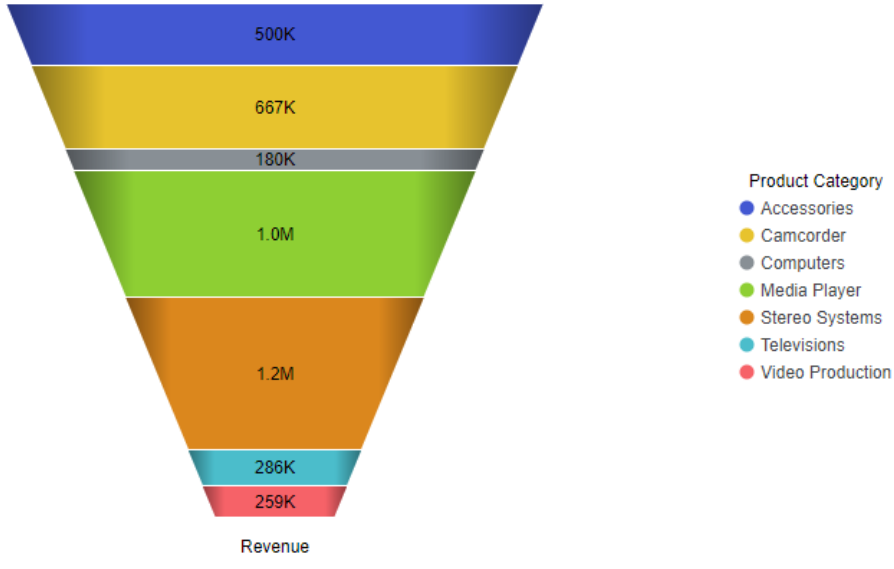
レイヤ間隔を定義する数値をピクセル単位で指定するか、レイヤの平均高さに対する割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます ("0%" から "100%")。たとえば、「"riserGap": "10%"」と指定すると、各じょうごレイヤの 10% がレイヤ間の隙間として使用されます。デフォルト値は 0 (ゼロ) です。

例　じょうごグラフのレイヤ間隔の制御

次のリクエストは、じょうごグラフを作成し、レイヤ間隔を 1 ピクセルに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"legend": {"visible": true},
"funnelProperties": {"riserGap": 1}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



じょうごグラフのグループラベルのフォーマット設定

構文　じょうごグラフのグループラベルのフォーマット設定

groupLabel プロパティは、じょうごグラフのグループラベルの表示およびフォーマットを制御します。

これらのプロパティでは、グループラベルとして表示するテキストは制御されません。テキストは、標準の groupLabels プロパティで制御されます。デフォルトのグループラベルは、リクエストで指定されたメジャーのフィールド名です。

```
"funnelProperties": {
   "groupLabel": {
      "visible": boolean,
      "font": "string",
      "color": "string"
   },
},
```

説明

`"visible": boolean`
有効な値には、次のものがあります。

- true - グループラベルを表示します。

- false - グループラベルを表示しません。これがデフォルト値です。

"font": "string"

ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

"color": "string"

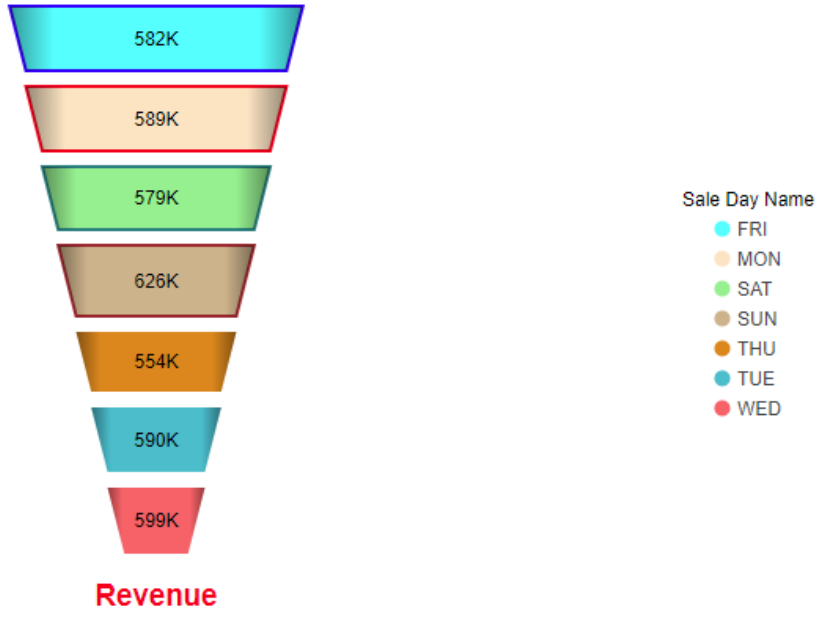
ラベルの色を定義する文字列です。デフォルト値は "black" です。

例　じょうごグラフのグループラベルの制御

次のリクエストは、じょうごグラフを作成し、グループラベルを赤色、14pt、Sans-Serif フォントで表示します。上端幅は 50%、下端幅は 3 ピクセル、ライザ間隔を 10 ピクセルに設定します。グループラベルは、メジャーフィールド名です (デフォルトラベル)。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US
BY TIME_DAYNAME
-* COLUMNS 'FRI' AND 'SAT' AND 'SUN' AND 'MON'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"legend": {"visible": true},
"funnelProperties": {"topWidth": "50%", "baseWidth": 3, "riserGap": 10,
   "groupLabel": {
      "visible": true, "font": "bold 14pt Sans-Serif", "color": "red"},
"series": [
   {"series": 0, "color": "cyan", "border": {"width": 2, "color": "blue"}},
   {"series": 1, "color": "bisque", "border": {"width": 2, "color": "red"}},
   {"series": 2, "color": "lightgreen", "border": {"width": 2, "color": "teal"}},
   {"series": 3, "color": "tan", "border": {"width": 2, "color": "brown"}}
   ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフのプロパティ (gaugeProperties)

メータグラフは、円形の面に 1 つまたは複数の値をメータ針とマーカーで表現したグラフです。このグラフタイプは、エグゼクティブダッシュボードアプリケーションでよく使用されます。

ここでは、メータグラフの全体的なレイアウトを制御するプロパティについて説明します。

次のコードセグメントは、メータグラフのプロパティおよびデフォルト値を示しています。

```
"gaugeProperties": {
    "startAngle": 135,
    "endAngle": 45,
    "secondaryRingFillColor": "undefined",
    "layout": "gauge",
    "groupLabel": {
        "visible": false,
        "font": "10pt Sans-Serif",
        "color": "black"},
    "totalLabel": {
        "visible": false,
        "font": "10pt Sans-Serif",
        "color": "black",
        "numberFormat": "auto"},
    "fill": {
        "color": "transparent"},
    "needleBase": {
        "size": "6%",
        "color": "#1f77b4",
        "border": {
            "width": 0,
            "color": "transparent",
            "dash": ""}
        },
    "axisWidth": "22%",
    "axisTickLength": "30%",
    "axisMinorTickLength": "8%",
    "outerBorder": {
        "width": "10%",
        "fill": {
            "color": "#1f77b4"},
        "border": {
            "width": 0,
            "color": "transparent",
            "dash": ""
            }
        }
    }
```

注意：値の範囲とフォーマット、罫線のフォーマット、メータ軸に表示する色バンドは、yaxis プロパティで制御します。デフォルト設定では、補助罫線は表示されますが、主罫線は表示されません。主罫線を表示した場合、主罫線の長さは axisTickLength プロパティで制御します。補助罫線の長さは axisMinorTickLength プロパティで制御します。

メータグラフのレイアウト制御

デフォルト設定では、メータグラフにはメータ針および目盛り付きバンドが表示されます。シンプル (simple) レイアウトでは、バンドと合計ラベルのみが表示されます。テキストアロー (textArrow) レイアウトでは、三角形のマーカーと合計ラベルが表示されます。シンプルテキスト (simpleText) レイアウトでは、合計ラベルのみが表示されます。

構文　メータグラフのレイアウト制御

```
"gaugeProperties": {
   "layout": "string"   },
```

説明

`"layout": "string"`

次のいずれかの値を指定します。

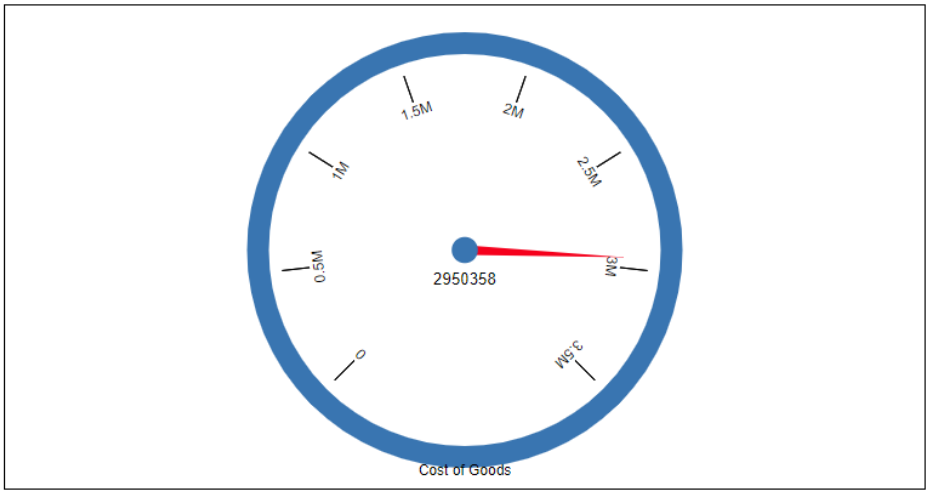
- **"gauge"** - 針付きの従来型メータグラフのレイアウトです。これがデフォルト値です。
- **"simple"** - バンドと合計ラベルのみのメータグラフレイアウトです。グループラベルはサポートされません。合計ラベルのフォントサイズは、"font" 文字列で指定したサイズにかかわらず自動的に生成されます。フォントサイズを手動で変更するには、"fontSize" プロパティを追加し、"fontSize": "20pt" のように、サイズと単位を指定する必要があります。
- **"textArrow"** - 合計ラベルおよび赤色の三角形マーカーが表示されます。三角形マーカーは、合計値が正の値の場合は上向き、負の値の場合は下向きに表示されます。totalLabel は常に表示されます。groupLabel は、表示または非表示に設定できます。デフォルト値は "visible" です。合計ラベルは、常にマーカーの右側に配置されます。グループラベルの配置は、"top"、"bottom" (デフォルト)、"right"、"left" のいずれかに設定できます。
- **"simpleText"** - 合計ラベルのみが表示され、イメージは表示されません。グループラベルは、表示または非表示に設定できます。デフォルト値は "visible" です。グループラベルの配置は、"top"、"bottom" (デフォルト)、"right"、"left" のいずれかに設定できます。

例　メータグラフでのシンプルレイアウトの使用

次のリクエストは、デフォルト設定のプロパティを使用して従来型のメータグラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=measure,$
ENDSTYLE
END
```

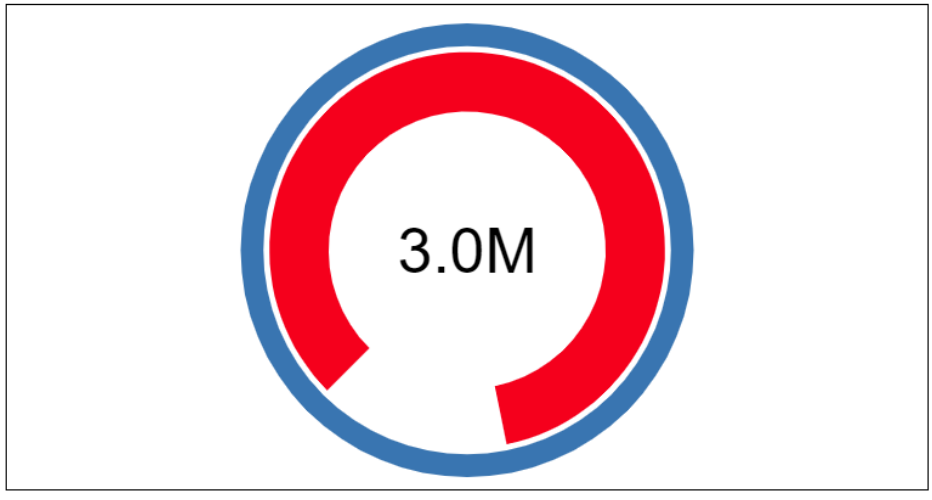
出力結果は次のとおりです。



次のリクエストでは、メータグラフのレイアウトを "simple" に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  COGS_US
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"layout": "simple"}
*END
ENDSTYLE
END
```

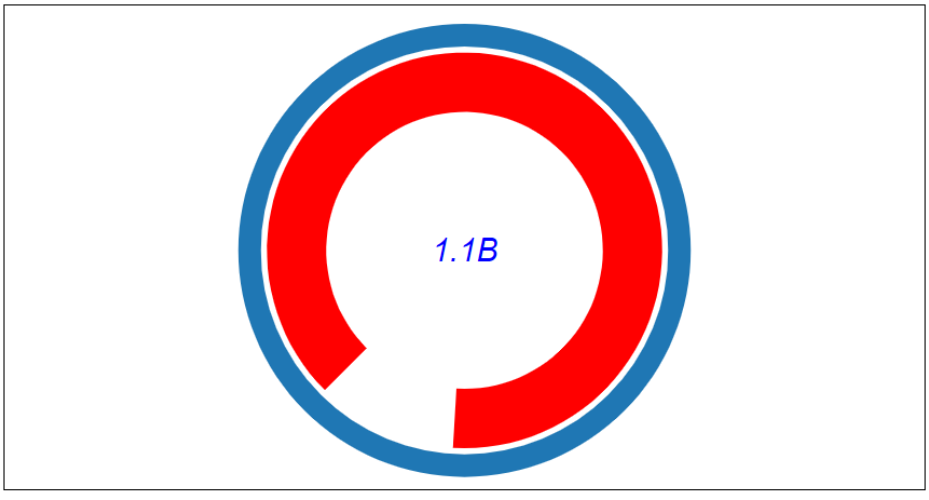
出力結果は次のとおりです。



次のリクエストでは、"font" 文字列を使用してフォントスタイルを変更し、"fontSize" 文字列を使用してフォントサイズを変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
  "layout": "simple",
  "totalLabel": {"font": "italic 12pt Sans-Serif", "color": "blue",
  "fontSize": "20pt"
  }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータ軸の色バンドの生成

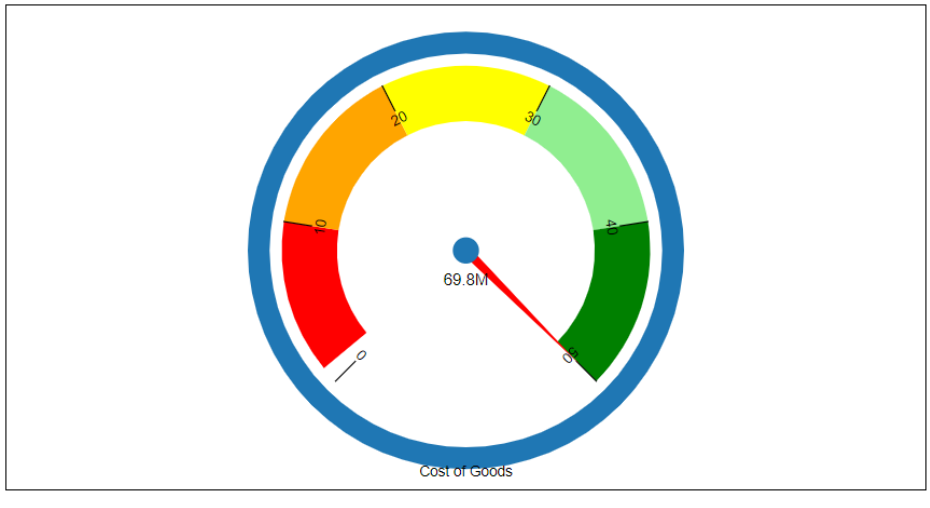
Y 軸プロパティを使用して、メータ軸の色バンドを生成することができます。詳細は、344 ページの 「 色バンドの定義 」 を参照してください。

例　メータグラフでの色バンドの定義

次のリクエストは、メータグラフを生成し、メータ軸に色バンドを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=COGS_US, bucket=measure,$
*GRAPH_JS
"yaxis": {
   "min": 0,
   "max": 50,
   "colorBands": [
      {"start": 1, "stop": 10, "color": "red"},
      {"start": 10, "stop": 20, "color": "orange"},
      {"start": 20, "stop": 30, "color": "yellow"},
      {"start": 30, "stop": 40, "color": "lightgreen"},
      {"start": 40, "stop": 50, "color": "green"}
      ]}
*END
ENDSTYLE
END
```

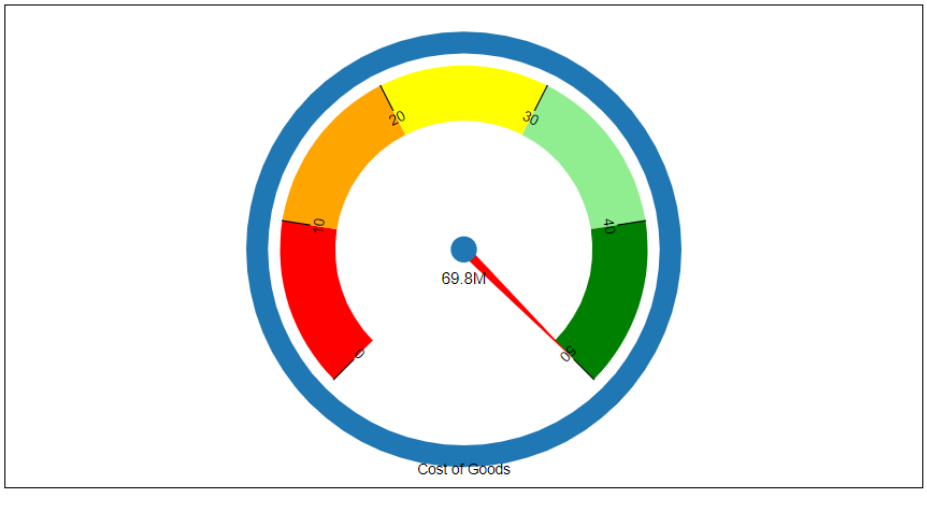
出力結果は次のとおりです。



次のリクエストは、色バンドの生成に、軸値の代わりにパーセント値を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=COGS_US, bucket=measure,$
*GRAPH_JS
"yaxis": {
"min": 0,
"max": 50,
"colorBands": [
{"start": "0%", "stop": "20%", "color": "red"},
{"start": "20%", "stop": "40%", "color": "orange"},
{"start": "40%", "stop": "60%", "color": "yellow"},
{"start": "60%", "stop": "80%", "color": "lightgreen"},
{"start": "80%", "stop": "100%", "color": "green"}
]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフの主罫線目盛り長さの制御

axisTickLength プロパティは、軸の主罫線目盛りの長さを定義します。デフォルト設定では、メータグラフの軸の主罫線目盛りは表示されません。

構文　メータグラフの主罫線目盛り長さの制御

```
"gaugeProperties": {
   "axisTickLength": length   },
```

説明

`"axisTickLength": length`

長さを定義する数値をピクセル単位で指定するか、メータ内径に対する長さの割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます。デフォルト値は "30%" です。

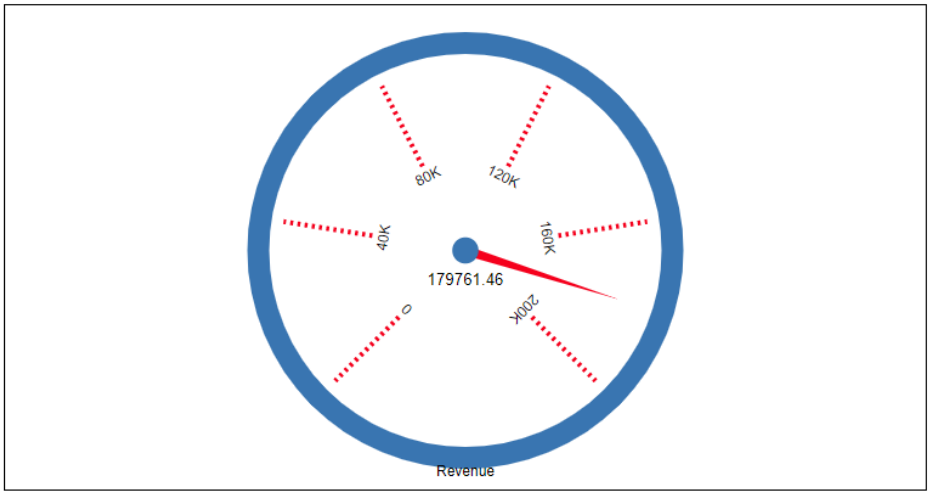
注意：目盛りのフォーマットを設定するには、yaxis majorGrid プロパティを使用します。

例　メータグラフの主罫線目盛り長さの制御

次のリクエストは、軸の主罫線の目盛り長さを 50% に設定します。Y 軸の majorGrid プロパティで、線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"axisTickLength": "50%"},
"yaxis": {"majorGrid": {
   "visible": true, "lineStyle": {
      "width": 4, "color": "red", "dash": "2 4"}
   }}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフの補助罫線目盛り長さの制御

axisMinorTickLength プロパティは、軸の補助罫線目盛りの長さを定義します。メータグラフの軸の補助罫線目盛りは、デフォルト設定で表示されます。

構文　メータグラフの補助罫線目盛り長さの制御

```
"gaugeProperties": {
   "axisMinorTickLength": length   },
```

説明

`"axisMinorTickLength": length`

長さを定義する数値をピクセル単位で指定するか、メータ内径に対する長さの割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます。デフォルト値は "8%" です。

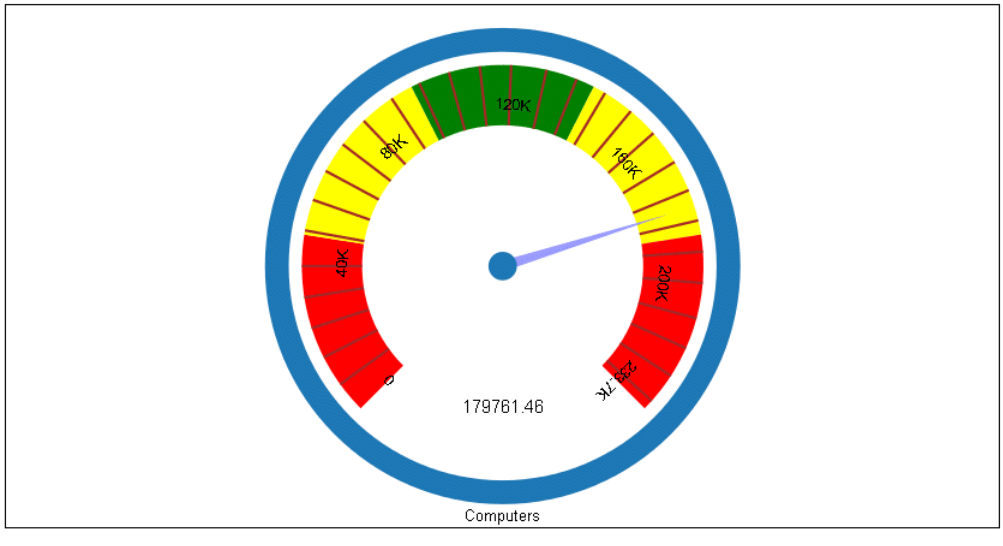
注意：目盛りのフォーマットを設定するには、yaxis minorGrid プロパティを使用します。

例　メータグラフの補助罫線目盛り長さの制御

次のリクエストは、軸の補助罫線目盛り長さを 30% に設定します。Y 軸の minorGrid プロパティで、線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"axisMinorTickLength": "30%"},
"yaxis": {
    "colorBands": [
        {start: 0, stop: 50000000, color: "red"},
        {start: 50000000, stop: 100000000, color: "yellow"},
        {start: 100000000, stop: 1200000000, color: "green"},
    ],

    "minorGrid": {"visible": true,
   "lineStyle": {
     "width": 2, "color": "brown", "dash": "1 4"}}
   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフの軸幅の制御

axisWidth プロパティは、メータグラフの軸幅を定義します。

構文　メータグラフの軸幅の制御

```
"gaugeProperties": {
   "axisWidth": width   },
```

説明

`"axisWidth": width`

幅を定義する数値をピクセル単位で指定するか、メータ内径に対する幅の割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます。デフォルト値は "22%" です。

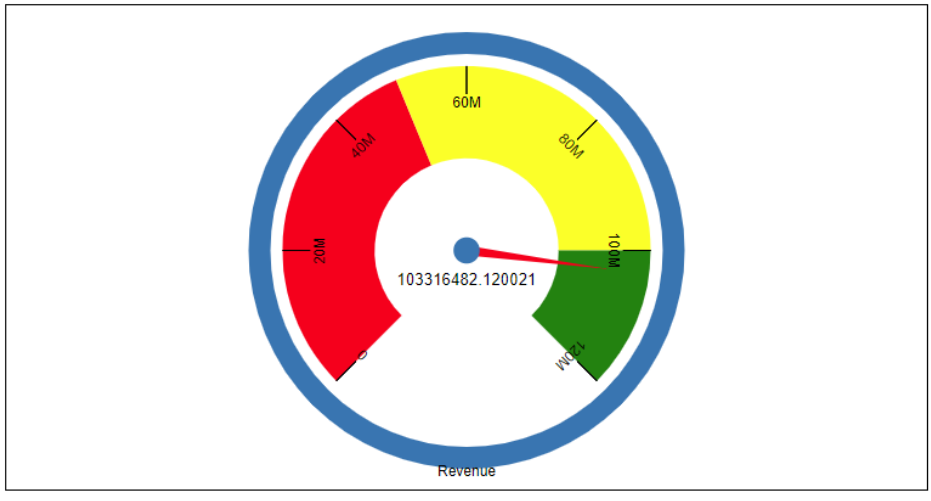
注意：メータグラフの軸のデフォルト色を変更するには、yaxis colorBands プロパティを使用します。

例　メータグラフの軸幅の制御

次のリクエストは、メータグラフを生成し、軸幅を 50% に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"axisWidth": "50%"},
"yaxis": {
"colorBands": [
    {start: 0, stop: 50000000, color: "red"},
    {start: 50000000, stop: 100000000, color: "yellow"},
    {start: 100000000, stop: 1200000000, color: "green"},
    ],
   }
*END
ENDSTYLE
END
```

出力結果では、軸を含む色バンドがメータ領域の 50% を占有しています。



メータ面の内側塗りつぶし色の制御

fill プロパティは、メータ面の内側の塗りつぶし色を制御します。

構文　メータ面の内側塗りつぶし色の制御

```
"gaugeProperties": {
   "fill": {
      "color": "color"   },
}
```

説明

`"color": "color"`

JSON グラデーション定義を指定するか、色名または数値指定の文字列で定義する色、あるいはグラデーションを定義する文字列を指定します。文字列は、二重引用符 (") で囲みます。

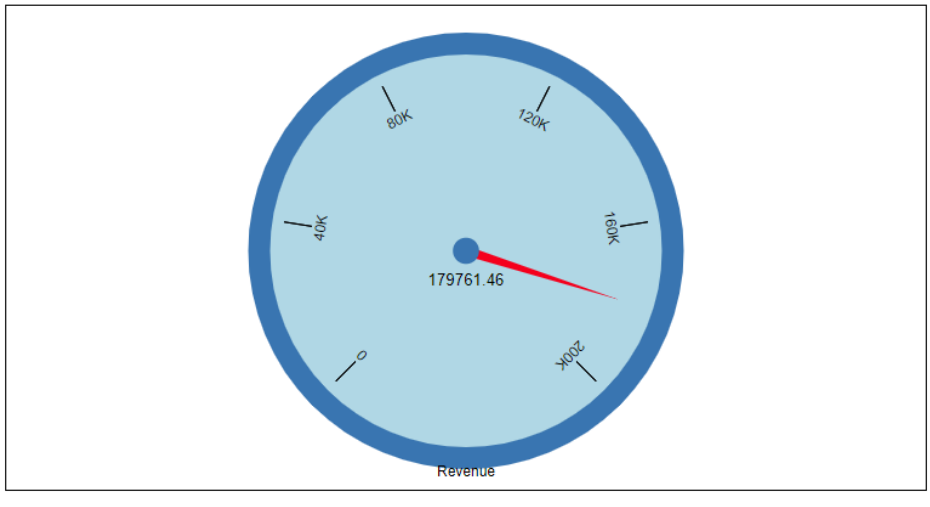
注意：色およびグラデーションの定義についての詳細は、63 ページの「色およびグラデーション」を参照してください。

例　メータ面の塗りつぶし色の制御

次のリクエストは、メータ面の内側を水色 ("lightBlue") で塗りつぶします。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"fill": {"color": "lightBlue"}},

*END
ENDSTYLE
END
```

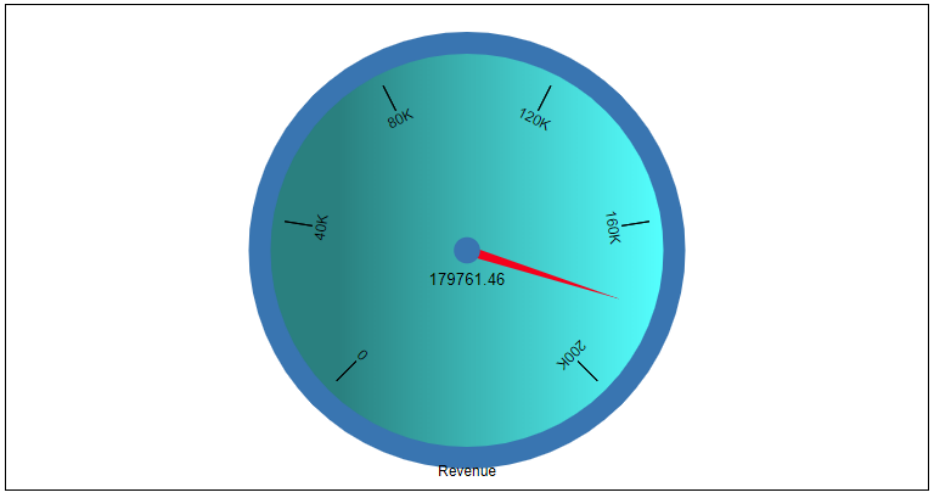
出力結果は次のとおりです。



次のリクエストは、塗りつぶし色を、青緑色からシアンに変化する線状グラデーションに変更します。

```
"gaugeProperties": {
    "fill": {"color": "linear-gradient(0%,0%,100%,0%, 20% teal, 95% cyan)"}}
```

出力結果は次のとおりです。



メータグラフのグループラベルのフォーマット設定

groupLabel プロパティは、メータグラフのグループラベルの表示およびフォーマットを制御します。

構文　メータグラフのグループラベルのフォーマット設定

```
"gaugeProperties": {
   "groupLabel": {
      "visible": boolean,
      "font": "string",
      "color": "string"
   },
},
```

説明

`"visible": boolean`

有効な値には、次のものがあります。

- true - グループラベルを表示します。これがデフォルト値です。
- false - グループラベルを表示しません。

`"font": "string"`

ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

```
"color": "string"
```

ラベルの色を定義する文字列です。デフォルト値は "black" です。

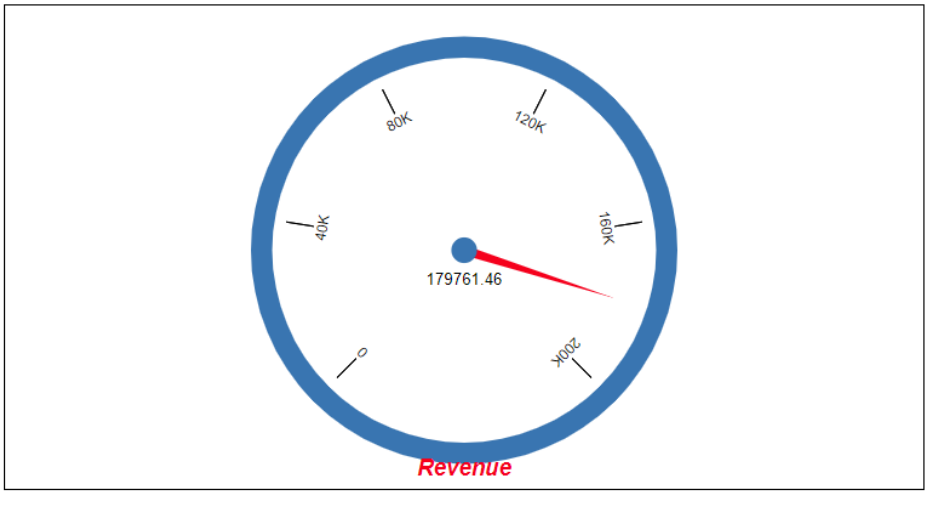
注意：gaugeProperties の groupLabel プロパティでは、グループラベルに表示するテキストは変更されません。テキストを変更するには、標準の groupLabels プロパティを使用します。デフォルトのグループラベルは、メータグラフの描画に使用されるソートフィールド値です。

例　メータグラフのグループラベルのフォーマット設定

次のリクエストは、グループラベルを赤色、太字、斜体、14pt、Sans Serif フォントに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"groupLabel": {
   "font": "bold italic 14pt Sans-Serif", "color": "red"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータ針の回転軸部のフォーマット設定

needleBase プロパティは、メータ針の回転軸部のフォーマットを制御します。

構文　メータ針の回転軸部のフォーマット設定

```
"gaugeProperties": {
   "needleBase": {
      "size": size,
      "color": "color",
      "border": {
         "width": number,
         "color": "string",
         "dash": "string"
      }
   },
},
```

説明

`"size": size`

回転軸部の半径を定義する数値をピクセル単位で指定するか、メータ内径に対する半径の割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます。デフォルト値は "6%" です。

`"color": "color"`

JSON グラデーション定義を指定するか、色名または数値指定の文字列で定義する色、あるいはグラデーションを定義する文字列を指定します。文字列は、二重引用符 (") で囲みます。色およびグラデーションの指定についての詳細は、63 ページの「色およびグラデーション」を参照してください。

`"border":`

メータ針の回転軸部境界線のプロパティを定義します。

`"width": number`

メータ針の回転軸部を囲む境界線の幅をピクセル単位で指定します。デフォルト値は 0 (ゼロ) です (境界線なし)。

`"color": "string"`

色名または数値指定の文字列で定義する色です。デフォルト値は "transparent" です (境界線なし)。

```
"dash": "string"
```

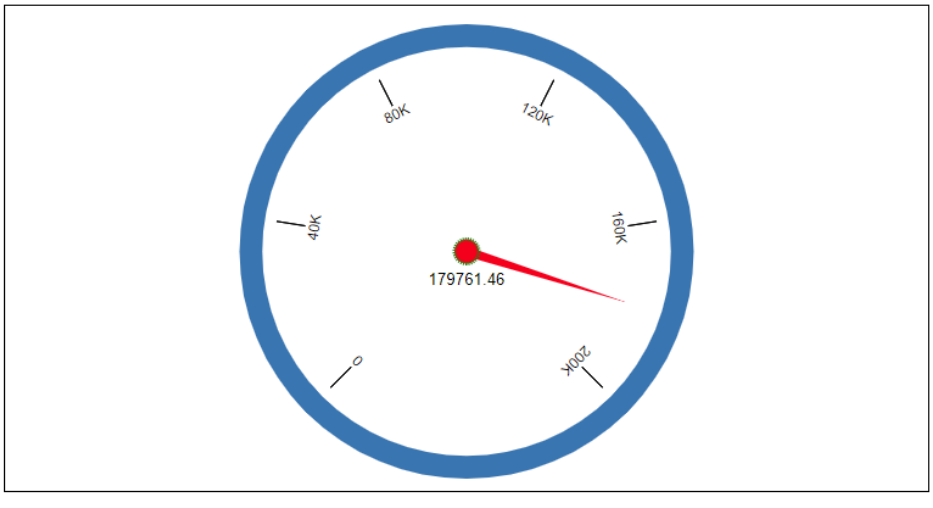
境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。

例　メータ針の回転軸部のフォーマット設定

次のリクエストは、メータ針の回転軸部の半径を 10 ピクセルの赤色、境界線を破線幅 3 ピクセルの緑色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
   "groupLabel": {"visible": false},
   "needleBase": {
      "size": 10, "color": "red",
      "border": {"width": 3, "color": "green", "dash": "1 1"}
      }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフの外側境界のフォーマット設定

outerBorder プロパティは、メータグラフの外側境界のフォーマットを制御します。

構文　メータグラフの外側境界のフォーマット設定

```
"gaugeProperties": {
   "outerBorder": {
      "width": width,
      "shape": "string",
      "fill": {
         "color": "color",
      },
      "border": {
         "width": number,
         "color": "string",
         "dash": "string"
      }
   }
},
```

説明

`"width": width`

外側境界のリング幅を定義する数値をピクセル単位で指定するか、メータ内径に対するリング幅の割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます。デフォルト値は "10%" です。

`"shape": "string"`

外側境界の形状を制御します。有効な値には、次のものがあります。

- "circle" - メータバンド、メータ針、合計ラベルを囲む円として外側境界を描画します。これがデフォルト値です。
- "tight" - メータバンドとメータ針のみを囲む外側境界を描画します (半円メータ形状)。

`"color": "color"`

メータの外側境界の塗りつぶし色を定義します。JSON グラデーション定義を指定するか、色名または数値指定の文字列で定義する色、あるいはグラデーションを定義する文字列を指定します。文字列は、二重引用符 (") で囲みます。デフォルト値は "#1f77b4" です。色およびグラデーションの指定についての詳細は、63 ページの 「色およびグラデーション」 を参照してください。

"border":
メータの外側境界を囲む境界線のプロパティを定義します。

"width": number

メータの外側境界の境界線の幅をピクセル単位で定義します。デフォルト値は 0 (ゼロ) です (境界線なし)。

"color": "string"

色名または数値指定の文字列で定義する色です。デフォルト値は "transparent" です (境界線なし)。

"dash": "string"

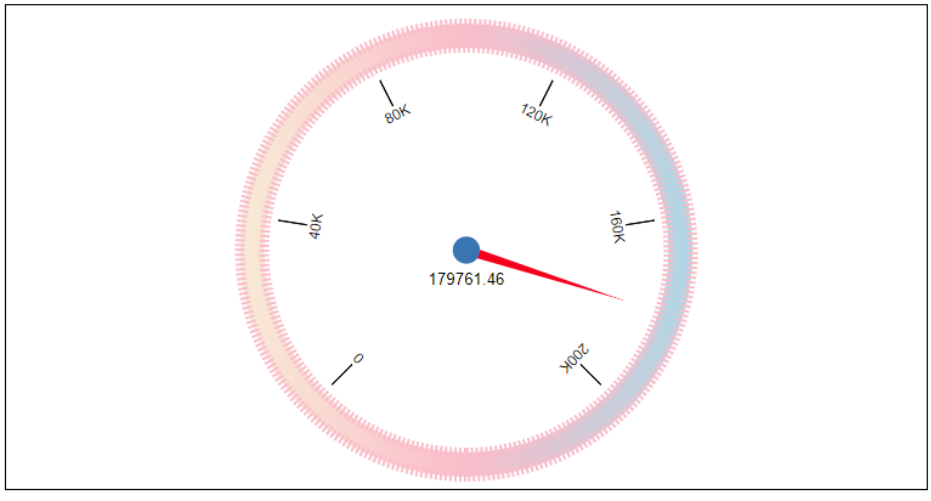
境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。

例　メータグラフの外側境界のフォーマット設定

次のリクエストは、メータグラフの外側境界の幅を 20 ピクセルに設定し、塗りつぶし色としてアンティークホワイト、ピンク、水色に変化する線状グラデーションを指定します。外側境界の境界線は、破線幅が 8 ピクセルのピンク色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
   "groupLabel": {"visible": false},
   "outerBorder": {
      "width": 20, "fill": {
         "color": "linear-gradient(0%,0,100%,0, 0 antiquewhite, 0.5 pink,1
lightblue)"},
      "border": {"width": 8, "color": "pink", "dash": "2 2"}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

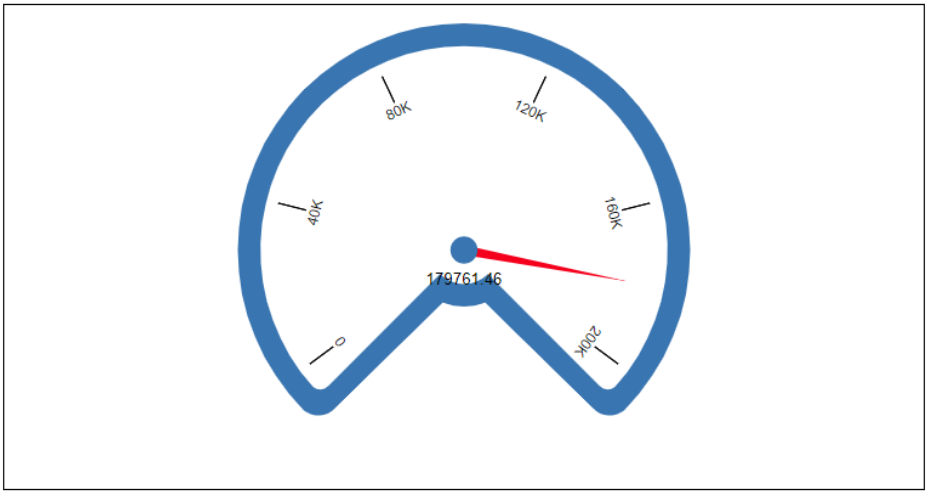


例　半円状のメータグラフ外側境界の描画

次のリクエストは、メータグラフの外側境界の形状を "tight" に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
   "groupLabel": {"visible": false},
   "outerBorder": {"shape": "tight"}
   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフの軸の開始角度と終了角度の制御

startAngle および endAngle プロパティは、メータ値バンドの開始角度と終了角度を制御します。角度は、度数で定義します。正の角度は時計回り、負の角度は反時計回りを表します。0 (ゼロ) を指定すると、バンドの開始位置または終了位置は 3 時の位置になります。90 を指定すると、バンドの開始位置または終了位置は 6 時の位置になります。-90 を指定すると、12 時の位置になります。

注意：メータグラフの軸は「値バンド」とも呼ばれます。

構文　メータグラフの軸の開始角度と終了角度の制御

```
"gaugeProperties": {
   "startAngle": number,
   "endAngle": number},
```

説明

`"startAngle": number`

メータ値バンドの開始位置を定義する角度 (度数) を数値で指定します。デフォルト値は 135 です。

"endAngle": number

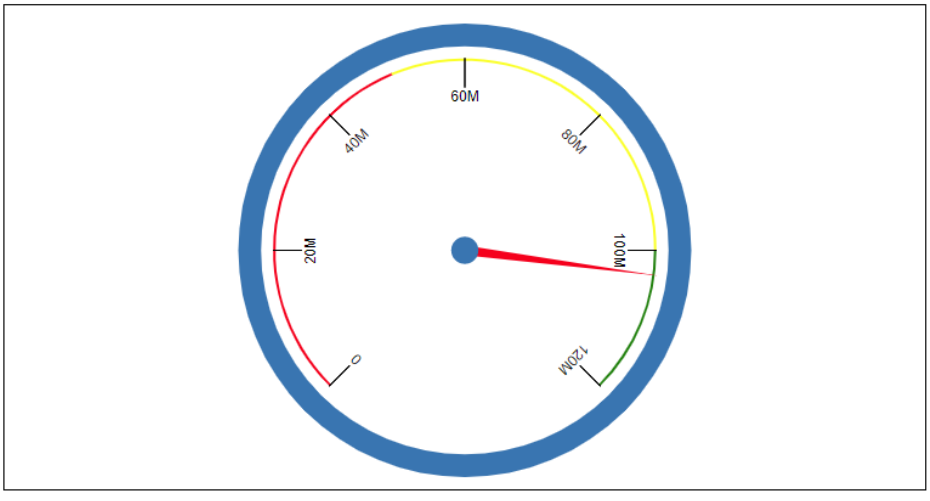
メータ値バンドの終了位置を定義する角度 (度数) を数値で指定します。デフォルト値は 45 です。

例　メータグラフの軸の開始角度と終了角度の制御

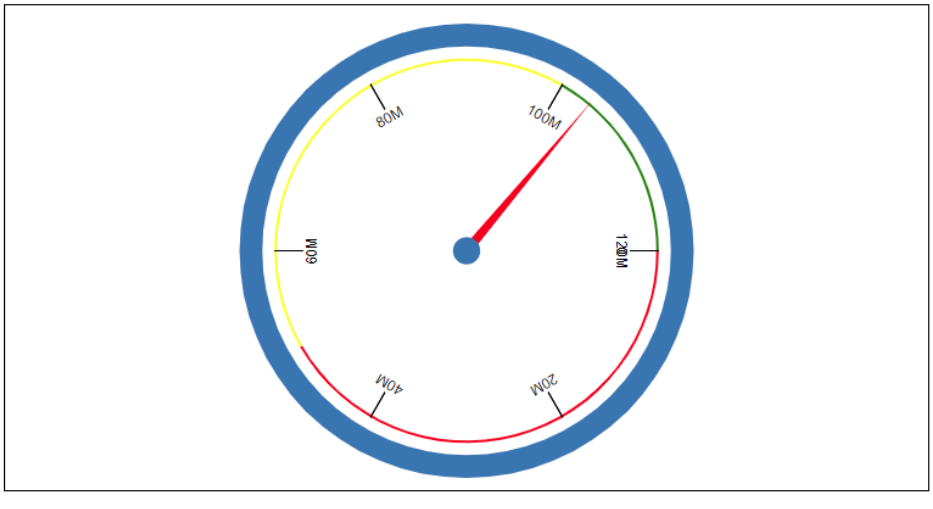
次のリクエストは、メータグラフの値バンドにデフォルトの開始角度 (135) および終了角度 (45) を使用します。軸幅を 2 ピクセルに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
   "groupLabel": {"visible": false}, "totalLabel": {"visible": false},
   "axisWidth": 2, "startAngle": 135, "endAngle": 45},
"yaxis": {
    "colorBands": [
        {start: 0, stop: 50000000, color: "red"},
        {start: 50000000, stop: 100000000, color: "yellow"},
        {start: 100000000, stop: 1200000000, color: "green"},
    ]
   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



開始角度および終了角度を 0 (ゼロ) に変更すると、下図のように、値バンドが完全な円になります。



シンプルメータグラフのセカンダリリング塗りつぶし色の制御

メータグラフのレイアウトが "simple" に設定されている場合、開始角度から終了角度までを超えるメータバンド部分の塗りつぶし色を定義することができます。

構文　シンプルメータグラフのセカンダリリング塗りつぶし色の制御

```
"gaugeProperties": {
   "layout": "simple",
   "secondaryRingFillColor": "string"
              }
```

説明

`"secondaryRingFillColor": "string"`

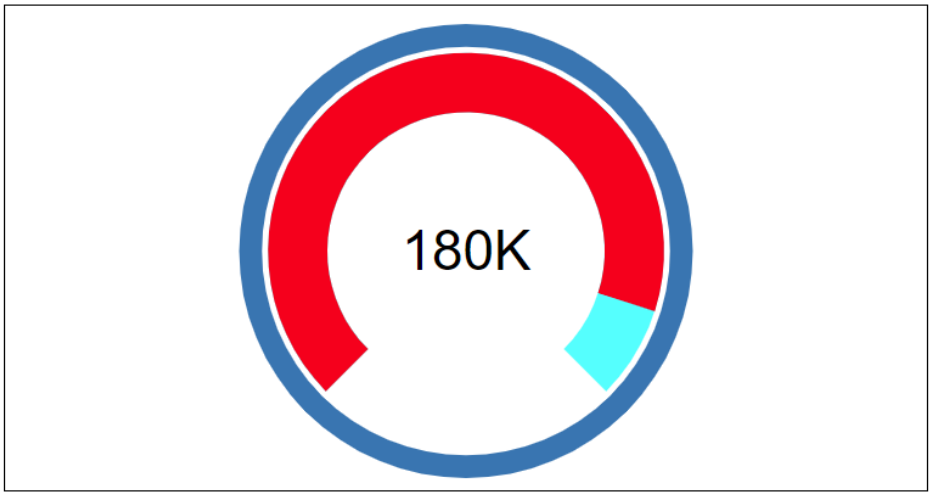
開始角度から終了角度までを超えるメータバンド部分の塗りつぶし色を定義します。デフォルト値は undefined です。 このプロパティは、従来のメータグラフレイアウトには有効ではありません。

例　シンプルメータグラフのセカンダリリング塗りつぶし色の制御

次のリクエストは、デフォルト設定の開始角度から終了角度までを超えるメータバンド部分の色をシアンに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {"layout": "simple",
   "secondaryRingFillColor": "cyan"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



メータグラフの合計ラベルのフォーマット設定

totalLabel プロパティは、メータグラフの合計ラベルの表示およびフォーマットを制御します。

構文　メータグラフの合計ラベルのフォーマット設定

```
"gaugeProperties": {
   "totalLabel": {
      "visible": boolean,
      "font": "string",
      "color": "string",
      "numberFormat": numformat   },
},
```

説明

`"visible": boolean`

有効な値には、次のものがあります。

- true - ラベルを表示します。
- false - ラベルを表示しません。 これがデフォルト値です。

`"font": "string"`

ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

"color": "string"

色名または数値指定の文字列でラベルの色を定義します。デフォルト値は "black" です。

色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"numberFormat": numformat

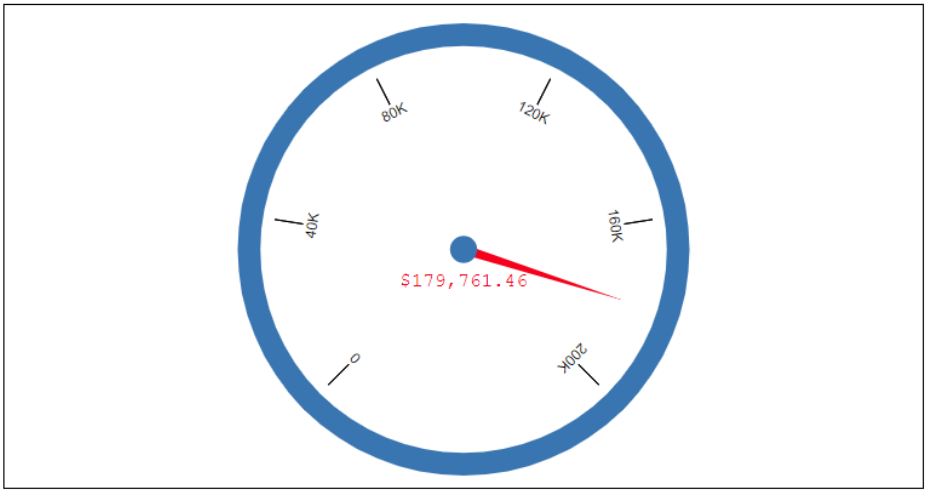
JSON オブジェクト、フォーマット文字列、またはユーザ定義関数として指定することができます。デフォルト値は "auto" です。数値フォーマットの指定についての詳細は、86 ページの 「 数値のフォーマット設定 」 を参照してください。

例　メータグラフの合計ラベルのフォーマット設定

次のリクエストは、メータグラフに合計ラベルを表示し、ラベルを赤色、12 pt Courier フォント、通貨記号フォーマットに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH GAUGE
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=measure,$
*GRAPH_JS
"gaugeProperties": {
   "groupLabel": {"visible": false},
   "totalLabel": {"visible": true, "font": "12pt Courier", "color": "red",
      "numberFormat": {"mode": "currency"}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ヒストグラムのプロパティ (histogramProperties)

ヒストグラムは、データの度数分布を視覚的に表現したグラフです。histogramProperties オブジェクトは、データ分布に関連するプロパティを制御します。

構文　ヒストグラムのプロパティの制御

```
"histogramProperties": {
   "binCount": number,
   "binSize": size,
   "startBinValue": number}
```

説明

`"binCount": number`

描画するグループラベルの個数、または undefined (自動) を指定します。デフォルト値は undefined です。

`"binSize": size`

データ値をビン (棒) に割り当てる値範囲のサイズです。数値、数値配列 (可変ビンサイズ)、undefined (自動) のいずれかを指定します。binSize が配列の場合、binCount は無視され、binCount の値が binSize 配列の長さと見なされます。デフォルト値は undefined です。

"startBinValue": number

先頭のビンとして使用する数値、または undefined (自動) を指定します。デフォルト値は undefined です。

注意

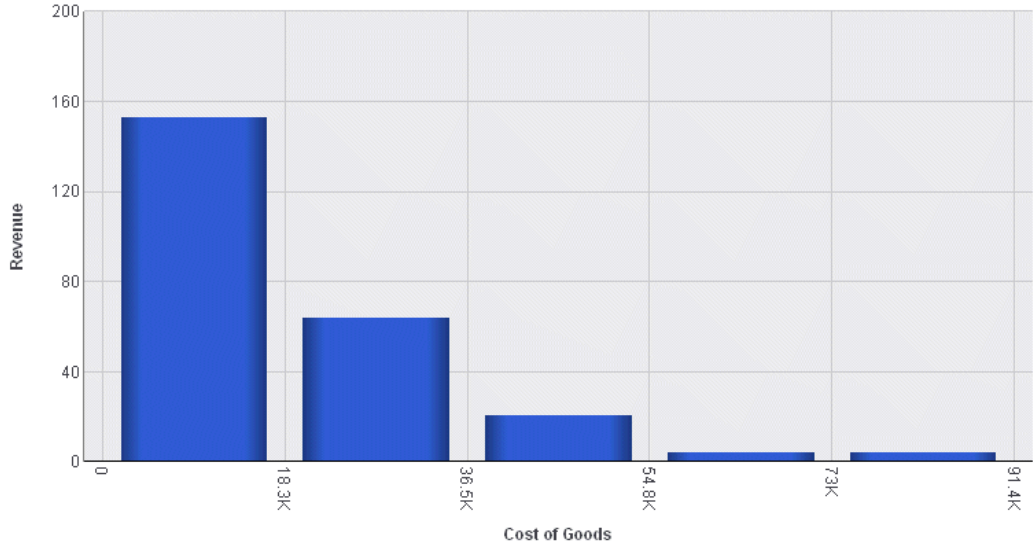
- blaProperties の orientation プロパティを使用すると、横方向または縦方向のフォーマットでグラフが描画されます。ただし、このプロパティの代わりに LOOKGRAPH パラメータを使用することで、グラフの方向を制御することもできます。このパラメータには、ヒストグラムの各方向を設定する値があります。LOOKGRAPH パラメータについての詳細は、40 ページの 「 グラフタイプの制御 」 を参照してください。
- 縦ヒストグラムでは、グラフの左側に Y 軸、グラフの下側に X 軸が描画されます。
- 横ヒストグラムでは、グラフの左側に X 軸、グラフの下側に Y 軸が描画されます。
- ヒストグラムのライザ間の間隔を制御するには、"blaProperties": "barGroupGapWidth" を使用します。
- ヒストグラムのライザの色および境界 (存在する場合) を制御するには、シリーズ固有のプロパティを使用します。
- Y 軸のタイトルおよびラベルのフォーマットを制御するには、yaxis プロパティを使用します。
- X 軸のタイトルおよびラベルのフォーマットを制御するには、xaxis プロパティを使用します。X 軸のラベルには、計算された値が表示されます。これらのラベルは、groupLabels プロパティでは定義されません。

例 ヒストグラムのビン数の設定

次のリクエストは、縦ヒストグラム (ON GRAPH SET LOOKGRAPH VHISTOGR) を作成し、ビン数を 5 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY COGS_US
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VHISTOGR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"histogramProperties": {"binCount": 5},
"legend": {"visible": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックスグラフのプロパティ (matrixProperties)

マトリックスグラフのプロパティは、マトリックスのラベル、見出し、線、および各セルのグラフタイプのフォーマットを定義します。マトリックスグラフの生成には、WebFOCUS グラフ属性構文を使用します。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

マトリックス棒グラフ、折れ線グラフ、面グラフでは、X 軸タイトルが各 X 軸で繰り返し表示されず、マトリックス列見出しの末尾に連結されます。

以下は、マトリックスグラフのプロパティとそのデフォルト値を示しています。

```
"matrixProperties": {
    "chartType": 'bar',
    "colWrapCount": 0,
    "rowLabels": {
        "labels": undefined,
        "font": "8pt Sans-Serif",
        "color": "black",
        "tooltip": undefined
                },
    "colLabels": {
      "labels": undefined,
      "font": "8pt Sans-Serif",
      "color": "black",
      "tooltip": undefined
         },
    "rowHeader": {
      "text": undefined,
      "font": "10pt Sans-Serif",
      "color": "black",
      "tooltip": undefined
         },
    "colHeader": {
      "text": undefined,
      "font": "10pt Sans-Serif",
      "color": "black",
      "tooltip": undefined
         },
    "cellBorder": {
      "width": 1,
      "color": "grey",
      "dash": ""
         },
    "layout":
        {"cellPadding":
             {
             "top": 0,
             "bottom": 0,
             "left": 0,
             "right": 0
             }
         }
    "minCellSize": "auto",
   "yaxis": {
     "min": "autoPerRow",
     "max": "autoPerRow"
        }
}
```

マトリックスの各セルに表示されるグラフタイプの制御

chartType プロパティは、マトリックスの各セルに表示されるグラフタイプを制御します。マトリックスの各セルに表示するグラフタイプに対して、正しいソートフィールドとメジャー、および対応する属性カテゴリが設定されていることを確認します。

たとえば、棒グラフ、折れ線グラフ、面グラフは同じ属性カテゴリをサポートするため、chartType プロパティの値を変更することでこれらを相互に置き換えて表示することができます。また、WebFOCUS のオートプロンプト機能を使用して、選択可能なグラフタイプのドロップダウンリストをユーザに提示することもできます。

構文　マトリックスグラフに表示されるグラフタイプの制御

```
"matrixProperties": {
    "chartType": "string"},
```

説明

`"chartType": "string"`

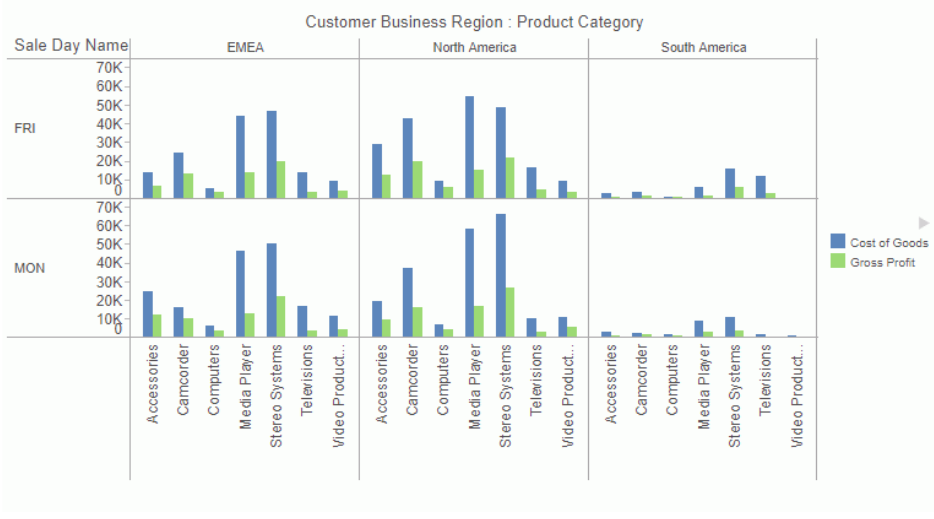
リクエストの属性カテゴリでサポートされるグラフタイプです。デフォルト値は "bar" です。

例　マトリックスグラフに表示されるグラフタイプの制御

次のリクエストは、chartType プロパティの定義にダイアログマネージャ変数を使用します。-SET コマンドの値に "bar" を指定します。

```
-SET &TYPE=bar;
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {"chartType": "&TYPE"}
*END
ENDSTYLE
END
```

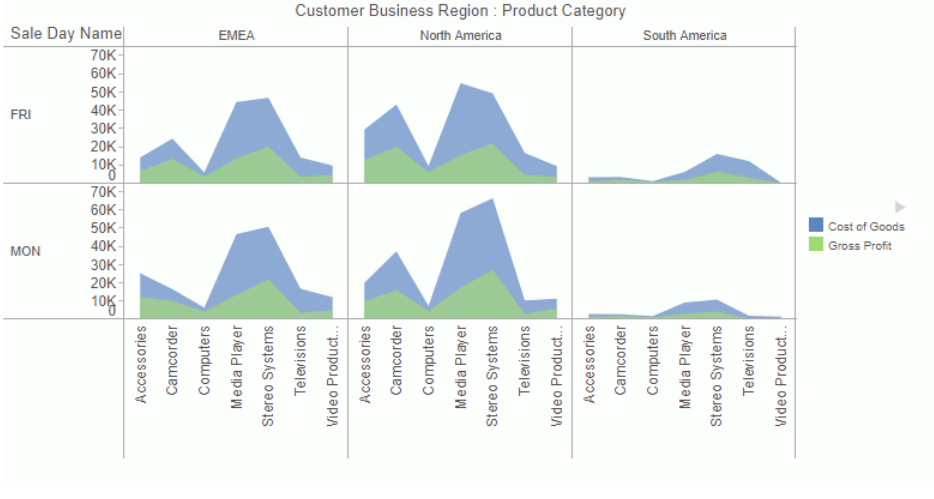
出力結果は次のとおりです。



-SET コマンドを以下に変更して面グラフを生成します。

```
-SET &TYPE=area;
```

出力結果は次のとおりです。



マトリックス列グラフのファセットモード

マトリックスグラフに行が指定されておらず、列の属性カテゴリに割り当てられたディメンションが 1 つのみの場合、ファセットモードを実装することで、マトリックスグラフの列に折り返しを設定することができます。ファセットモードを実装するには、折り返しまでの数を指定します。この数に到達すると、折り返しが実行されて、新しい行から列が開始されます。この機能は、「ファセット」と呼ばれます。デフォルト設定の折り返し数は 0 (ゼロ) で、折り返しは実行されません。

ファセットマトリックス列グラフには、マトリックス列見出しはありません。代わりに、各グラフに列ディメンション値で構成されるそれぞれの見出しがあります。

構文　マトリックス列グラフのファセットモードの実装

```
{"matrixProperties":
          "colWrapCount": number
}
```

説明

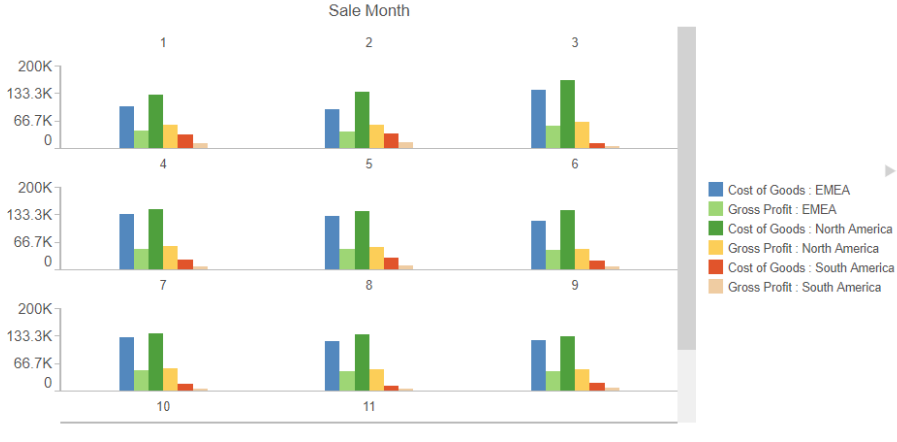
`colWrapCount: number`
: 折り返しを実行するまでに表示するグラフの数を指定します。値 "auto" を使用して、この数の指定をグラフエンジンに委ねることもできます。デフォルト値の 0 (ゼロ) は、グラフの折り返しを行いません。

例　マトリックス列グラフでの棒グラフのファセット設定

次のリクエストでは、列属性カテゴリとして TIME_MTH ディメンションが割り当てられています。"colWrapCount": 3 プロパティによって、3 つのグラフを表示するごとに、折り返しが実行されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
GROSS_PROFIT_US
BY TIME_MTH
BY BUSINESS_REGION
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_MTH, BUCKET=column, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"matrixProperties":
{
"colWrapCount": 3,
"cellBorder": {"width": 0}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



下図は、"colWrapCount" を "auto" に変更して出力した結果を示しています。

マトリックス行ラベルのフォーマット設定

マトリックス行ラベルは、各行の左側に表示されるラベルです。

構文　マトリックス行ラベルのフォーマット設定

```
"matrixProperties": {
       "rowLabels": {
       "labels": undefined,
       "font": "string",
       "color": "string",
       "tooltip": "string"
}
```

説明

`"font": "string"`

行ラベルテキストのフォントを定義する文字列です。デフォルト値は "8pt Sans-Serif" です。

`"color": "string"`

行ラベルの色を定義する文字列です。デフォルト値は "black" です。

`"tooltip": "string"`

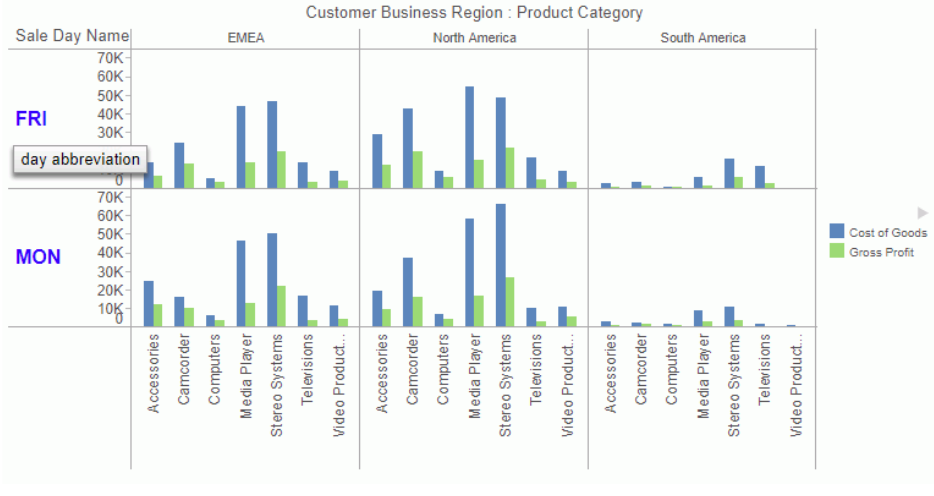
マウスポインタを行ラベル上に置いたときに表示するツールヒントです。

例　マトリックス行ラベルのフォーマット設定

次のリクエストは、行ラベルを青色、太字、12 pt、Sans Serif フォントに設定し、ツールヒントに「day abbreviation」と表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {"rowLabels": {
    "font": "bold 12pt Sans-Serif",
    "color": "blue",
    "tooltip": "day abbreviation"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックス列ラベルのフォーマット設定

マトリックス列ラベルは、各列の上部に表示されるラベルです。

構文　マトリックス列ラベルのフォーマット設定

```
"matrixProperties": {
       "colLabels": {
       "labels": undefined,
       "font": "string",
       "color": "string",
       "tooltip": "string"
}
```

説明

`"font": "string"`

列ラベルテキストのフォントを定義する文字列です。デフォルト値は "8pt Sans-Serif" です。

`"color": "string"`

列ラベルの色を定義する文字列です。デフォルト値は "black" です。

`"tooltip": "string"`

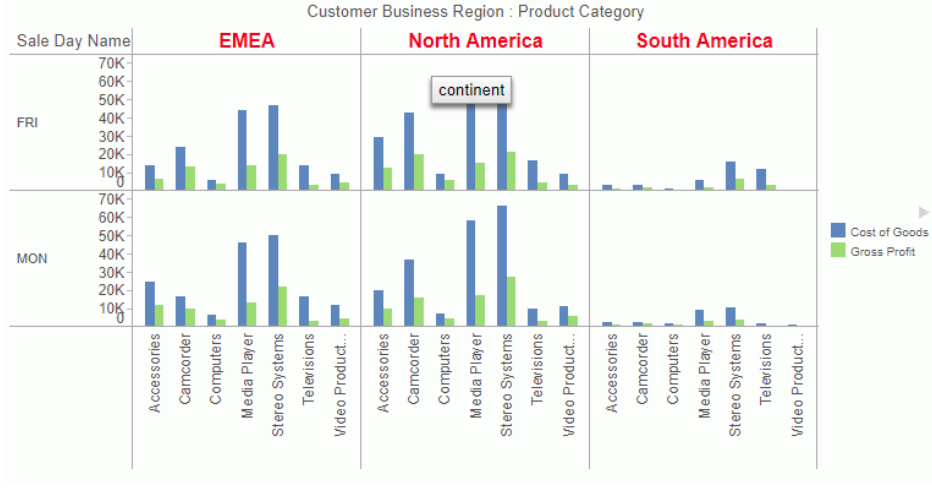
マウスポインタを列ラベル上に置いたときに表示されるツールヒントです。

例　マトリックス列ラベルのフォーマット設定

次のリクエストは、列ラベルを赤色、太字、12 pt、Sans Serif フォントに設定し、ツールヒントに「continent」と表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {"colLabels": {
    "font": "bold 12pt Sans-Serif",
    "color": "red",
    "tooltip": "continent"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックス行見出しのフォーマット設定

マトリックスグラフの行見出しは、行の最上部に描画される単一ラベルです。

構文　マトリックス行見出しのフォーマット設定

```
"matrixProperties": {
       "rowHeader": {
       "text": undefined,
       "font": "string",
       "color": "string",
       "tooltip": "string"
}
```

説明

`"font": "string"`

行見出しテキストのフォントを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

行見出しの色を定義する文字列です。デフォルト値は "black" です。

`"tooltip": "string"`

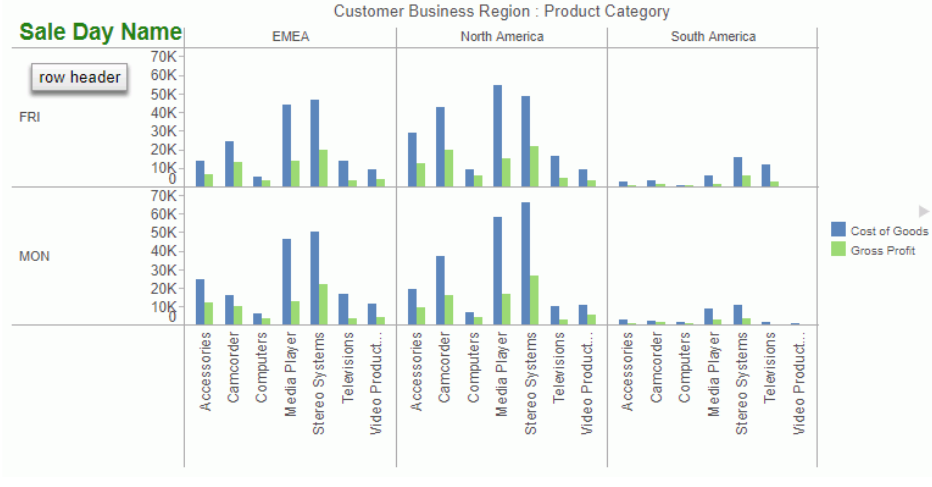
マウスポインタを行見出し上に置いたときに表示されるツールヒントです。

例　マトリックス行見出しのフォーマット設定

次のリクエストは、行見出しを緑色、太字、14 pt、Sans Serif フォントに設定し、ツールヒントに「row header」と表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {"rowHeader": {
    "font": "bold 14pt Sans-Serif",
    "color": "green",
    "tooltip": "row header"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックス列見出しのフォーマット設定

マトリックスグラフの列見出しは、列の最上部に描画される単一ラベルです。

構文　マトリックス列見出しのフォーマット設定

```
"matrixProperties": {
       "colHeader": {
       "text": undefined,
       "font": "string",
       "color": "string",
       "tooltip": "string"
}
```

説明

`"font": "string"`

列見出しテキストのフォントを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

列見出しの色を定義する文字列です。デフォルト値は "black" です。

`"tooltip": "string"`

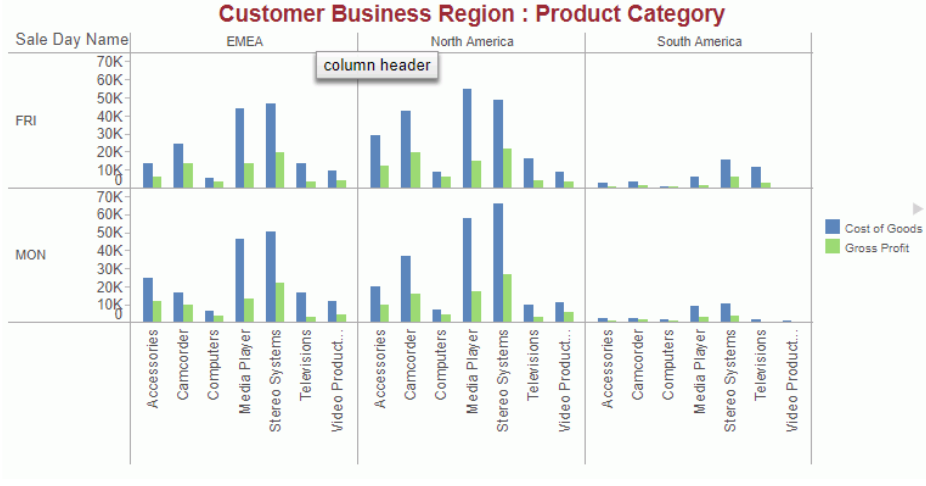
マウスポインタを列見出し上に置いたときに表示されるツールヒントです。

例　マトリックス列見出しのフォーマット設定

次のリクエストは、列見出しを茶色、太字、14 pt、Sans Serif フォントに設定し、ツールヒントに「column header」と表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {"colHeader": {
    "font": "bold 14pt Sans-Serif",
    "color": "brown",
    "tooltip": "column header"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックスセル境界線のフォーマット設定

セル境界線は、マトリックスの各セル間に描画される線です。

構文　マトリックスセル境界線のフォーマット設定

```
"matrixProperties": {
       "cellBorder": {
       "width": number,
       "color": "string",
       "dash": "string"
}
```

説明

`"width": number`

セル境界線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

セル境界線の色を定義する文字列です。デフォルト値は "grey" です。

`"dash": "string"`

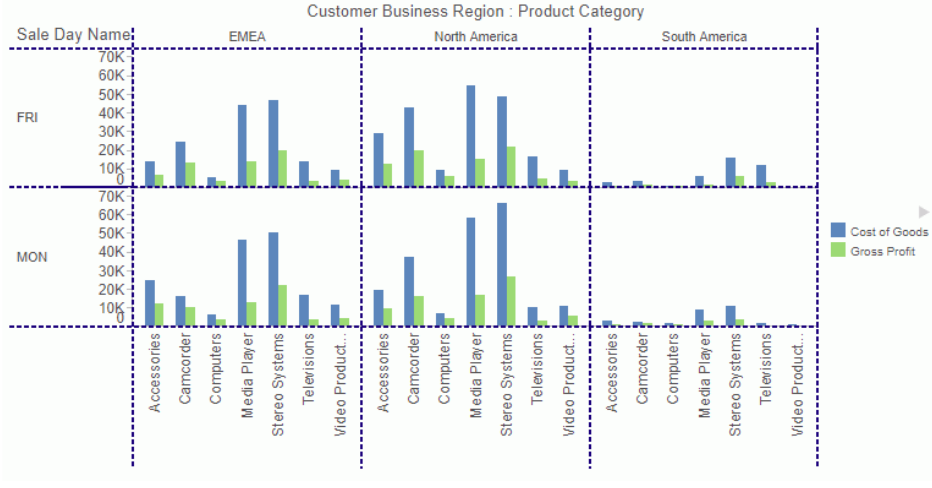
セル境界線の破線スタイルを定義します。破線長さ、破線間隔の順にピクセル数で指定します。デフォルト値は「""」です (実線)。

例　マトリックスセル境界線のフォーマット設定

次のリクエストは、セル境界線を紺色の幅 2 ピクセルで設定し、破線長さ 4 ピクセル、破線間隔 2 ピクセルで破線スタイルを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {"cellBorder": {
    "width": 2,
    "color": "navy",
    "dash": "4 2"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



マトリックスセルのセルパディングの制御

"layout": "cellpadding" プロパティを使用して、マトリックスセルの上下左右のマージンに追加するブランク (パディング) をピクセル数で指定することができます。この方法は、分割線を非表示にした場合に特に有効です。

参照　セルパディングの制御

```
matrixProperties":
{
 "layout":
          {"cellPadding":
             {
             "top": number,
             "bottom": number,
             "left": number,
             "right": number
             }
          }
}
```

説明

`"top": number`
　マトリックスセルの上側のパディングをピクセル数で定義します。デフォルト値は 0 (ゼロ) です。

`"bottom": number`
　マトリックスセルの下側のパディングをピクセル数で定義します。デフォルト値は 0 (ゼロ) です。

`"left": number`
　マトリックスセルの左側のパディングをピクセル数で定義します。デフォルト値は 0 (ゼロ) です。

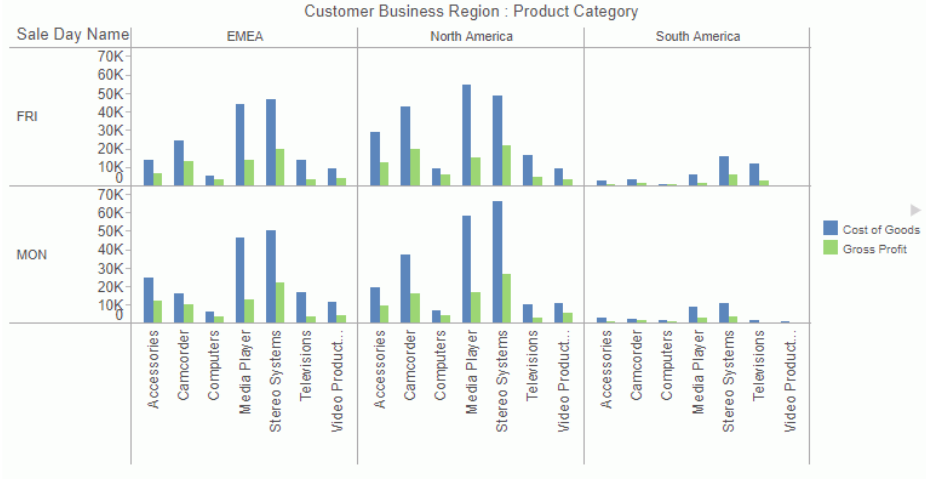
`"right": number`
　マトリックスセルの右側のパディングをピクセル数で定義します。デフォルト値は 0 (ゼロ) です。

例　マトリックスセルのパディング設定

次のリクエストは、マトリックスグラフを生成し、デフォルト設定のパディングを使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。



次のリクエストでは、セルの上側に 10 ピクセル、下側に 20 ピクセル、左側に 5 ピクセル、右側に 10 ピクセルのパディングを追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties":
 {
    "chartType": "bar",
    "layout": {
          "cellPadding":
            {
             "top": 10,
             "bottom": 20,
             "left": 5,
             "right": 10
            }
           }
 }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

マトリックスグラフの最小セルサイズの制御

マトリックスグラフでは、セルの幅と高さに対する最小値を指定することができます。また、"auto" を適用すると、グラフエンジンが読みやすい最小サイズのセルを生成します。

構文　マトリックスグラフの最小セルサイズの制御

```
matrixProperties: {
       "minCellSize": {"width": number, "height": number}}
```

説明

`"minCellSize"`

セルの最小サイズを定義します。データサイズで乗算されたセルサイズがグラフ領域より大きくなる場合は、スクロールが有効化されます。グラフ領域より小さい場合は、グラフ領域に合わせてセルが拡大します。この値に "auto" を入力し、グラフエンジンが最小サイズを決定するように設定することもできます。

`"width": number`

最小幅をピクセル単位で定義します。

`"height": number`

最小高さをピクセル単位で定義します。

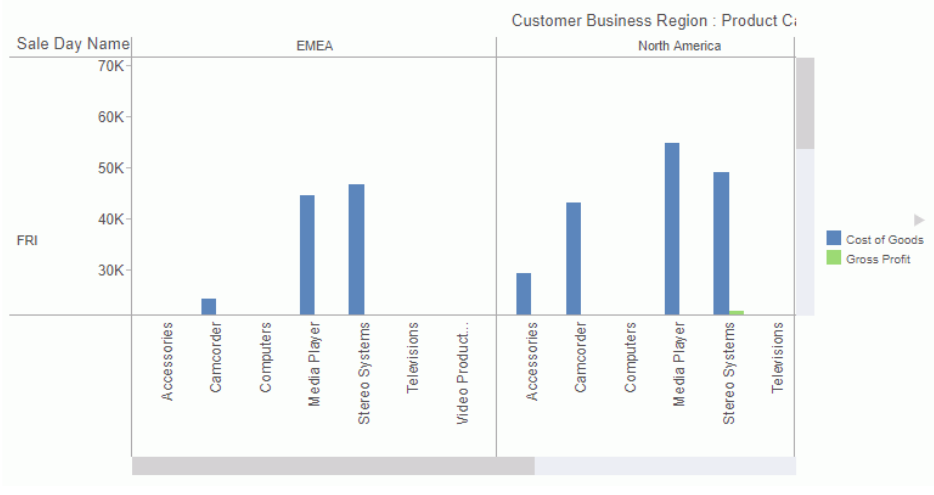
注意：使用可能なスペースを均等に分割するには、{"width":0, "height":0} を使用します。

例　マトリックスグラフの最小セルサイズの設定

次のリクエストは、最小セルサイズを {"width": 300, "height": 300} に定義し、スクロールを有効にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY TIME_DAYNAME
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
WHERE TIME_DAYNAME EQ 'FRI' OR 'MON'
WHERE BUSINESS_REGION NE 'Oceania'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=TIME_DAYNAME, BUCKET=row, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=GROSS_PROFIT_US, BUCKET=y-axis, $
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis, $
*GRAPH_JS
"matrixProperties": {
    "minCellSize": {"width": 300, "height": 300}}
*END
ENDSTYLE
END
```

下図は、スクロールしていない状態の出力結果を示しています。



下図は、スクロールバーをドラッグし、列 North America および行 MON を表すセルのグラフを完全に表示した状態の出力結果を示しています。

マトリックスグラフの数値軸範囲の設定

axis "min" および "max" プロパティを使用して、マトリックスグラフの数値軸の値範囲を、マトリックス行またはマトリックス列のそれぞれで独立して設定することができます。この範囲は、特定の最小値と最大値に設定することも、行または列のそれぞれで独立して自動計算することも可能です。

構文　マトリックスグラフの数値軸範囲の設定

縦方向のグラフでは、Y 軸範囲が各マトリックス行に適用され、X 軸範囲が各マトリックス列に適用されます。横方向のグラフでは、Y 軸範囲が各マトリックス列に適用され、X 軸範囲が各マトリックス行に適用されます。

自動計算の軸の値については、次のプロパティを使用します。

```
{
 "axisname": {
        "min": "autoPerRow",
        "max": "autoPerRow"
        }
}
```

説明

`"axisname"`
　数値軸の名前です。"yaxis" または "xaxis" を指定します。

```
"min": "autoPerRow"
```
各行または各列固有の軸の最小値を自動計算します。

```
"max": "autoPerRow"
```
各行または各列固有の軸の最大値を自動計算します。

特定の軸の値を指定する場合は、次のプロパティを使用します。

```
{
 "axisname": {
        "min": [number, number, number, ...],
        "max": [number, number, number, ...]
        }
}
```

説明

```
"axisname"
```
数値軸の名前です。"yaxis" または "xaxis" を指定します。

```
"min": [number, number, number, ...]
```
マトリックスグラフの各軸の最小値を指定します。数値軸が Y 軸である場合、配列の 1 つ目の要素はマトリックスの 1 行目の Y 軸最小値、2 つ目の要素はマトリックスの 2 行目の Y 軸最小値となります (以下同様)。要素が Null 値の場合 (2 つの連続したカンマ (,,) を含む配列、カンマ (,) で開始される配列、またはマトリックス行または列より要素の数が少ない配列)、グラフエンジンからこれらの軸の値が提供されます。配列のエントリ数がマトリックス行数より多い場合、余分な数値は無視されます。

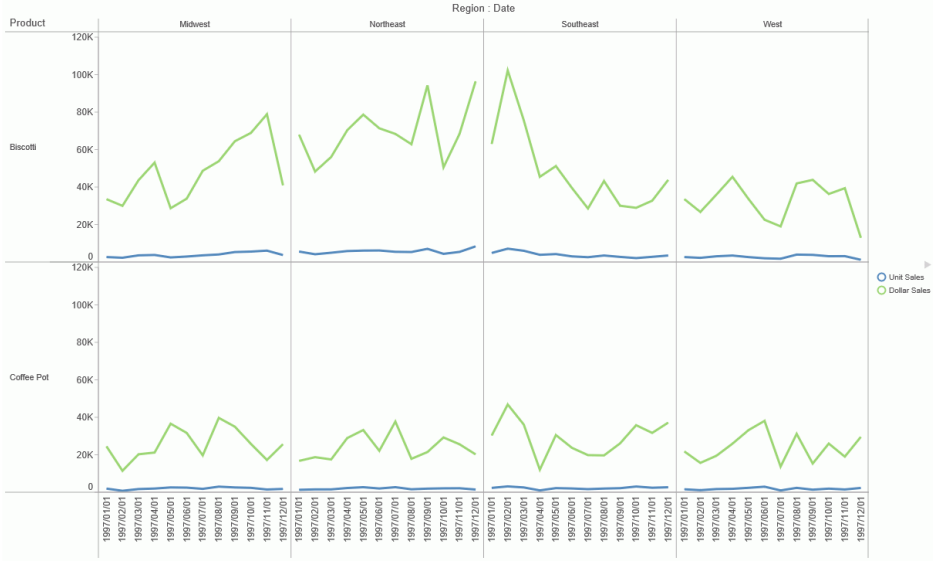
```
"max": [number, number, number, ...]
```
マトリックスグラフの各軸の最大値を指定します。数値軸が Y 軸である場合、配列の 1 つ目の要素はマトリックスの 1 行目の Y 軸最大値、2 つ目の要素はマトリックスの 2 行目の Y 軸最大値となります (以下同様)。要素が Null 値の場合 (2 つの連続したカンマ (,,) を含む配列、カンマ (,) で開始される配列、またはマトリックス行または列より要素の数が少ない配列)、グラフエンジンからこれらの Y 軸の値が提供されます。配列のエントリ数がマトリックス行数より多い場合、余分な数値は無視されます。

例　マトリックスグラフでのY軸最小値および最大値の設定

次のリクエストは、2行のマトリックス行を生成し、デフォルト設定のY軸範囲を使用します。

```
GRAPH FILE GGSALES
SUM UNITS DOLLARS
BY PRODUCT
BY REGION
BY DATE
WHERE PRODUCT EQ 'Biscotti' OR 'Coffee Pot'
WHERE DATE GE '19970101'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=PRODUCT, BUCKET=row, $
TYPE=DATA, COLUMN=REGION, BUCKET=column, $
TYPE=DATA, COLUMN=UNITS, BUCKET=y-axis, $
TYPE=DATA, COLUMN=DOLLARS, BUCKET=y-axis, $
TYPE=DATA, COLUMN=DATE, BUCKET=x-axis, $
ENDSTYLE
END
```

出力結果は次のとおりです。両方のY軸の範囲が0 (ゼロ) から120,000までに設定されています。



次のリクエストでは、2 行のマトリックス行を生成し、各行の Y 軸に対して最小値と最大値を設定します。

```
GRAPH FILE GGSALES
SUM UNITS DOLLARS
BY PRODUCT
BY REGION
BY DATE
WHERE PRODUCT EQ 'Biscotti' OR 'Coffee Pot'
WHERE DATE GE '19970101'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=PRODUCT, BUCKET=row, $
TYPE=DATA, COLUMN=REGION, BUCKET=column, $
TYPE=DATA, COLUMN=UNITS, BUCKET=y-axis, $
TYPE=DATA, COLUMN=DOLLARS, BUCKET=y-axis, $
TYPE=DATA, COLUMN=DATE, BUCKET=x-axis, $
*GRAPH_JS
"yaxis":
{
"min": [800, 600],
"max": [100000, 50000]
}
ENDSTYLE
END
```

出力結果は次のとおりです。1 行目の Y 軸範囲は 800 から 100,000、2 行目の Y 軸範囲は 600 から 50,000 に設定されています。

次のリクエストでは、2 行のマトリックス行を生成し、各行の Y 軸に対して "autoPerRow" を使用します。

```
GRAPH FILE GGSALES
SUM UNITS DOLLARS
BY PRODUCT
BY REGION
BY DATE
WHERE PRODUCT EQ 'Biscotti' OR 'Coffee Pot'
WHERE DATE GE '19970101'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=PRODUCT, BUCKET=row, $
TYPE=DATA, COLUMN=REGION, BUCKET=column, $
TYPE=DATA, COLUMN=UNITS, BUCKET=y-axis, $
TYPE=DATA, COLUMN=DOLLARS, BUCKET=y-axis, $
TYPE=DATA, COLUMN=DATE, BUCKET=x-axis, $
*GRAPH_JS
"yaxis":
{
"min": "autoPerRow",
"max": "autoPerRow"
}
ENDSTYLE
END
```

出力結果は次のとおりです。グラフエンジンが、各行のグラフデータに基づき、各 Y 軸に固有の範囲を計算します。1 行目では、Y 軸範囲は 0 (ゼロ) から 120,000 に設定されています。2 行目では、Y 軸範囲は 0 (ゼロ) から 50,000 に設定されています。

パラボックスのプロパティ (paraboxProperties)

ここでは、パラボックスでアクティブグループを定義するプロパティについて説明します。

構文　　パラボックスのプロパティの制御

```
"paraboxProperties": {
  "activeGroup": "string"
},
```

説明

`"activeGroup": "string"`

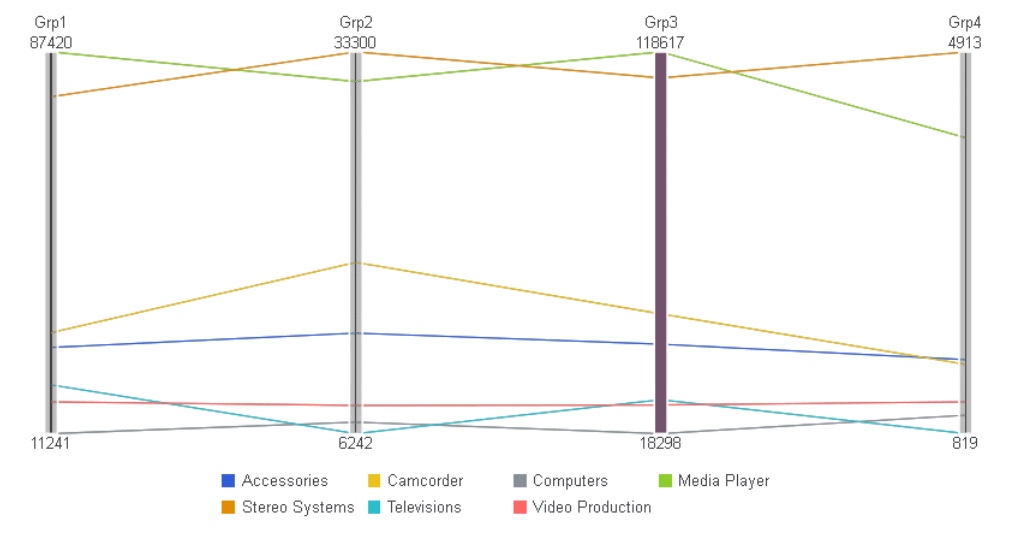
アクティブグループを定義する文字列です (グループのラベル名で指定)。アクティブグループを定義すると、そのグループを通過する線が色分けされます。デフォルト値は undefined です。

例　　パラボックスでのアクティブグループの設定

次のリクエストは、Grp3 グループをアクティブグループに設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US AS 'Cost'
GROSS_PROFIT_US AS 'Profit'
REVENUE_US DISCOUNT_US MSRP_US
AVE.COGS_US AS 'Ave. Cost'
AVE.GROSS_PROFIT_US AS 'Ave. Profit'
MAX.REVENUE_US
MIN.DISCOUNT_US
MDN.MSRP_US AS 'MDN MSRP'
BY PRODUCT_CATEGORY
ON TABLE PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PARABOX
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": true},
"groupLabels": ["Grp1", "Grp2", "Grp3", "Grp4"],
"paraboxProperties": {"activeGroup": "Grp3"}
*END
ENDSTYLE
END
```

出力結果では、Grp3 グループが他のグループと異なる色で表示されています。



円グラフのプロパティ (pieProperties)

ここでは、円グラフの基本的なレイアウトを制御するプロパティについて説明します。

次のコードセグメントは、円グラフのプロパティおよびデフォルト値を示しています。

```
"pieProperties": {
  "sort": {
        "by": undefined,
        "order": undefined
     },
    "holeSize": 0,
 "effect": undefined,
    "rotation": 0,
    "skew": 0,
    "label": {
        "visible": false,
        "font": "10pt Sans-Serif",
        "color": "black"
        },
    "totalLabel": {
        "visible": false,
        "font": "10pt Sans-Serif",
        "color": "black",
        "numberFormat": "auto"
        },
    "feelerLine": (deprecated, use "dataLabels":"feelerLine")


"otherSlice": {
        "threshold": "undefined",
        "legendLabel": "Other",
        "color": "grey",
        "border": {
            "width": 1,
            "color": "transparent",
            "dash": ""
            },
        "showDataValues": true,
        "marker": {
            "shape": "square",
            "border": {
                "width": 0,
                "color": "transparent",
                "dash": ""
                }
            }
        },
```

```
"explodeClick": {
    "enabled": false,
    "duration": 700,
    "distance": 25,
    "limitExplodeCount": false
    }
}
```

注意：円グラフに影響する他のプロパティについては、「シリーズ固有のプロパティ」および「グラフ全体のプロパティ」の章を参照してください。

- 円グラフの円項目の周囲に境界線を描画する。詳細は、436 ページの 「 シリーズライザの境界線の定義 」 を参照してください。
- 円グラフから円項目を分離する (円の中心から外側へ移動)。詳細は、463 ページの「 円グラフからの円項目の分離 」 を参照してください。
- 円グラフから円項目を削除する。詳細は、461 ページの 「 円グラフからの円項目の削除 」 を参照してください。
- 複数円グラフで chartsPerRow プロパティを使用し、1 行に描画する円グラフの個数を定義する。詳細は、223 ページの 「 1 行に表示するグラフ数の制御 」 を参照してください。
- "dataLabels": "position" プロパティを "outside" に設定し、フィーラ線を使用してデータテキストラベルを円項目の外側に描画する。詳細は、445 ページの 「 データテキストラベルの表示およびフォーマット設定 」 を参照してください。
- "dataLabels": "content" プロパティを "%" に設定し、円項目の値を円グラフ全体に対するパーセントで表示する。
- "dataLabels": "feelerLine" プロパティを設定し、フィーラ線のフォーマットを設定する。詳細は、445 ページの 「 データテキストラベルの表示およびフォーマット設定 」 を参照してください。

円項目クリック時のアニメーションの制御

explodeClick プロパティは、ユーザがグラフの円項目をクリックした際のアニメーションの有効と無効の制御、およびアニメーションを定義します。

構文　円項目クリック時のアニメーションの制御

```
"pieProperties": {
   "explodeClick": {
      "enabled": boolean,
      "duration": number,
      "distance": number,
      "limitExplodeCount": boolean   }
}
```

説明

`"enabled": boolean`

有効な値には、次のものがあります。

- true - マウスクリック時に円項目を分離するアニメーションを有効にします。
- false - マウスクリック時に円項目を分離するアニメーションを無効にします。これがデフォルト値です。

`"duration": number`

アニメーションの継続時間をミリ秒で指定します (1000 ミリ秒は 1 秒)。値を小さくすると、アニメーションの動作が速くなります。値を大きくすると、アニメーションの動作が遅くなります。デフォルト値は 700 です。

`"distance": number`

クリック時に円グラフから円項目を分離する距離をピクセル単位で指定します。デフォルト値は 25 です。

`"limitExplodeCount": boolean`

有効な値には、次のものがあります。

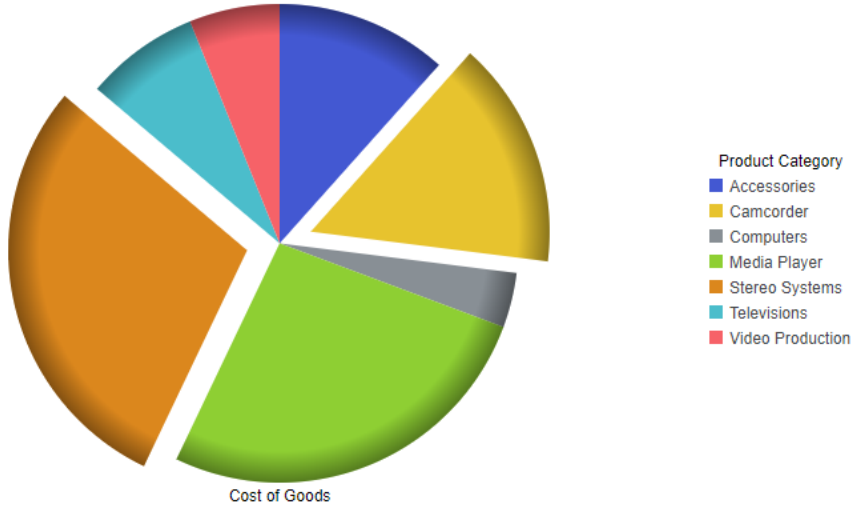
- true - 分離する円項目を 1 つの円項目のみに限定します。
- false - 複数の円項目を分離可能にします。これがデフォルト値です。

例　　円項目クリック時のアニメーションの制御

次のリクエストでは、explodeClick プロパティを有効にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"pieProperties": {
   "explodeClick": {"enabled": true}}
*END
ENDSTYLE
END
```

出力結果では、円項目がクリックされ、2 つの円項目が分離されています。



分離した円項目を元の位置に戻すには、その円項目を再度クリックします。

リンググラフの描画 (中央に穴の開いた円グラフ)

holeSize プロパティは、円グラフの中央に描画する透明な穴のサイズを定義します。

注意：LOOKGRAPH パラメータの PIERING 値を使用して、中央に穴の開いた円グラフを描画することもできます。LOOKGRAPH パラメータ設定についての詳細は、40 ページの「 グラフタイプの制御 」 を参照してください。

構文　円グラフの穴サイズの制御

```
"pieProperties": {
   "holeSize": size   }
```

説明

`"holeSize": size`

穴のサイズを定義します。有効な値には、次のものがあります。

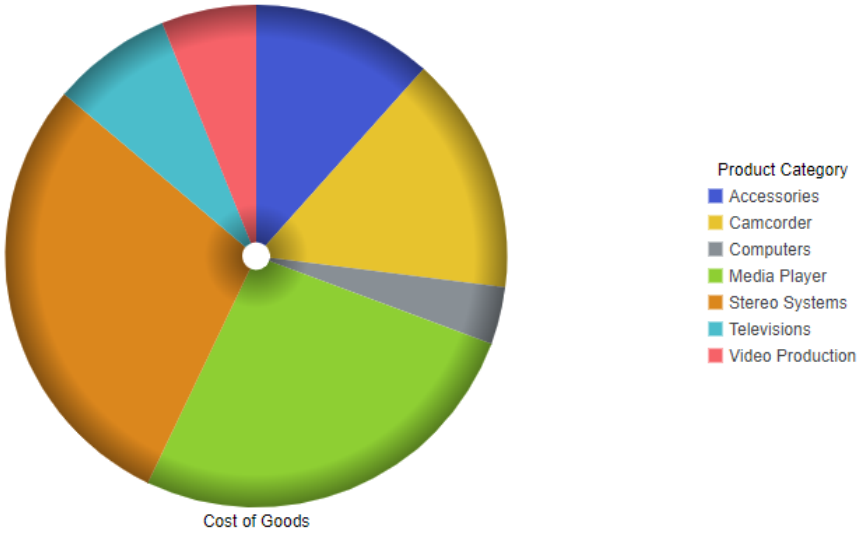
- サイズを定義する数値をピクセル単位で指定します。
- 円半径に対するサイズの割合をパーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます (例、"10%")。デフォルト値は 0 (ゼロ) です。

例　円グラフの穴サイズの設定

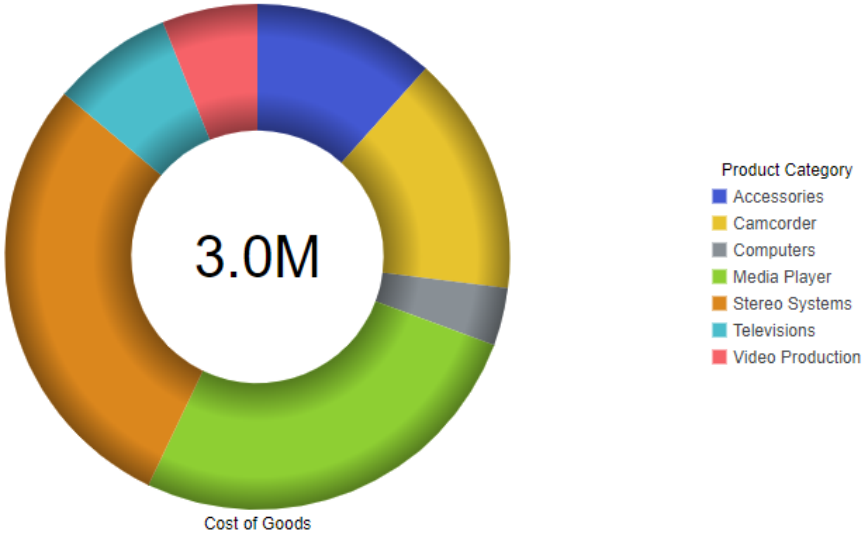
次のリクエストは、円グラフを生成し、中央に 10 ピクセルの穴を描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"pieProperties": {
   "holeSize": 10}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



穴のサイズを "50%" に変更すると、下図のグラフが生成されます。



リンググラフ円項目端部への丸みの適用

pieProperties:effect プロパティを使用して、リンググラフの円項目の端部に丸みを適用することができます。

構文　リンググラフ円項目端部への丸みの適用

```
"pieProperties": {
   "effect": "string"
           }
```

説明

`"effect": "string"`

デフォルト設定の円項目を生成する場合、"effect" プロパティは、省略することも、"undefined" にすることもできます。端部に丸みを適用するには、"roundRing" に設定します。

注意：円項目に丸みを設定する場合、次の制限事項が適用されます。

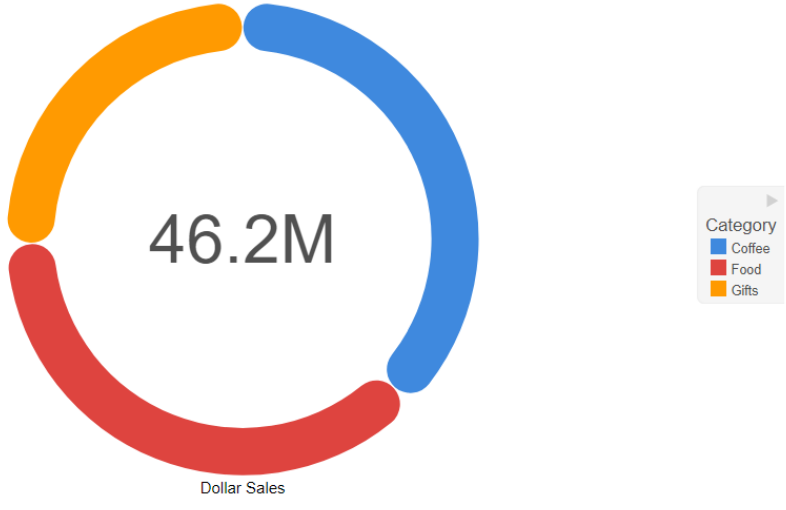
- 穴の最小サイズは 80% です。最小サイズをこれより小さい値に設定した場合、80% が使用されます。
- 外側ラベルはサポートされません。
- 円項目の切り離しはサポートされません。
- 複数リングはサポートされません。
- 2.5D グラフはサポートされません。
- 動的サイズ変更はサポートされません。

例　端部の丸み付きリンググラフの生成

次のリクエストは、デフォルト設定の穴のサイズ (80%) のリンググラフを生成します。

```
GRAPH FILE ibisamp/ggsales
SUM DOLLARS
BY CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENID_Default.sty,$
TYPE=DATA, COLUMN=N2, BUCKET=measure, $
TYPE=DATA, COLUMN=N1, BUCKET=color, $
*GRAPH_JS
"pieProperties": {
    "holeSize": "80%",
    "effect": "roundRing"
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



円グラフのラベルのフォーマット設定

label プロパティは、円グラフのラベルの表示およびフォーマットを制御します。

構文　円グラフのラベルのフォーマット設定

```
"pieProperties": {
   "label": {
      "visible": boolean,
      "font": "string",
      "color": "string"

   }
}
```

説明

`"visible": boolean`

有効な値には、次のものがあります。

- true - ラベルを表示します。これがデフォルト値です。
- false - ラベルを表示しません。

`"font": "string"`

円グラフのラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列で円グラフのラベルの色を定義します。デフォルト値は "black" です。

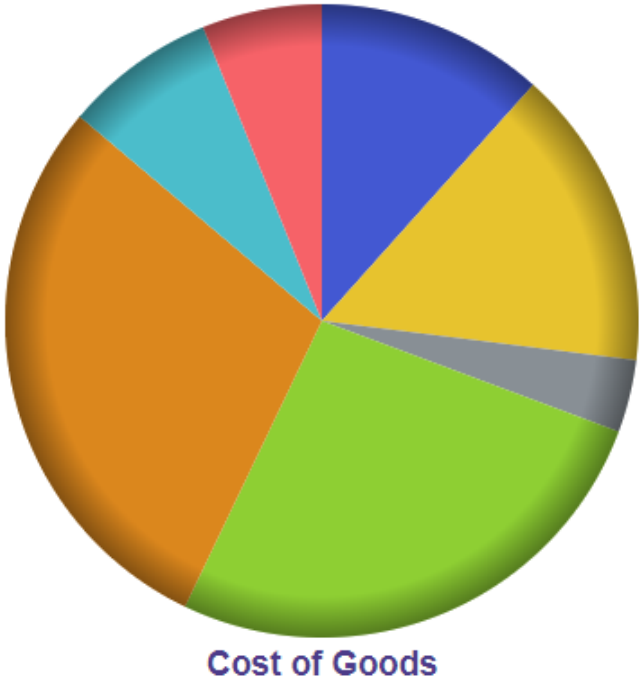
色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　円グラフのラベルのフォーマット設定

次のリクエストは、円グラフのラベルの色を濃い青灰色に、フォントを 14pt、Sans Sefif、太字に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"legend": {"visible": false},
"pieProperties": {"label": {
   "visible": true, "font": "Bold 14pt Sans-Serif", "color": "DarkSlateBlue"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



同心円グラフの生成

multiRing プロパティは、同心円グラフを有効または無効にします。同心円グラフは、積み上げ棒グラフに類似しています。円グラフの各円項目は、それぞれ異なる色の同心円リングに分割されます。このグラフタイプでは、通常の円グラフと同一のデータが使用されます。

構文　同心円グラフの生成

```
"pieProperties": {
   "multiRing": boolean   }
```

説明

`"multiRing": boolean`

有効な値には、次のものがあります。

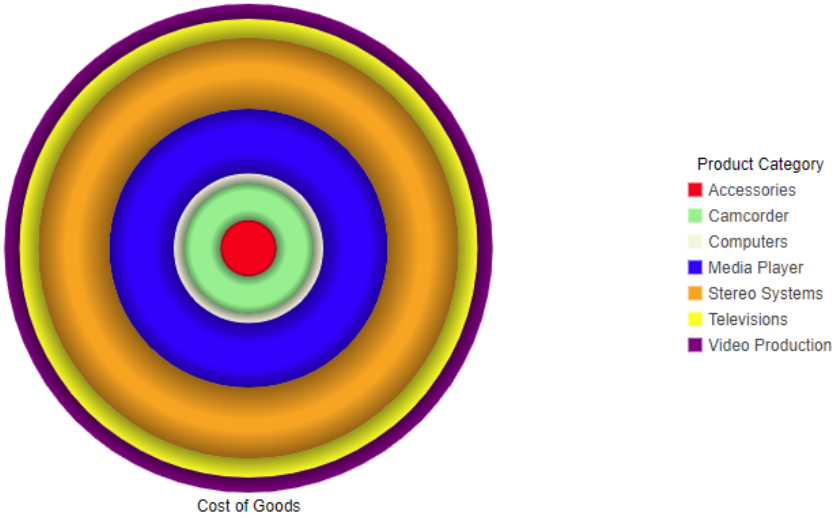
- true - 各円項目を同心円リングで表現した円グラフを常に描画します。
- false - 通常の円グラフを描画します。デフォルト値は false です。

例　同心円グラフの生成

次のリクエストは、同心円グラフを生成します。

```
GGRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"pieProperties": {"multiRing": true},
"series": [
   {"series": "all", "border": {"width": 1, "color": "grey"}},
   {"series": 0, "color": "red"},
   {"series": 1, "color": "lightgreen"},
   {"series": 2, "color": "beige"},
   {"series": 3, "color": "blue"},
   {"series": 4, "color": "orange"},
   {"series": 5, "color": "yellow"},
   {"series": 6, "color": "purple"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



指定した値未満の円項目の統合

otherSlice プロパティは、指定した値未満の円項目の表示およびフォーマットを制御します。otherSlice 値を定義すると、そのカテゴリに属する円項目がすべて統合され、1 つの円項目になります。

また、その円項目をクリックしたときに [その他] 円項目にドリルダウンする interaction プロパティを定義することもできます。詳細は、769 ページの 「 その他の機能 」 を参照してください。

構文　　指定した値未満の円項目のフォーマット設定

```
"pieProperties": {
   "otherSlice": {
      "threshold": value,
      "legendLabel": "string",
      "color": "color",
      "border": {"width": number, "color": "string", "dash": "string"}
      "showDataValues": boolean,
      "marker": {
         "shape": "string",
         "border": {"width": number, "color": "string", "dash": "string"}
      }
   }
}
```

説明

`"threshold": value`

値を定義します。この値未満のデータが [その他] 円項目カテゴリに統合されます。有効な値には、次のものがあります。

- 数値 (絶対データ値) - しきい値が数値の場合、このしきい値未満のすべての値が [その他] 円項目に統合されます。
- パーセント文字列 - パーセント記号 (%) を含めて二重引用符 (") で囲みます。しきい値がパーセント文字列の場合 (例、"2%")、円全体に対する割合がこの値未満のすべての円項目が [その他] 円項目に統合されます。
- "top n" 文字列 (例、"top 2") - n は値の個数を表し、この個数の値は [その他] 円項目に統合されません。
- undefined - [その他] 円項目を無効にします。これがデフォルト値です。

`"legendLabel": "string"`

[その他] 円項目のラベルを識別する文字列です。この文字列が、凡例のラベルとして表示されます。デフォルト値は "Other" です。

`"color": "color"`

[その他] 円項目の色を定義します。有効な値には、次のものがあります。

- 色のグラデーションを定義する JSON オブジェクト。
- グラデーション文字列、色名、数値指定の文字列で定義する色。デフォルト値は "grey" です。

色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"border":`

[その他] 円項目の境界線のプロパティを定義します。

`"width": number`

[その他] 円項目の境界幅をピクセル単位で指定します。デフォルト値は 1 です。

`"color": "string"`

色名または数値指定の文字列で [その他] 円項目の境界線の色を定義します。デフォルト値は "transparent" です。

"dash": "string"

[その他] 円項目の境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"showDataValues": boolean

有効な値には、次のものがあります。

- true - [その他] 円項目のデータテキストラベルを表示します。
- false - [その他] 円項目のデータテキストラベルを表示しません。これがデフォルト値です。

"marker":

[その他] 円項目のマーカーのプロパティを定義します。このマーカーは、凡例領域に表示されます。

"shape": "string"

[その他] 円項目のマーカー形状を定義する文字列です。このマーカー形状は、凡例領域に表示されます。サポートされる形状には、arrow、bar、circle、cross、diamond、fiveStar、hexagon、hourglass、house、pin、pirateCross、plus、rectangle、sixStar、square、thinPlus、tick、circlePlus、circleMinus、triangle があります。デフォルト値は "square" です。bar、cross、circlePlus、circleMinus、tick マーカーを使用する場合は、境界幅と色を指定する必要があります。

"border":

マーカー境界線のプロパティを定義します。

"width": number

マーカー境界線の幅をピクセル単位で指定します。デフォルト値は 1 です。

"color": "string"

色名または数値指定の文字列でマーカーの境界色を定義します。デフォルト値は "transparent" です。

"dash": "string"

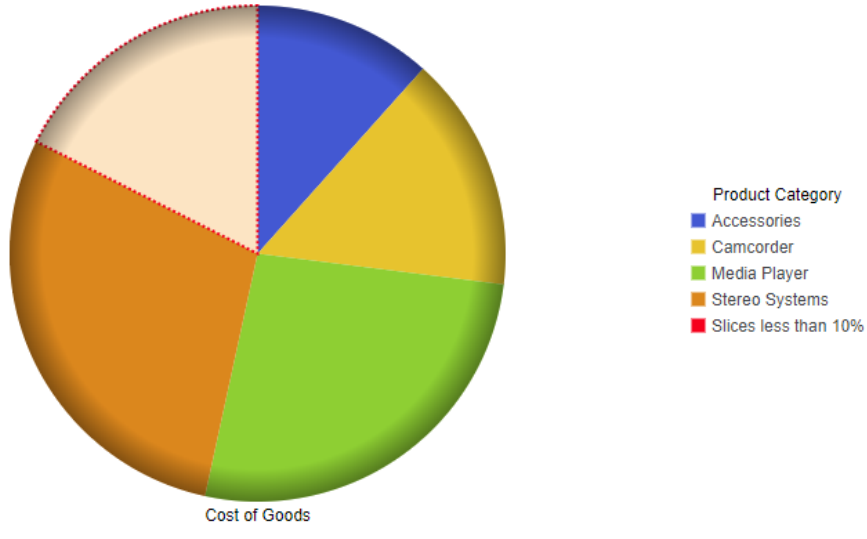
マーカー境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　円グラフでの「その他」円項目の生成

次のリクエストは、値が 10% 未満のすべての円項目を [その他] 円項目に統合します。[その他] 円項目の凡例ラベルには、「Slices less than 10%」を指定します。[その他] 円項目の色はビスクで、境界は赤色の破線です。凡例に表示するマーカー形状は赤色の正方形です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
*GRAPH_JS
"pieProperties": {
   "totalLabel": {"visible": false},
      "otherSlice": {
         "threshold": "10%", "legendLabel": "Slices less than 10%", "color": "bisque",
         "border": {"width": 2, "color": "red", "dash": "2 2"},
         "marker": {"shape": "square", "width": 1, "color": "red"}}}
*END
ENDSTYLE
END
```

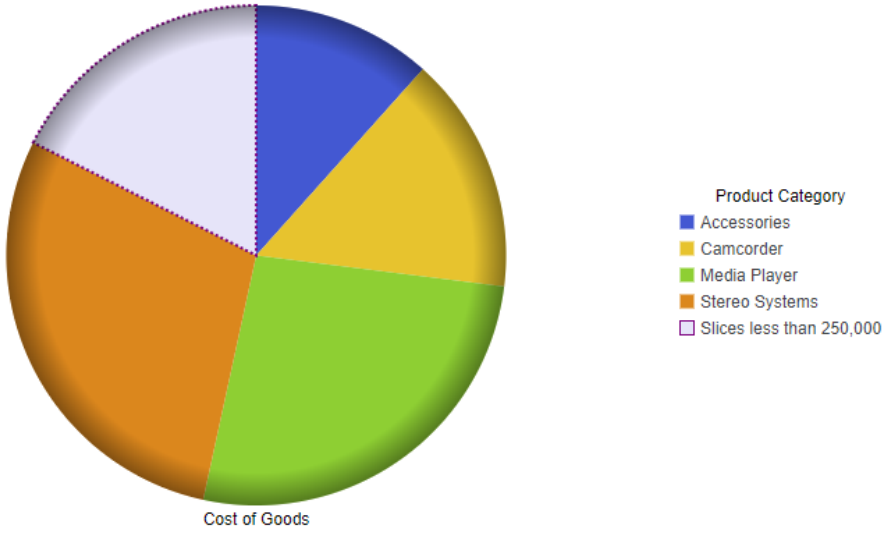
出力結果は次のとおりです。



次のリクエストでは、250,000 未満のすべての値を [その他] 円項目に統合します。[その他] 円項目の色は薄紫色で、境界線は紫色の破線です。凡例での [その他] 円項目のマーカーは薄紫色の四角形で、境界線は紫色です。凡例テキストは「Slices less than 250,000」です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"pieProperties": {
   "totalLabel": {"visible": false},
   "otherSlice": {"threshold": 250000, "legendLabel": "Slices less than 250,000",
      "color": "lavender", "border": {"width": 2, "color": "purple", "dash": "2 2"},
      "marker": {"shape": "triangle",
         "border": {"width": 1, "color": "purple"}}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



円グラフの回転

rotation プロパティは、指定した角度で円グラフを回転します。

構文　円グラフの回転

```
"pieProperties": {
   "rotation": number   }
```

説明

`"rotation": number`

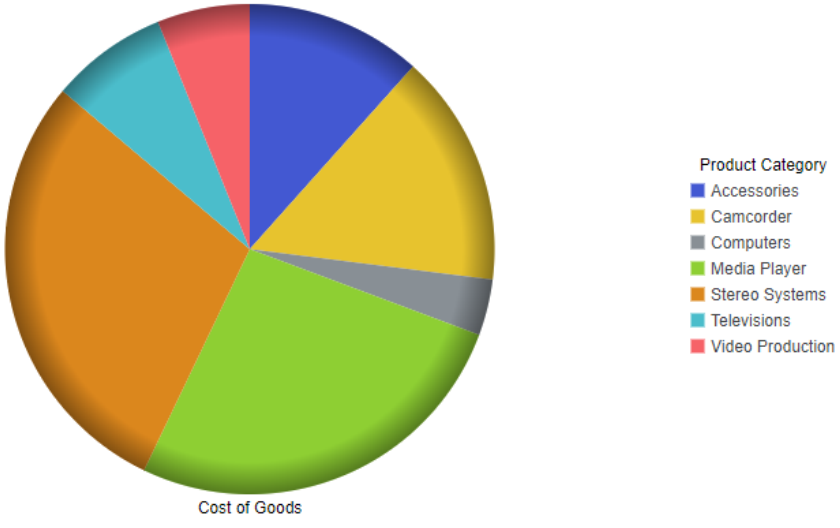
円グラフの回転角度を 0 (ゼロ) から 359 までの値で指定します。デフォルト値は 0 (ゼロ) です。

例　円グラフの回転

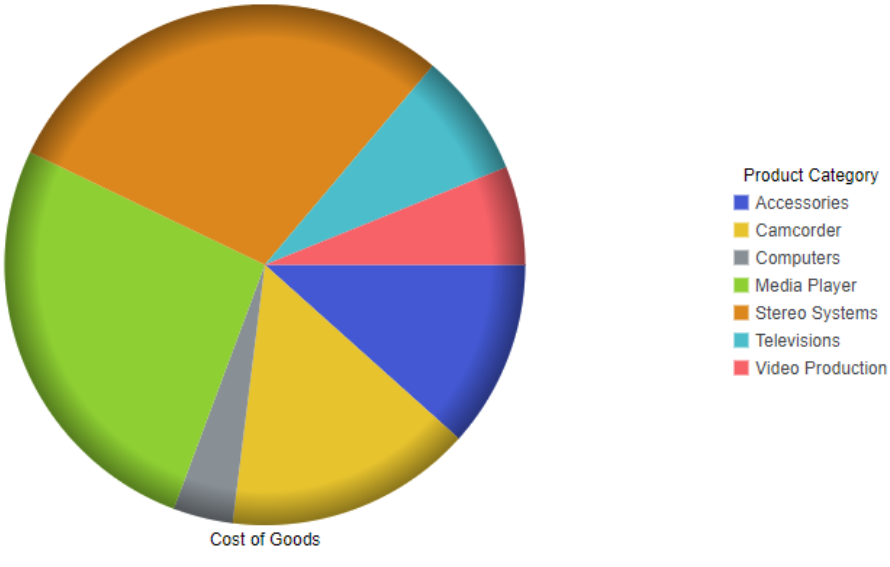
次のリクエストは、回転なし ({"rotation":0}) の円グラフを生成し、青色の円項目が 0 (ゼロ) 度で開始されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"pieProperties": {"rotation": 0
   }
*END
ENDSTYLE
END
```

出力結果では、青色の円項目が、円グラフの 0 度の位置 (時計の針の 12 時の位置) から開始されています。



rotation 値を 90 に変更すると、下図のグラフが生成されます。このグラフでは、青色の円項目が、円グラフの 90 度の位置 (時計の針の 3 時の位置) から開始されています。



円グラフの傾斜

円グラフに奥行きが適用されている場合、skew プロパティで円グラフの傾斜を制御します。奥行きの適用についての詳細は、227 ページの 「 グラフへの奥行きの適用 」 を参照してください。

構文　円グラフの傾斜

```
"pieProperties": {
   "skew": number   }
```

説明

`"skew": number`

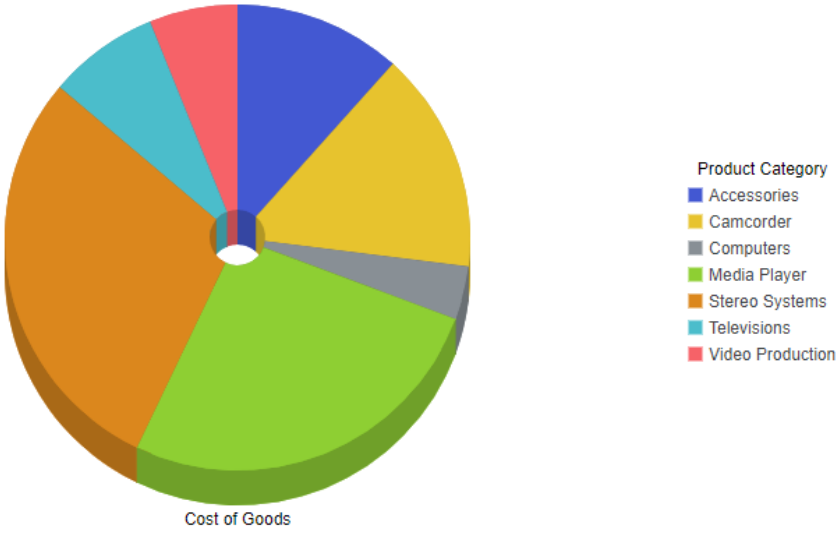
0 (ゼロ) から 100 までの数値です。デフォルト値は 0 (ゼロ) です。

例　円グラフの傾斜

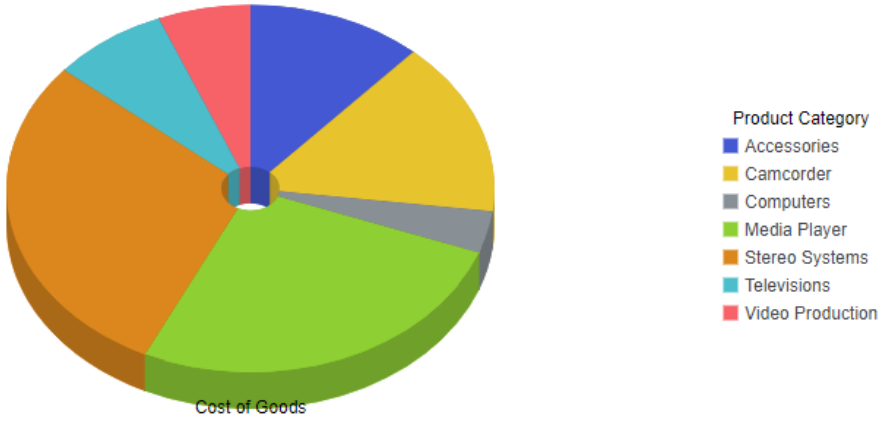
次のリクエストは、傾斜なしの円グラフ ({"skew":0}) を生成し、奥行きを 25 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"depth": 25,
"pieProperties": {"holeSize": 20, "skew": 0}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



skew 値を 25 に変更すると、下図のようなグラフが生成されます。



円グラフの円項目のソート

円項目を、項目のメジャー値またはラベルの昇順または降順でソートすることができます。複数の円グラフが生成される場合、ソートはグラフごとに個別に行えます。凡例の順序は、円項目のソート表示による影響を受けません。

構文　円グラフの円項目のソート

```
"pieProperties": {
     "sort": {
        "by": "string",
        "order": "string"
     }
 }
```

説明

`"by": "string"`

ソートに使用する値を定義します。有効な値には、次のものがあります。

- **"value"** - 各円項目を、項目のメジャー値でソートします。
- **"label"** - 各円項目を、項目のラベルの値でソートします。

`"order": "string"`

ソート順を定義します。有効な値には、次のものがあります。

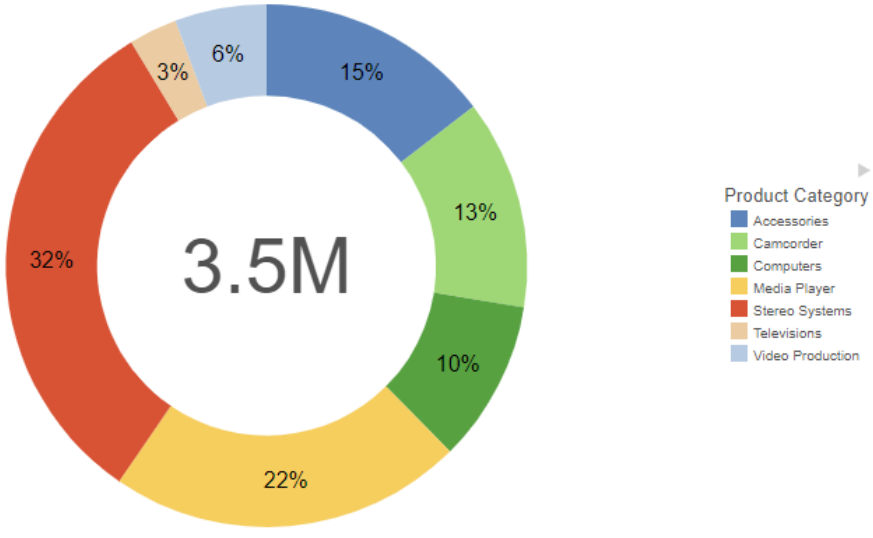
- **"ascending"** - 円項目を値の昇順でソートします。
- **"descending"** - 円項目を値の降順でソートします。

例　円グラフの円項目のソート

次のリクエストは、円項目を生成しますが、円項目のソート順は指定しません。

```
GRAPH FILE wf_retail_lite
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=color, $
TYPE=DATA, COLUMN=N2, BUCKET=measure, $
*GRAPH_JS_FINAL
"dataLabels": {"visible": true, "font": "12pt Sans-Serif"},
"pieProperties": {
    "holeSize": "65%",
    "label": {"visible": false}
},
*END
ENDSTYLE
END
```

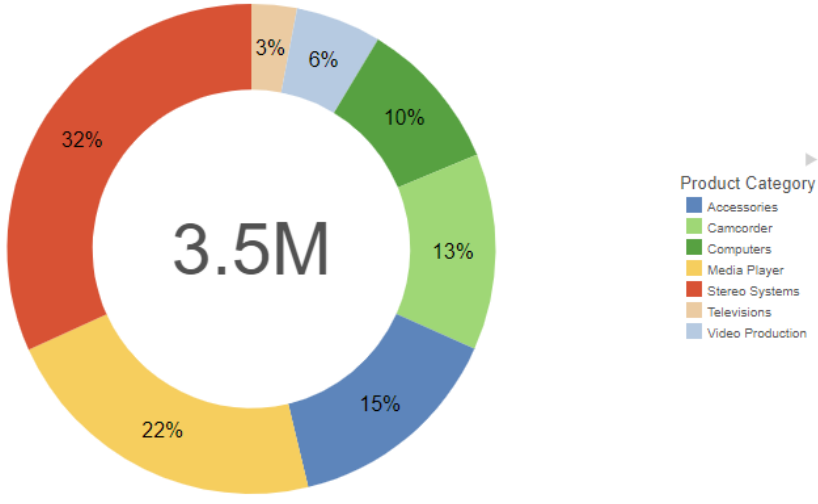
出力結果は次のとおりです。円項目は、製品区分の値の順序でソートされ、右上の円項目から時計回りに順に表示されます。



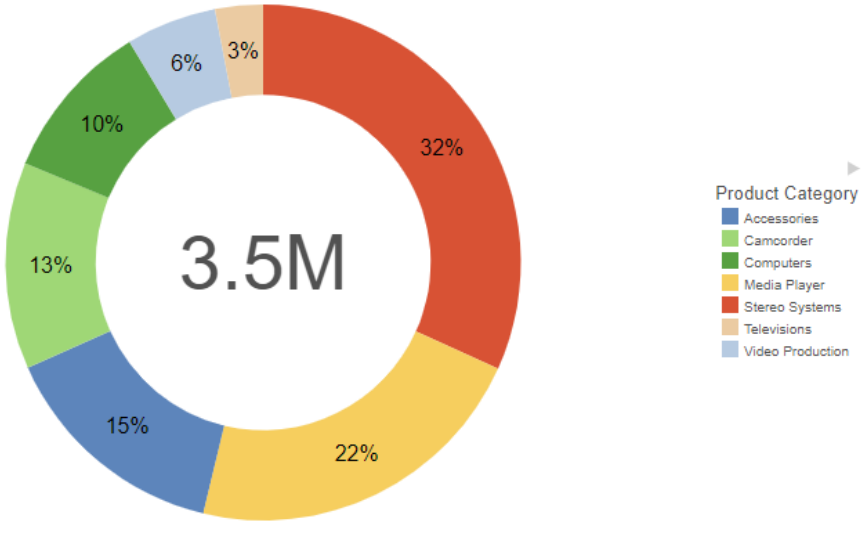
次のリクエストでは、円項目をメジャー値の昇順でソートします。

```
GRAPH FILE wf_retail_lite
SUM QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=color, $
TYPE=DATA, COLUMN=N2, BUCKET=measure, $
*GRAPH_JS_FINAL
"dataLabels": {"visible": true, "font": "12pt Sans-Serif"},
"pieProperties": {
    "sort": {
        "by": "value",
        "order": "ascending"
    },
    "holeSize": "65%",
    "label": {"visible": false}
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。円項目は、メジャー値の昇順でソートされ、右上の円項目から時計回りに順に表示されます。



ソート順を "descending" に変更した場合の出力結果は次のとおりです。円項目は、メジャー値の降順でソートされ、右上の円項目から時計回りに順に表示されます。



円グラフの合計ラベルのフォーマット設定

totalLabel プロパティは、円グラフの合計ラベルの表示およびフォーマットを制御します。合計ラベルは、グラフの中央に表示される合計値です。

構文　円グラフの合計ラベルのフォーマット設定

```
"pieProperties": {
   "totalLabel": {
      "visible": boolean,
      "font": "string",
      "color": "string",
      "numberFormat": numformat
      "fontSize": "string"
   }
}
```

説明

`"visible": boolean`

有効な値には、次のものがあります。

- true - ラベルを表示します。
- false - ラベルを表示しません。 これがデフォルト値です。
- "auto" - 円グラフに合計ラベルが収まるサイズの穴がある場合に、合計ラベルを表示します。可能な場合はフォントサイズが拡大縮小されます。

`"font": "string"`

ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "10pt Sans-Serif" です。

`"color": "string"`

色名または数値指定の文字列でラベルの色を定義します。デフォルト値は "black" です。

色の指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"numberFormat": numformat`

JSON オブジェクト、フォーマット文字列、またはユーザ定義関数として指定することができます。デフォルト値は "auto" です。autoNumberFormats を使用して、円グラフの合計ラベルのすべてに数値フォーマットを適用することもできます。数値フォーマットについての詳細は、86 ページの 「 数値のフォーマット設定 」 を参照してください。

"fontSize": "string"

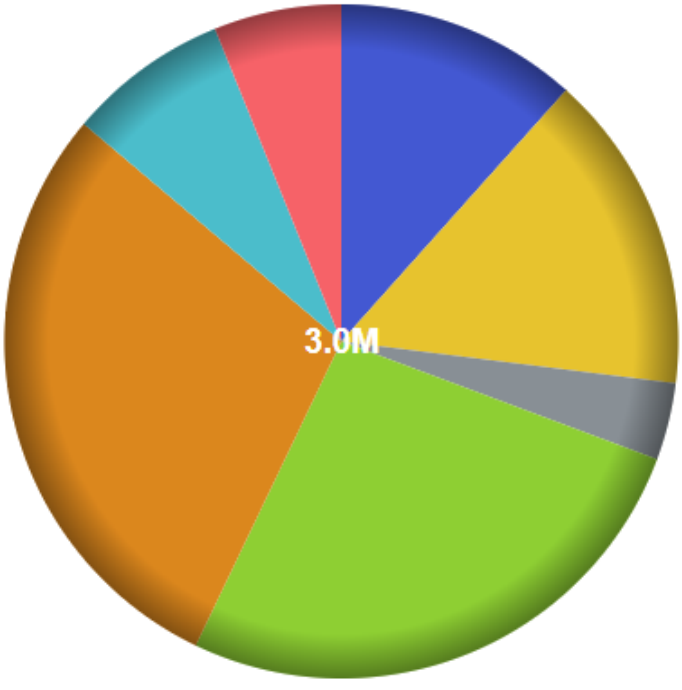
ラベルのサイズを定義する文字列です。リンググラフのサイズにかかわらず、"font" 関数で設定された固定値を上書きします。

例　円グラフの合計ラベルのフォーマット設定

次のリクエストは、円グラフを生成し、合計ラベルを白色、太字、14pt Sans-Serif フォントで表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"legend": {"visible": false},
"pieProperties": {
   "label": {"visible": false},
   "totalLabel": {
      "visible": true, "font": "Bold 14pt Sans-Serif", "color": "white"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



極グラフのプロパティ (polarProperties)

ここでは、極グラフおよびレーダーグラフの全体的な外観を制御するプロパティについて説明します。

構文　　極グラフおよびレーダーグラフのプロパティの指定

```
"polarProperties": {
   "straightGridLines": boolean,
   "extrudeAxisLabels": boolean,
   "drawAsArea": boolean},
```

説明

`"straightGridLines": boolean`
罫線のスタイルを定義します。有効な値には、次のものがあります。

- ❏ true - "yaxis": "majorGrid" で指定された罫線を直線で描画します。

- false - "yaxis": "majorGrid" で指定された罫線を円形状に描画します。デフォルト値は false です。

`"extrudeAxisLabels": boolean`

Y 軸ラベルの描画位置を定義します。有効な値には、次のものがあります。

- true - Y 軸ラベルを円形罫線の外側に描画します。
- false - Y 軸ラベルを円形罫線の内側に描画します。デフォルト値は false です。

`"drawAsArea": boolean`

レーダーグラフのみに適用されます。有効な値には、次のものがあります。

- true - 円形の面グラフを描画します。
- false - 円形の折れ線グラフを描画します。デフォルト値は false です。

注意

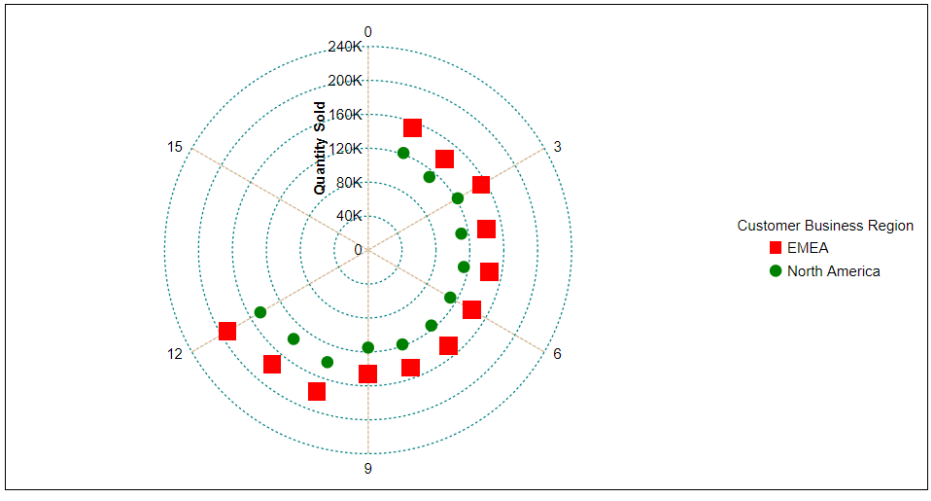
- 極グラフは、円形の散布図です。
- 散布図と同様に、極グラフには各マーカーを描画するための値が 2 つ必要です。
- yaxis プロパティは、極グラフのマーカー周囲の円形罫線を制御します。これらのプロパティは、軸ラベルおよび罫線の外観を制御する際にも使用します。
- xaxis プロパティは、X 軸のラベルと値の外観 (円形罫線の最も外側の周囲)、および X 軸の罫線を制御します。

例　極グラフのプロパティの設定

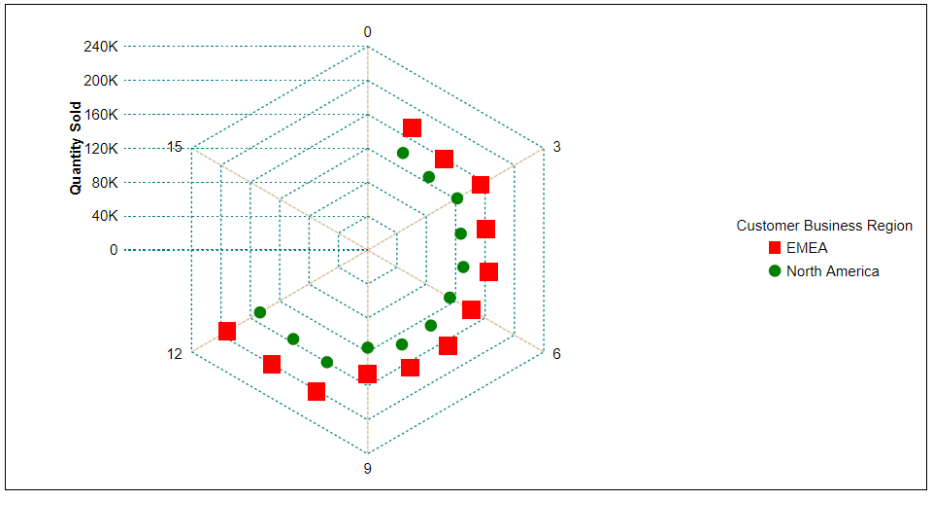
次のリクエストは、デフォルトのプロパティを使用して極グラフを生成します。主罫線は青緑色の破線の円形罫線で、series プロパティでマーカーの形状とサイズを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY BUSINESS_REGION
BY TIME_MTH
WHERE BUSINESS_REGION EQ 'North America' OR 'EMEA'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH POLAR
ON GRAPH SET STYLE *
type=data, column=BUSINESS_REGION, bucket=color, $
type=data, column=TIME_MTH, bucket=x-axis, $
type=data, column=QUANTITY_SOLD, bucket=y-axis, $
*GRAPH_JS
"polarProperties": {
  "straightGridLines": false,
  "extrudeAxisLabels": false
},
"yaxis": {
   "majorGrid": {
   "lineStyle": {"width": 1, "color": "teal", "dash": "2 2"},
   }
},
"xaxis": {
   "majorGrid": {
   "lineStyle": {"width": 1, "color": "tan", "dash": "2 2"},
   }
},
"series": [
    {"series": 0, "marker": {"size": 15, "shape": "square"}},
    {"series": 1, "marker": {"size": 10, "shape": "circle"}},
        ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



straightGridLines を true、extrudeAxisLabels を true に変更すると、下図のグラフが生成されます。

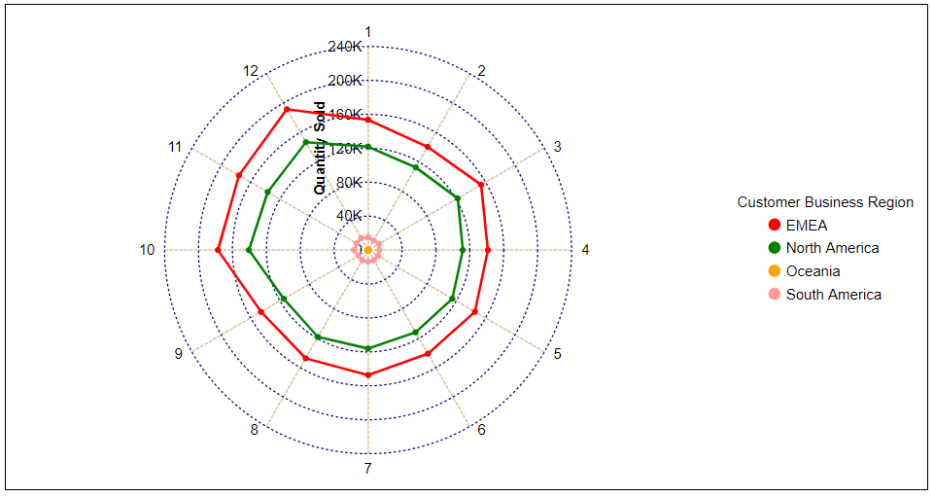


例　レーダーグラフのプロパティの設定

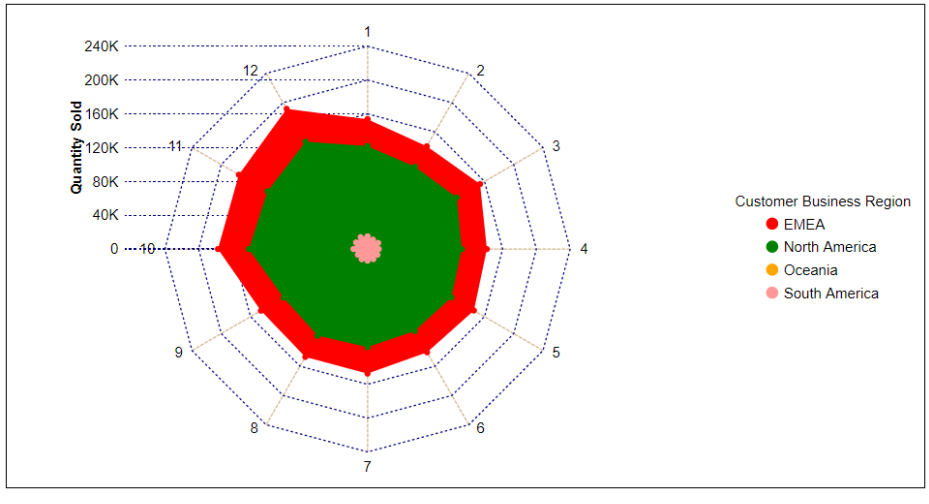
次のリクエストは、デフォルトのプロパティを使用してレーダーグラフを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
BY BUSINESS_REGION
BY TIME_MTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH RADAR
ON GRAPH SET STYLE *
type=data, column=BUSINESS_REGION, bucket=color, $
type=data, column=TIME_MTH, bucket=x-axis, $
type=data, column=QUANTITY_SOLD, bucket=y-axis, $
*GRAPH_JS
"polarProperties": {
"straightGridLines": false,
"extrudeAxisLabels": false,
"drawAsArea": false
},
"yaxis": {
"majorGrid": {
"lineStyle": {"width": 1,"color": "navy","dash": "2 2"},
}
},
"xaxis": {
"majorGrid": {
"lineStyle": {"width": 1,"color": "tan","dash": "2 2"},
}
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



straightGridLines、extrudeAxisLabels、drawAsArea を true に変更すると、下図のグラフが生成されます。



タグクラウドのプロパティ (tagloudProperties)

以下のプロパティは、タグクラウドの描画に使用されるエンジン、ラベルのフォントおよびラベルの最大数を制御します。

次のコードセグメントは、タグクラウドで使用可能なプロパティおよびデフォルト値を示しています。

```
"tagcloudProperties": {
    "engine": "old",
    "maxNumberOfTags": 50,
    "font": "bold 12pt Georgia"
    }
```

構文　タグクラウドのプロパティの設定

```
"tagcloudProperties": {
   "engine": "string",
   "maxNumberOfTags": number,
   "font": "string"
},
```

説明

`"engine": "string"`

使用するタグクラウドエンジンのバージョンを指定します。有効な値には、次のものがあります。

- "new" - 新しいバージョンのタグクラウドエンジンを使用します。
- "old" - 古いバージョンのタグクラウドエンジンを使用します。これがデフォルト値です。

`"maxNumberOfTags": number`

タグクラウドに表示するラベルの最大数です。デフォルト値は 50 です。

`"font": "string"`

ラベルのタイプフェースを定義する文字列です。デフォルト値は "bold 12pt Georgia" です。

注意：フォント指定の文字列にはポイントサイズを含めることができますが、指定した値は使用されません。 ラベルのサイズは、タグクラウドの描画に使用されるデータに基づいて決定されます。

例　タグクラウドのプロパティの設定

次のリクエストは、デフォルトのプロパティを使用してタグクラウドを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD DAYSDELAYED
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TAGCLOUD
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
type=data, column=QUANTITY_SOLD, bucket=size,$
 type=data, column=DAYSDELAYED, bucket=color,$
type=data, column=PRODUCT_CATEGORY, bucket=detail,$
END
```

出力結果は次のとおりです。

次のリクエストでは、新しいバージョンのタグクラウドエンジンを使用して、タグ数を 5 に制限し、フォントを 14pt Algerian に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD DAYSDELAYED
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TAGCLOUD
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
type=data, column=QUANTITY_SOLD, bucket=size,$
 type=data, column=DAYSDELAYED, bucket=color,$
type=data, column=PRODUCT_CATEGORY, bucket=detail,$
*GRAPH_JS
"tagcloudProperties": {
   "engine": "new",
   "maxNumberOfTags": 5,
   "font": "14pt Algerian"}
*END
END
```

出力結果は次のとおりです。

注意：フォント指定の文字列にはポイントサイズを含めることができますが、指定した値は使用されません。 ラベルのサイズは、タグクラウドの描画に使用されるデータに基づいて決定されます。

3D グラフのプロパティ (threedProperties)

ここでは、3D グラフ ("area3D"、"bar3D"、"surface3D" グラフタイプ) の全体的な外観を制御するプロパティについて説明します。

構文　3D グラフのプロパティの設定

```
"threedProperties": {
   "rotate": number,
   "tilt": number,
   "shadeSides": boolean
},
```

説明

`"rotate": number`

3D キューブの回転角度を 0 (ゼロ) から 90 までの数値で定義します。デフォルト値は 40 です。

`"tilt": number`

3D キューブの傾斜角度を 0 (ゼロ) から 90 までの数値で定義します。デフォルト値は 40 です。

`"shadeSides": boolean`

有効な値には、次のものがあります。

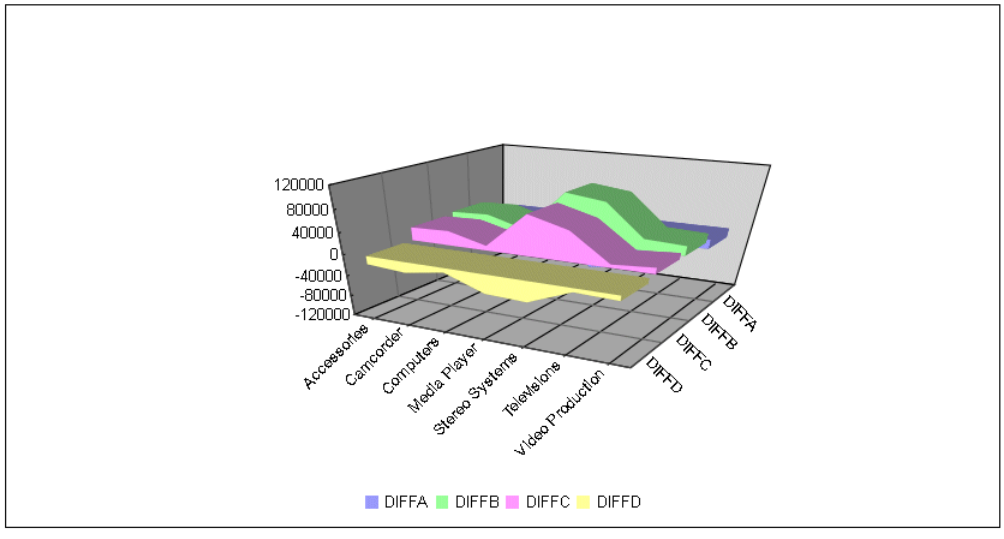
- true - 3D ライザに影を付けます。これがデフォルト値です。
- false - 3D ライザに影を付けません。

例　3D 面グラフのプロパティの設定

次のリクエストは、デフォルトのプロパティを使用して 3D 面グラフを生成します。

```
DEFINE FILE WF_RETAIL_LITE
DIFFA = GROSS_PROFIT_US - REVENUE_US;
DIFFB = REVENUE_US - GROSS_PROFIT_US;
DIFFC = COGS_US - (COGS_US * DISCOUNT_US)/100;
DIFFD = COGS_US - MSRP_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFFA DIFFB DIFFC DIFFD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH 3DAREAS
END
```

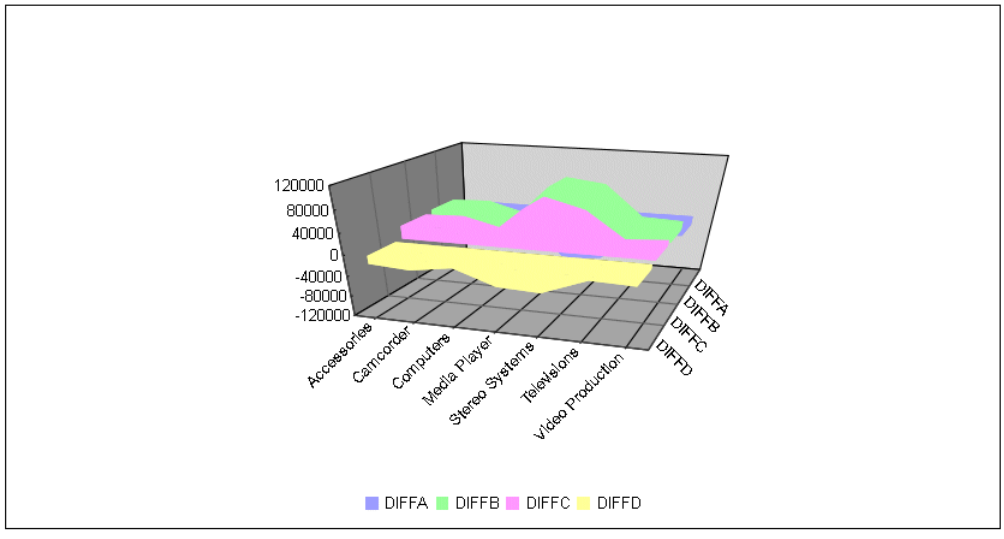
出力結果は次のとおりです。



次のリクエストでは、"rotate" を 20、"tilt" を 30、"shadeSides" を false に設定します。

```
DEFINE FILE WF_RETAIL_LITE
DIFFA = GROSS_PROFIT_US - REVENUE_US;
DIFFB = REVENUE_US - GROSS_PROFIT_US;
DIFFC = COGS_US - (COGS_US * DISCOUNT_US)/100;
DIFFD = COGS_US - MSRP_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFFA DIFFB DIFFC DIFFD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH 3DAREAS
ON GRAPH SET STYLE *
*GRAPH_JS
"threedProperties": {
    "rotate": 20, "tilt": 30, "shadeSides": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

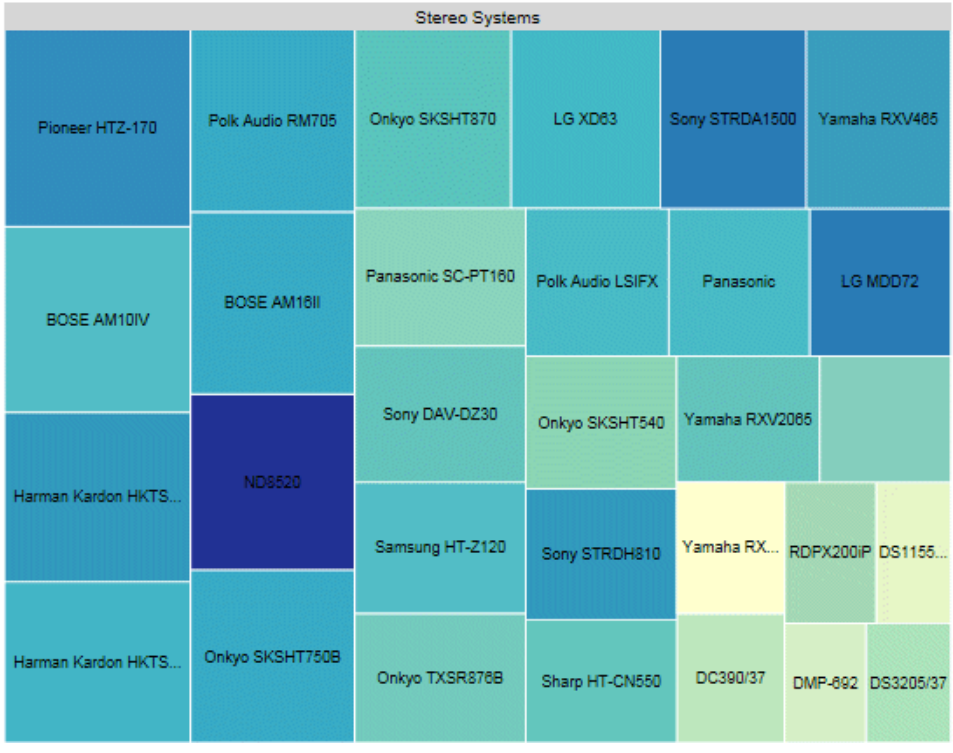


ツリーマップのプロパティ (treemapProperties)

ここでは、ツリーマップのヘッダセルおよびセル境界の外観を制御するプロパティについて説明します。

注意：ツリーマップで "legend": {"visible": true} が有効な場合、色スケールの凡例は、グラフに色データが含まれている場合にのみ表示されます。シリーズの凡例は、グラフにシリーズ ID データが含まれている場合にのみ表示されます。

下図のように、ツリーマップのデータラベルが長方形に収まらない場合は、末尾が切り捨てられ、省略記号 (...) が表示されます。



次のコードセグメントは、デフォルト値を示しています。

```
"treemapProperties": {
    "scaleCellFonts": false,
    "header": {
        "height": undefined,
        "fill": "lightgrey",
        "border": {
            "width": 0,
            "color": "lightgrey",
            "dash": ""},
        "label": {
            "visible": true,
            "font": "8pt Sans-Serif",
            "color": "black"}},
    "cellBorder": {
        "width": 1,
        "color": "white",
        "dash": "",
        "outerCellWidth": 3}}
```

ツリーマップセルのフォントの拡大縮小

scaleCellFonts プロパティは、ツリーマップのセル内でラベルを描画する方法を制御します。

構文　ツリーマップセルのフォントの拡大縮小

```
"treemapProperties": {
    "scaleCellFonts": boolean,
    }
```

説明

`"scaleCellFonts": boolean`

有効な値には、次のものがあります。

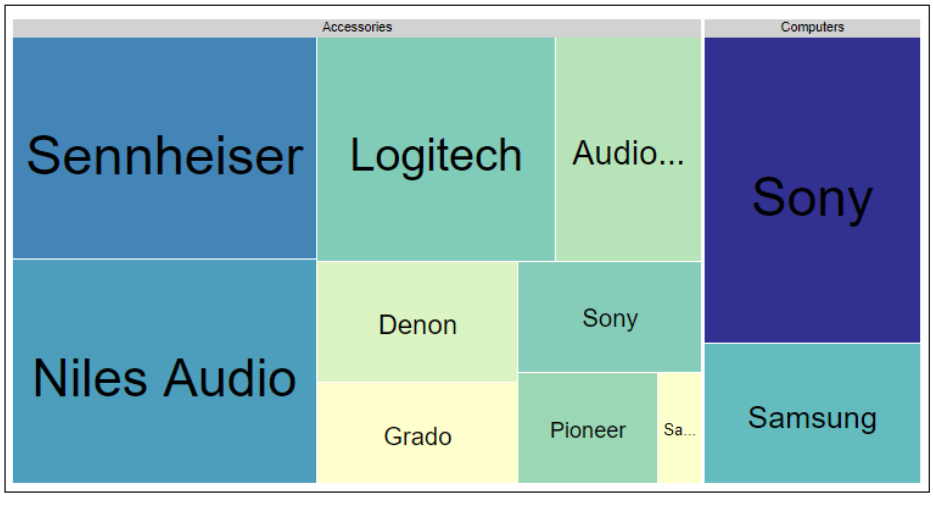
- true - 各セル内に描画するラベルのフォントサイズをセル領域に応じて拡大縮小します。
- false - すべてのセル内のラベルに同一のフォントサイズを使用します。ラベルサイズがセルサイズを超える場合はラベルを描画しません。デフォルト値は false です。

例　ツリーマップセルのフォントの拡大縮小

次のリクエストは、ツリーマップを生成します。scaleCellFonts プロパティは true に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM GROSS_PROFIT_US COGS_US
BY PRODUCT_CATEGORY
BY BRAND
WHERE PRODUCT_CATEGORY EQ 'Computers' OR 'Accessories'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
type=report, column=COGS_US, bucket=size, $
type=report, column=GROSS_PROFIT_US, bucket=color, $
type=report, column=PRODUCT_CATEGORY, bucket=detail, $
type=report, column=BRAND, bucket=detail, $
*GRAPH_JS
"legend": {"visible": false},
"dataLabels": {"visible": true},
"treemapProperties": {"scaleCellFonts": true}
*END
ENDSTYLE
END
```

出力結果では、セル領域が大きくなると、フォントサイズも拡大されています。



ツリーマップのヘッダのフォーマット設定

ここでは、ツリーマップのヘッダのフォーマットを設定するプロパティについて説明します。

構文　　ツリーマップのヘッダのフォーマット設定

```
"treemapProperties": {
      "header": {
      "height": height,
      "fill": "string",
      "border": {
         "width": width,
         "color": "string",
         "dash": "string"
      },
      "label": {
         "visible": boolean,
         "font": "string",
         "color": "string"
      }
```

`"height": height`

ヘッダの高さを定義します。有効な値には、次のものがあります。

- undefined - ヘッダ高さを自動的に計算し、ヘッダラベル高さより 33% 高い値に設定します。これがデフォルト値です。
- 高さを定義する数値をピクセル単位で指定します。
- 高さをパーセント文字列で定義し、パーセント記号 (%) を含めて二重引用符 (") で囲みます (例、"5%")。このプロパティをパーセント文字列に設定すると、ヘッダ高さは、ツリーマップ全体の高さに対する割合で計算されます。

`"fill": "string"`

undefined を指定するか、色またはグラデーションを定義する文字列を指定します。デフォルト値は "lightgrey" です。色およびグラデーションの定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

`"border":`

ヘッダ境界線のプロパティを定義します。

`"width": width`

ヘッダ境界線の幅をピクセル単位で指定します。デフォルト値は 0 (ゼロ) です。

`"color": "string"`

境界線の色を色名またはグラデーション定義で指定します。デフォルト値は "lightgrey" です。

"dash": "string"

ヘッダ境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"label":

ヘッダラベルのプロパティを定義します。

"visible": boolean

ヘッダラベルの表示を制御します。有効な値には、次のものがあります。

- true - ヘッダラベルを表示します。これがデフォルト値です。
- false - ヘッダラベルを表示しません。

"font": "string"

ヘッダラベルのフォントを定義する文字列です。デフォルト値は "8pt Sans-Serif" です。

"color": "string"

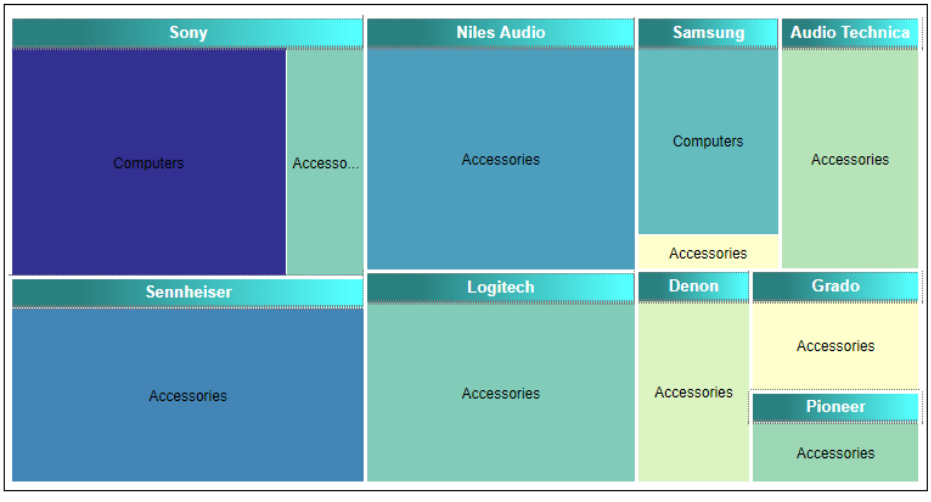
ヘッダラベルの色を定義する文字列です。デフォルト値は "black" です。

例　ツリーマップのヘッダのフォーマット設定

次のリクエストは、ツリーマップを生成します。ヘッダの高さは 25 ピクセルです。ヘッダの境界は灰色の 4 ピクセル幅の破線で、ヘッダの背景は青緑色からシアンに変化する線状グラデーションで塗りつぶします。ヘッダ内のラベルは、白色、太字、10pt Sans-Serif に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM GROSS_PROFIT_US COGS_US
BY BRAND
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers' OR 'Accessories'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
type=report, column=COGS_US, bucket=size, $
type=report, column=GROSS_PROFIT_US, bucket=color, $
type=report, column=PRODUCT_CATEGORY, bucket=detail, $
type=report, column=BRAND, bucket=detail, $
*GRAPH_JS
"legend": {"visible": false},
"treemapProperties": {
   "header": {
      "height": 25,
      "fill": "linear-gradient(0%,0%,100%,0%, 20% teal, 95% cyan)",
      "border": {"width": 4, "color": "grey", "dash": "1 1"},
      "label": {"visible": true, "font": "bold 10pt Sans-Serif", "color": "white"}}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ツリーマップのセル境界線のフォーマット設定

これらのプロパティは、ツリーマップのセル境界線のフォーマットを設定します。

構文　ツリーマップのセル境界線のフォーマット設定

```
"treemapProperties": {
    "cellBorder": {
      "width": number,
      "color": "string",
      "dash": "string",
      "outerCellWidth": number   }
},
```

説明

`"width": number`

セル境界線の幅をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

セル境界線の色を色名またはグラデーション定義の文字列で指定します。デフォルト値は "white" です。

"dash": "string"

セル境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

"outerCellWidth": number

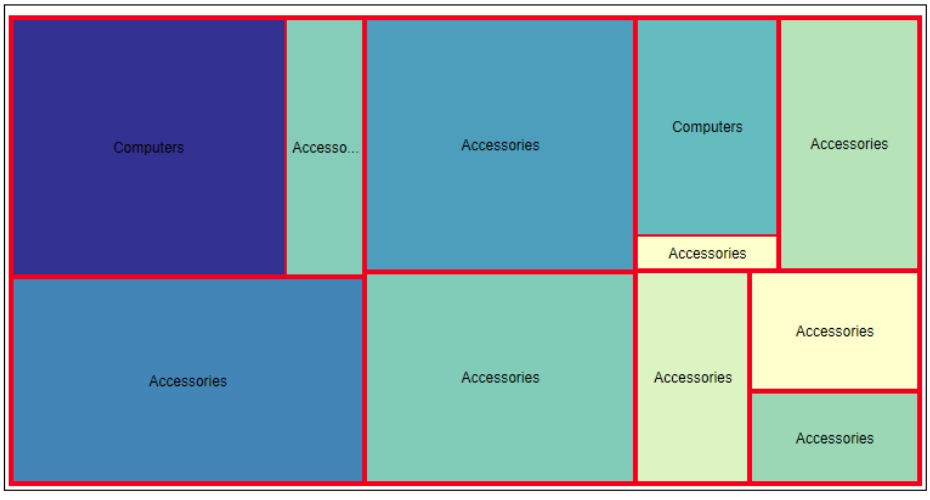
最上位セルの周囲に描画する境界線の幅をピクセル単位で定義します。このプロパティで制御される境界幅は、最上位 (ルート) ノード周囲の境界のみです。デフォルト値は 3 です。

例　ツリーマップのセル境界線のフォーマット設定

次のリクエストは、ツリーマップを生成します。ヘッダの高さを 0 (ゼロ) に設定して、ヘッダを非表示にします。セルの境界線は赤色の 2 ピクセル幅で、外側の境界線は 4 ピクセル幅です。

```
GRAPH FILE WF_RETAIL_LITE
SUM GROSS_PROFIT_US COGS_US
BY BRAND
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Computers' OR 'Accessories'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
type=report, column=COGS_US, bucket=size, $
type=report, column=GROSS_PROFIT_US, bucket=color, $
type=report, column=PRODUCT_CATEGORY, bucket=detail, $
type=report, column=BRAND, bucket=detail, $
*GRAPH_JS
"legend": {"visible": false},
"treemapProperties": {
   "header": {"height": 0},
   "cellBorder": {"width": 2, "color": "red", "outerCellWidth": 4}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



滝型グラフのプロパティ (waterfallProperties)

滝型グラフは、初期値に正の値または負の値を連続的に加算することで、累積結果を視覚的に表現するグラフです。初期値および最終値 (合計値) は、X 軸のベースラインから伸びたライザの長さで表されます。初期値から最終値までの正の値および負の値は、浮動ライザとして表されます。

ここでは、滝型グラフの全体的なレイアウトを制御するプロパティについて説明します。

次のコードセグメントは、デフォルト設定を示しています。

```
"waterfallProperties":
{
   "appendTotalRiser": true,
   "totalLabel: "Total",
   "positiveRiserColor": "#77b39a",
   "negativeRiserColor": "#e2675b",
   "zeroRiserColor": "#7593bd",
   "totalRiserColor: "#7593bd",
   "otherRiserColor": "#aaaaaa",
   "subtotalRisers": [],
   "otherRisers": [],
   "subtotalFormat":
   {
      "bold": undefined,
      "color": undefined,
      "numberFormat": undefined,
      "dataLabelPosition": undefined,
      "dividerLine":
        {
         "width": 0,
         "color": "black",
         "dash": ""
         }
    },
  "firstLastFormat":
  {
    "color": undefined,
    "border":
     {
      "width": 0,
      "color": undefined,
      "dash": ""
     }
  },
   "connectorLine":
     {
      "width": 1,
      "color": "black",
      "dash": ""
      }
},
```

滝型グラフでの合計ライザの追加

ここでは、滝型グラフで最後のライザとして合計ライザを生成するプロパティについて説明します。

構文　滝型グラフでの合計ライザの追加

```
"waterfallProperties": {
   "appendTotalRiser": boolean   },
```

説明

`"appendTotalRiser": boolean`

有効な値には、次のものがあります。

- true - 合計ライザを描画します。これがデフォルト値です。
- false - 合計ライザを描画しません。

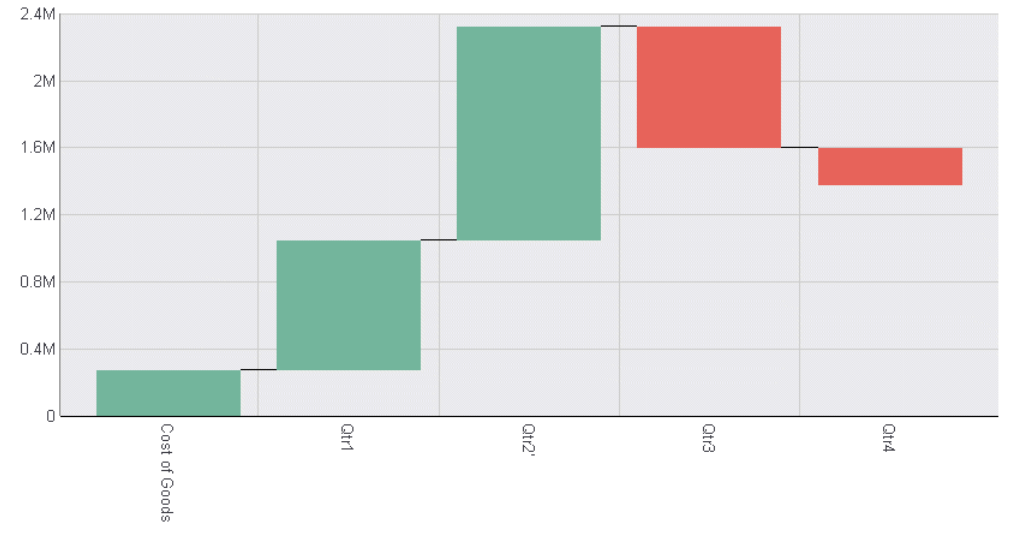
例　滝型グラフでの合計ライザの追加

次のリクエストは、滝型グラフを生成し、合計ライザを表示します (デフォルト)。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
DECR1 = (COGS_US - 1000);
DECR2 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 AS 'Qtr1'
INCR2 AS 'Qtr2'
DECR1  AS 'Qtr3'
DECR2 AS 'Qtr4'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"waterfallProperties": {"appendTotalRiser": true}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

appendTotalRiser を false に変更すると、下図のグラフが生成されます。



合計ライザへのカスタムラベルの追加

ここでは、滝型グラフで合計ライザを生成する場合に、合計ライザのラベルをカスタマイズするプロパティについて説明します。

構文　合計ライザラベルのカスタマイズ

```
"waterfallProperties": {
   "appendTotalRiser": true,
   "totalLabel": "string"   },
```

説明

`"appendTotalRiser": true`
　合計ラベルを生成するには、true に設定します。これがデフォルト値のため、このプロパティは省略することができます。

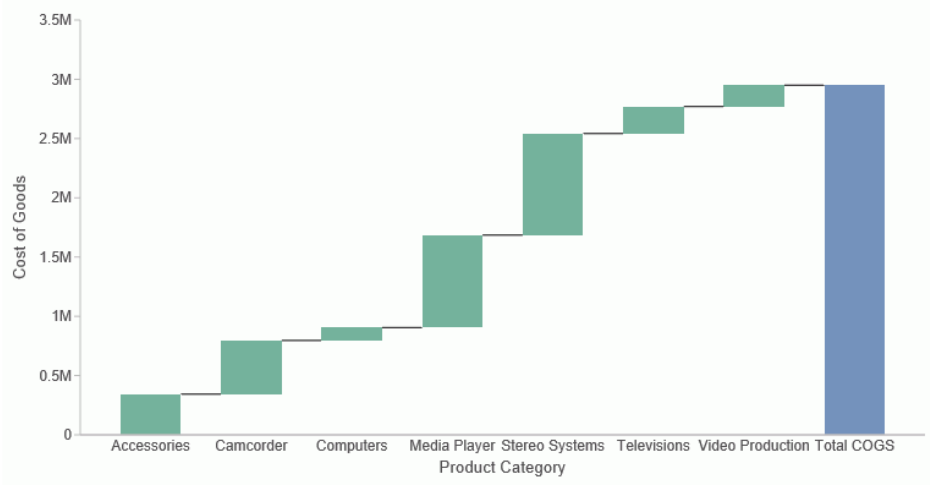
`"totalLabel": "string"`
　合計ライザのラベルとして使用する文字列です。デフォルトラベルは「Total」です。

例　合計ライザラベルのカスタマイズ

次のリクエストは合計ライザを生成し、X 軸のラベルを「Total COGS」にカスタマイズします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH WATERFALL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS
"waterfallProperties":
{
"appendTotalRiser": true,
"totalLabel": "Total COGS",
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



滝型グラフの接続線のフォーマット設定

ここでは、滝型グラフで隣接するライザ間の接続線の外観を制御するプロパティについて説明します。

構文　滝型グラフの接続線のフォーマット設定

```
"waterfallProperties": {
   "connectorLine": {
      "width": number,
      "color": "string",
      "dash": "string"
   }
   },
```

説明

`"width": number`

接続線の幅をピクセル単位で指定します。デフォルト値は 1 です。

`"color": "string"`

色名または数値指定の文字列で定義する色です。デフォルト値は "black" です。

"dash": "string"

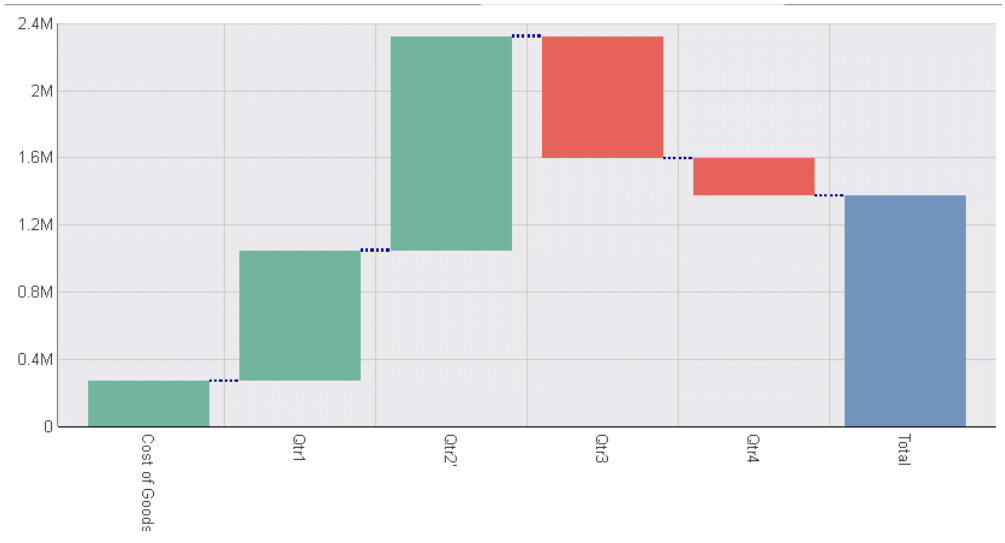
接続線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　滝型グラフの接続線のフォーマット設定

次のリクエストは、滝型グラフを生成し、接続線を 2 ピクセル幅、濃い青色、破線に設定します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
DECR1 = (COGS_US - 1000);
DECR2 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 AS 'Qtr1'
INCR2 AS 'Qtr2'
DECR1  AS 'Qtr3'
DECR2 AS 'Qtr4'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"waterfallProperties": {
    "connectorLine": {"width": 2, "color": "darkblue", "dash": "2 2"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



滝型グラフでの負の値のライザ色の制御

ここでは、滝型グラフで負の値を表すライザの色を制御するプロパティについて説明します。

注意：シリーズ固有のプロパティを使用して、各ライザのフォーマットを個別に設定することができます。その場合は、シリーズ 0 の各グループに対して色を指定します。

構文　滝型グラフでの負の値のライザ色

```
"waterfallProperties": {
   "negativeRiserColor": "color"   },
```

説明

`"negativeRiserColor": "color"`

負の値のライザ色を定義します。有効な値には、次のものがあります。

- 色名または色を定義する数値指定の文字列、グラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "#e2675b" です。
- グラデーションを定義する JSON オブジェクト。

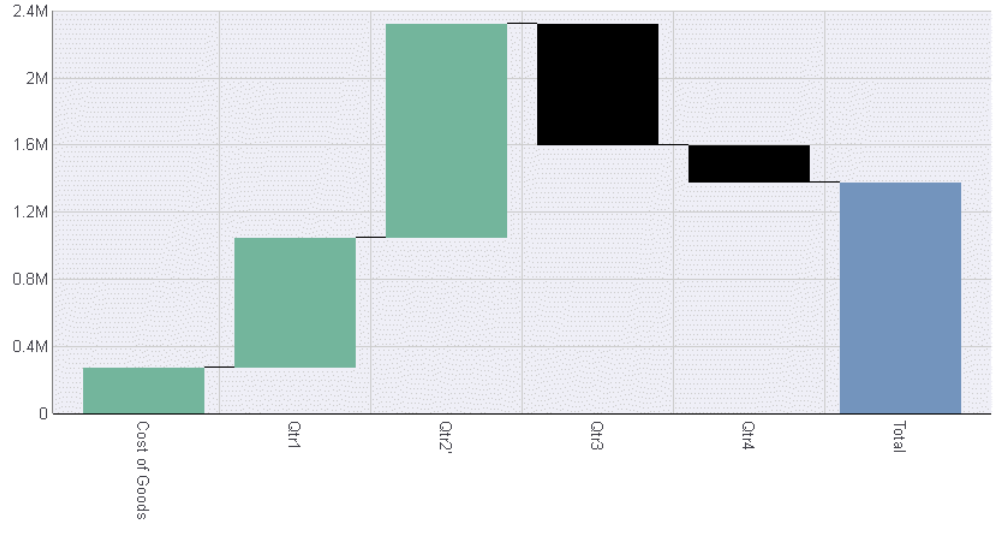
色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　　滝型グラフでの負の値のライザ色の設定

次のリクエストは、滝型グラフを生成し、負の値のライザを黒色に設定します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
DECR1 = (COGS_US - 1000);
DECR2 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 AS 'Qtr1'
INCR2 AS 'Qtr2'
DECR1  AS 'Qtr3'
DECR2 AS 'Qtr4'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"waterfallProperties": {"negativeRiserColor": "black"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



注意：シリーズ固有のプロパティを使用して、各ライザのフォーマットを個別に設定することができます。その場合は、シリーズ 0 の各グループに対して色を指定します。以下はその例です。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
DECR1 = (COGS_US - 1000);
DECR2 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 AS 'Qtr1'
INCR2 AS 'Qtr2'
DECR1  AS 'Qtr3'
DECR2 AS 'Qtr4'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"series": [
    {"series": 0, "group": 0, "color": "#7593bd"},
    {"series": 0, "group": 1, "color": "#77b39a"},
    {"series": 0, "group": 2, "color": "#77b39a"},
    {"series": 0, "group": 3, "color": "#7593bd"},
    {"series": 0, "group": 4, "color": "coral"},
    {"series": 0, "group": 5, "color": "pink"}]
*END
ENDSTYLE
END
```

出力結果では、各グループで指定された色がそれぞれのライザに反映されています。

滝型グラフの「その他」ライザ色の指定

otherRiserColor プロパティは、滝型グラフで [その他] の値 (存在する場合) を表すライザの色を制御します。[その他] の値は、otherRisers 配列で定義します。

構文　滝型グラフの「その他」ライザ色の指定

```
"waterfallProperties": {
   "otherRiserColor": "color"   },
```

説明

`"otherRiserColor": "color"`

[その他] ライザの色を指定します。有効な値には、次のものがあります。

- 色名または色を定義する数値指定の文字列、グラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "#aaaaaa" です。
- グラデーションを定義する JSON オブジェクト。

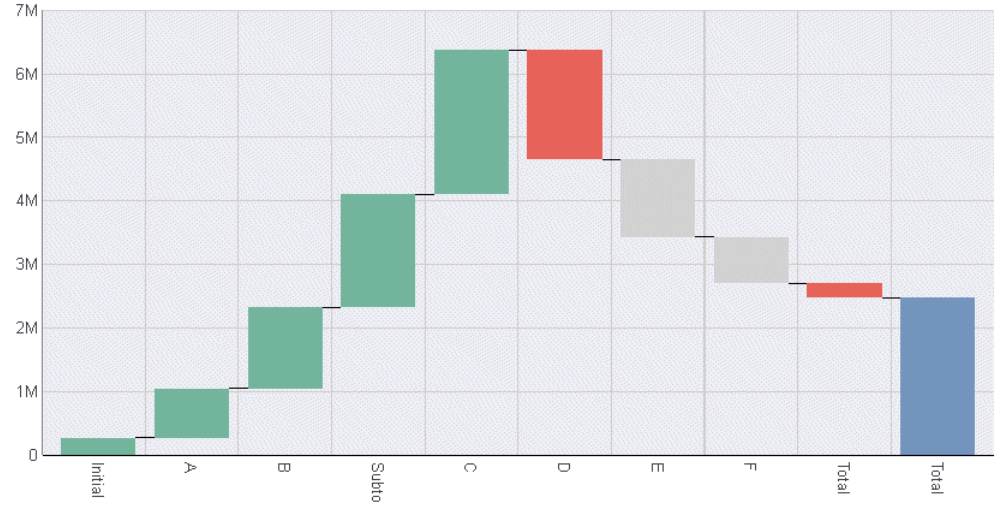
色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　　滝型グラフの「その他」ライザ色の指定

次のリクエストは、滝型グラフを生成し、[その他] ライザを薄い灰色に設定します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
INCR3 = COGS_US +1500;
INCR4 = COGS_US +2000;
DECR1 = (COGS_US - 2000);
DECR2 = (COGS_US -1500);
DECR3 = (COGS_US -1000);
DECR4 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 INCR2 INCR3 INCR4
DECR1 DECR2 DECR3 DECR4
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"groupLabels": ["Initial", "A", "B", "Subtotal", "C", "D", "E", "F", "Total"],
"waterfallProperties": {
    "otherRiserColor": "lightgrey", "otherRisers": ["E", "F"]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



注意：シリーズ固有のプロパティを使用して、特定のライザのフォーマットを個別に設定することもできます。 例については、741 ページの 「 滝型グラフでの負の値のライザ色の制御 」を参照してください。

「その他」ライザの指定

otherRisers プロパティは、[その他] の値として解釈される 1 つまたは複数のグループまたは値を定義します。これらの値を表すライザには、otherRiserColor プロパティで定義された色が割り当てられます。

構文　「その他」ライザの指定

```
"waterfallProperties": {
   "otherRisers": [array]
   },
```

説明

`"otherRisers": [array]`

次のいずれかの配列を指定することができます。

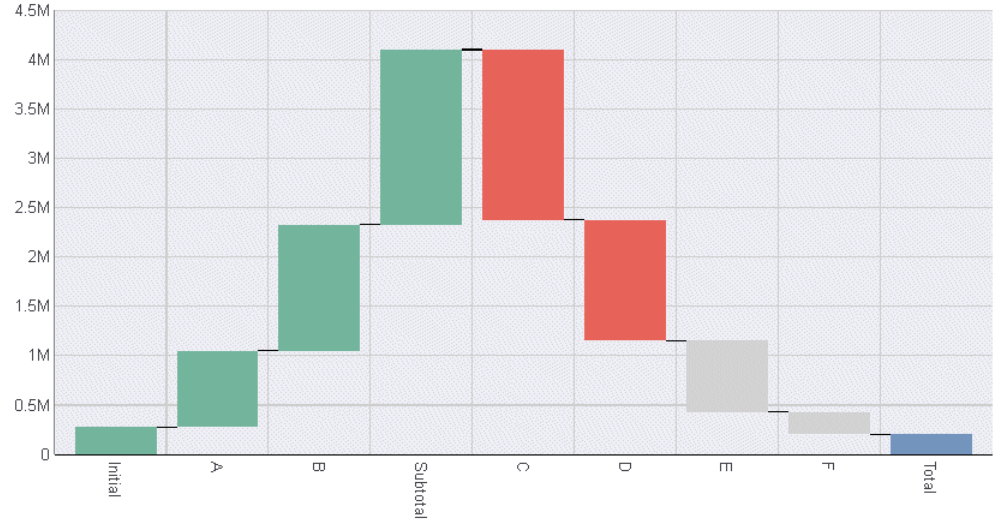
- グループインデックスを指定する番号。
- グループ名を指定する文字列。文字列は、二重引用符 (") で囲みます。

例　「その他」ライザの指定

次のリクエストは、滝型グラフを生成します。groupLabels オブジェクトで、各グループにグループラベル名を割り当てます。"waterfallProperties": "otherRisers" プロパティで、グループ E および F を [その他] ライザカテゴリに割り当てます。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
INCR3 = COGS_US +1500;
DECR1 = (COGS_US - 2000);
DECR2 = (COGS_US -1500);
DECR3 = (COGS_US -1000);
DECR4 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 INCR2 INCR3
DECR1  DECR2 DECR3 DECR4
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"groupLabels": ["Initial", "A", "B", "Subtotal", "C", "D", "E", "F", "Total"],
"waterfallProperties": {
    "otherRiserColor": "lightgrey", "otherRisers": ["E", "F"]}
*END
ENDSTYLE
END
```

出力結果では、[その他] ライザが、otherRiserColor プロパティで割り当てられた薄い灰色で描画されています。



滝型グラフでの正の値のライザ色の指定

positiveRiserColor プロパティは、滝型グラフの正の値を表すライザの色を制御します。

構文　滝型グラフでの正の値のライザ色の指定

```
"waterfallProperties": {
   "positiveRiserColor": "color"   },
```

説明

`"positiveRiserColor": "color"`

正の値のライザ色を定義します。有効な値には、次のものがあります。

- 色名または色を定義する数値指定の文字列、グラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "#77b39a" です。
- グラデーションを定義する JSON オブジェクト。

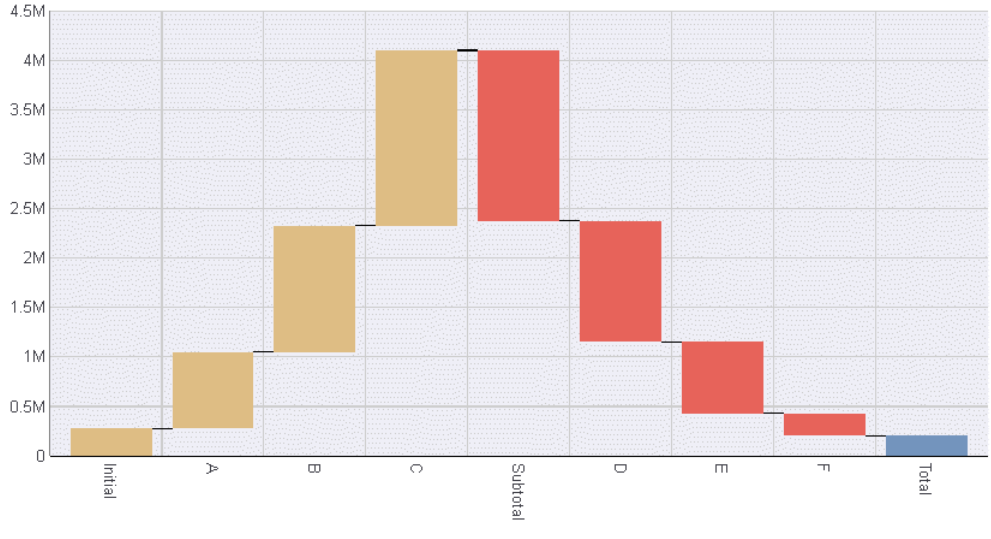
色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　　滝型グラフでの正の値のライザ色の指定

次のリクエストは、滝型グラフを生成し、正の値のライザ色をバーリーウッドに設定します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
INCR3 = COGS_US +1500;
DECR1 = (COGS_US - 2000);
DECR2 = (COGS_US -1500);
DECR3 = (COGS_US -1000);
DECR4 = (COGS_US -500);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 INCR2 INCR3
DECR1  DECR2 DECR3 DECR4
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"groupLabels": ["Initial", "A", "B", "C", "Subtotal", "D", "E", "F", "Total"],
"waterfallProperties": {"positiveRiserColor": "burlywood"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



注意：シリーズ固有のプロパティを使用して、特定のライザのフォーマットを個別に設定することもできます。 例については、741 ページの 「 滝型グラフでの負の値のライザ色の制御 」を参照してください。

中間合計値の指定

ここでは、「中間合計」の値として解釈される 1 つ以上のグループまたは値を定義するプロパティについて説明します。

構文　　中間合計値の割り当て

```
"waterfallProperties": {
   "subtotalRisers": [array]
   },
```

説明

`"subtotalRisers": [array]`

次のいずれかの配列を指定することができます。

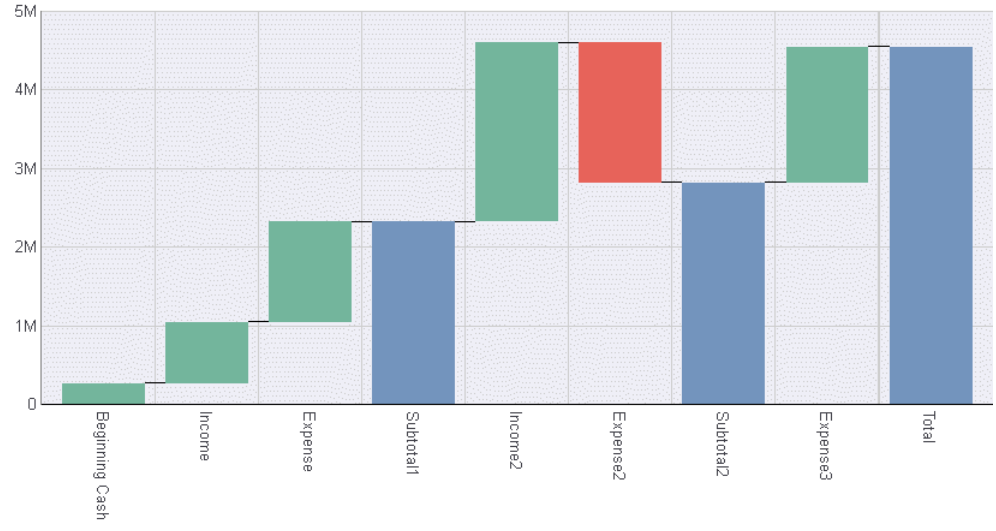
- グループインデックスを指定する番号。
- グループ名を指定する文字列。文字列は、二重引用符 (") で囲みます。

例　　中間合計ライザの指定

次のリクエストは、滝型グラフを生成し、2 つの中間合計ライザを表示します。groupLabels オブジェクトで、各ライザに名前を割り当てます。subtotalRisers オブジェクトの配列で、これらの名前を指定します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
INCR3 = COGS_US +INCR1 +INCR2;
INCR4 = COGS_US +2000;
DECR1 = -(INCR4- 500);
DECR2 = INCR3 +500;
DECR3 = -(COGS_US-2000);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 INCR2 INCR3 INCR4
DECR1 DECR2  DECR3
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"groupLabels": ["Beginning Cash", "Income", "Expense", "Subtotal1",
    "Income2", "Expense2", "Subtotal2", "Expense3", "TOTAL"],
"waterfallProperties": {"subtotalRisers": ["Subtotal1", "Subtotal2"]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



滝型グラフの中間合計ライザ色および合計ライザ色の指定

zeroRiserColor プロパティは、滝型グラフで subtotal 配列に配置されたライザの色、および合計値を表すライザの色を制御します。

構文　滝型グラフの初期値ライザ、中間合計ライザ、合計ライザの色指定

```
"waterfallProperties": {
   "zeroRiserColor": "color"   },
```

説明

`"zeroRiserColor": "color"`

subtotal 配列に配置されたライザの色、および合計値を表すライザの色を指定します。有効な値には、次のものがあります。

- 色名または色を定義する数値指定の文字列、グラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。デフォルト値は "#7593bd" です。
- グラデーションを定義する JSON オブジェクト。

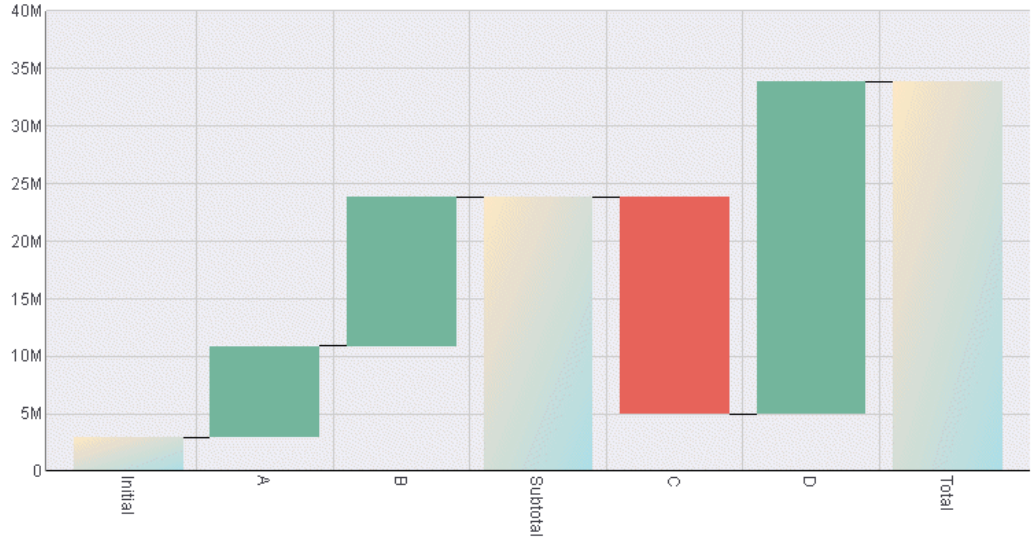
色およびグラデーションの指定についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

例　滝型グラフの zeroRiserColor プロパティの設定

次のリクエストは、滝型グラフを生成し、zeroRiserColor プロパティをビスクから水色に変化する線状グラデーションに設定します。また、初期値ライザを subtotal 配列に追加します。これにより、初期値ライザが、指定されたグラデーションで表示されます。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
INCR3 = COGS_US +INCR1 +INCR2;
DECR1 = -(INCR3- 500);
DECR2 = INCR3 +500;
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 INCR2 INCR3
DECR1 DECR2
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
*GRAPH_JS
"legend": {"visible": false},
"groupLabels": ["Initial", "A", "B", "Subtotal", "C", "D", "Total"],
"waterfallProperties": {
    "subtotalRisers": ["Initial", "Subtotal"],
    "zeroRiserColor": {
        "type": "linear",
        "start": {"x": "0%", "y": "0%"},
        "end": {"x": "100%", "y": "100%"},
        "stops": [[0, "bisque"], [1, "lightblue"]]}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



注意：シリーズ固有のプロパティを使用して、特定のライザのフォーマットを個別に設定することもできます。 例については、741 ページの 「 滝型グラフでの負の値のライザ色の制御 」を参照してください。

中間合計ライザの軸ラベルおよびデータラベルプロパティのフォーマット設定

subtotalFormat プロパティを使用して、中間合計ライザの軸ラベル、データラベル、分割線をカスタマイズすることができます。

構文　中間合計ライザの軸ラベルおよびデータラベルプロパティのフォーマット設定

```
"waterfallProperties":
{
  "subtotalFormat":
     {
        "bold": boolean,
        "color": "string",
        "numberFormat": format,
        "dataLabelPosition": "string",
        "dividerLine":
          {
           "width": number,
           "color": "string",
           "dash": "string"
           }
     }
}
```

説明

`"bold": boolean`

次のいずれかの値です。

- true - 中間合計ライザの軸ラベルを太字にします。
- false - 中間合計ライザの軸ラベルを太字にしません。

デフォルト値は undefined です。

`"color": "string"`

中間合計ライザのデータラベルの色を指定する色指定文字列です。デフォルト値は undefined です。

`"numberFormat": format`

中間合計ライザのデータラベルの数値フォーマットを定義する数値フォーマット文字列または JSON オブジェクトです。詳細は、61 ページの「 HTML5 グラフの JSON プロパティの概要 」 を参照してください。

デフォルト値は undefined です。

`"dataLabelPosition": "string"`

中間合計ライザのデータラベルの配置を定義する文字列です。詳細は、445 ページの「 データテキストラベルの表示およびフォーマット設定 」 を参照してください。

デフォルト値は undefined です。

"dividerLine"
各中間合計ライザの前に描画するオプションの分割線のプロパティを定義します。

"width": number
分割線の幅をピクセル単位で指定します。デフォルト値は 0 (ゼロ) です。

"color": "string"
分割線に適用する色指定の文字列です。デフォルト値は "black" です。

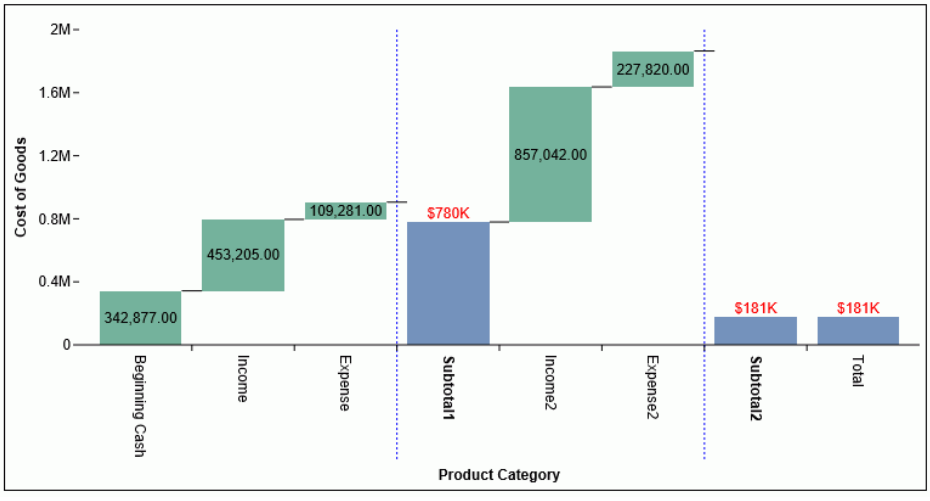
"dash": "string"
分割線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔をピクセル単位で定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　中間合計ライザの軸ラベルおよびデータラベルプロパティのフォーマット設定

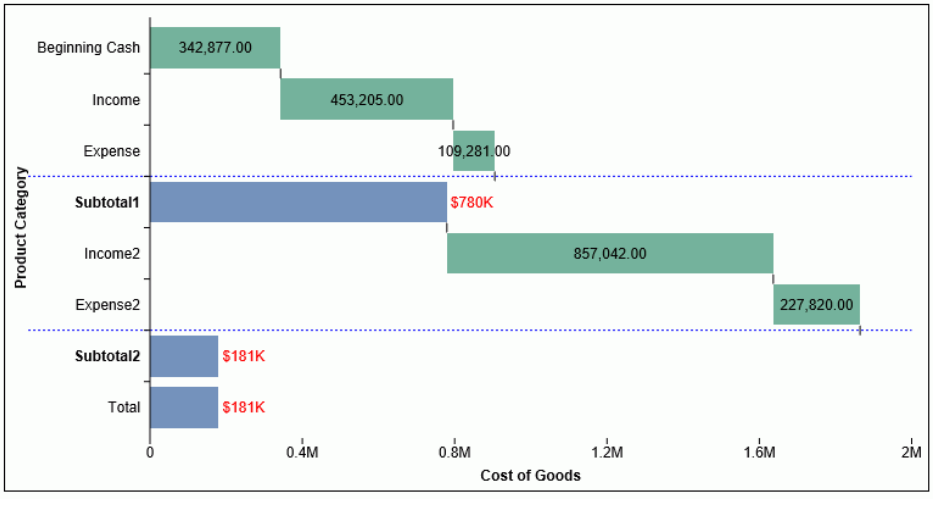
次のリクエストは、中間合計ライザの軸ラベルを太字にし、青色の破線で分割線を生成します。また、データラベルは赤色でライザの上側に配置し、JSON オブジェクトで定義された数値フォーマットを使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US

BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH WATERFALL
ON GRAPH SET STYLE *
*GRAPH_JS
"series": [{"series":0, "dataLabels": {"visible":true}}],
"groupLabels": ["Beginning Cash", "Income", "Expense", "Subtotal1",
    "Income2", "Expense2", "Subtotal2", "Expense3", "TOTAL"],
"waterfallProperties":
{
"subtotalRisers": ["Subtotal1", "Subtotal2"],
"subtotalFormat":
   {
    "bold": true,
    "color": "red",
    "numberFormat": {"mode": "currency", "decimalPlaces": 0,
                     "grouping": "K"},
    "dataLabelPosition": "top",
    "dividerLine":
      {
       "width": 1,
       "color": "blue",
       "dash": "2 2"
       }
   }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



LOOKGRAPH 値を HWATERFL に変更すると、次の横向きの滝型グラフが出力されます。



滝型グラフでの最初および最後のライザのフォーマット設定

firstLastFormat プロパティを使用して、滝型グラフの最初および最後のライザのフォーマットをカスタマイズすることができます。

構文　　滝型グラフでの最初および最後のライザのフォーマット設定

```
"firstLastFormat":
 {
   "color": color,
   "border":
     {
      "width": number,
      "color": "string",
      "dash": "string"
     }
 }
```

説明

`"color": color`

最初および最後のライザの色を指定します。デフォルト値は undefined です。この場合、最初および最後のライザと他のライザを区別しません。有効な値には、次のものがあります。

- 色名または色を定義する数値指定の文字列、グラデーションを定義する文字列。文字列は二重引用符 (") で囲みます。
- グラデーションを定義する JSON オブジェクト。

色およびグラデーションの指定についての詳細は、63 ページの「 色およびグラデーション 」 を参照してください。

`"border"`

最初および最後のライザの境界プロパティを定義します。

`"width": number`

境界線の幅をピクセル単位で指定します。デフォルト値は 0 (ゼロ) です。

`"color": "string"`

色名または数値指定の文字列で定義する色です。デフォルト値は undefined です。

`"dash": "string"`

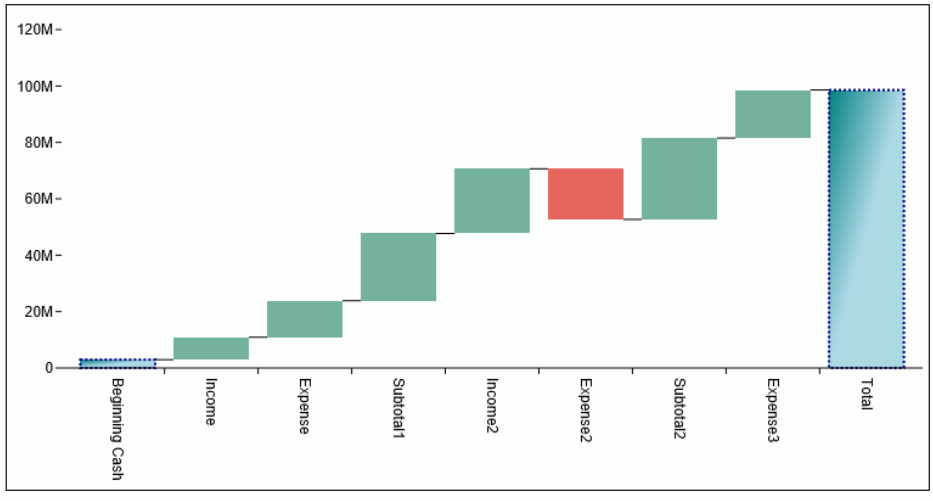
境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。

例　最初および最後のライザのフォーマット設定

次のリクエストは、最初および最後のライザのフォーマットを、青緑色から水色に変化するグラデーションに設定し、紺色の破線で境界線を描画します。

```
DEFINE FILE WF_RETAIL_LITE
INCR1 = COGS_US + 500;
INCR2 = COGS_US +1000;
INCR3 = COGS_US +INCR1 +INCR2;
INCR4 = COGS_US +2000;
DECR1 = -(INCR4- 500);
DECR2 = INCR3 +500;
DECR3 = -(COGS_US-2000);
END
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
INCR1 INCR2 INCR3 INCR4
DECR1 DECR2  DECR3
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VWATERFL
ON GRAPH SET STYLE *
*GRAPH_JS
"legend": {"visible": false},
"groupLabels": ["Beginning Cash", "Income", "Expense", "Subtotal1",
    "Income2", "Expense2", "Subtotal2", "Expense3", "TOTAL"],
"waterfallProperties":
{
  "firstLastFormat":
  {
   "color":
      {"type": "linear",
             "start": {"x": "0%", "y": "0%"},
             "end": {"x": "100%", "y": "100%"},
             "stops": [[0, "teal"], [0.5, "lightblue"]]},
    "border":
      {
       "width": 2,
       "color": "navy",
       "dash": "2 2"
      }
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



11

グラフアクセシビリティのプロパティ

アクセシビリティのプロパティを使用すると、スクリーンリーダーが読み上げを行うグラフコンポーネントを作成することができます。また、グラフのライザおよびマーカーで、Tab キーナビゲーションを有効にすることもできます。

トピックス

- アクセシビリティプロパティの概要
- アクセシビリティのタイトルおよび説明の定義
- グラフのライザおよびマーカーでの Tab キーナビゲーションの有効化

アクセシビリティプロパティの概要

次の構文は、アクセシビリティのプロパティおよびデフォルト値を示しています。

```
"accessibility": {
   "title": undefined,
   "description": undefined,
   "navigation": {
                  "enabled": false,
                  "mode": "cycleRisers",
                  "tabLimit": 100
                }
     }
```

アクセシビリティのタイトルおよび説明の定義

"accessibility": "title" プロパティは、グラフにタイトルを追加します。このタイトルは、グラフ上にマウスポインタを置いた際に、ツールヒントとして表示されます。このタイトルは、使用中のスクリーンリーダーによって読み上げられます。"accessibility": "description" は説明を追加しますが、この説明はグラフ上には表示されません。ただし、使用中のスクリーンリーダによって読み上げられます。

これらのプロパティには、グラフマクロを参照して動的コンテンツを追加することができます。詳細は、109 ページの 「 グラフテンプレートエンジンを使用したツールヒント、タイトル、データラベルのカスタマイズ 」 を参照してください。

タイトルおよび説明で最も役立つマクロは、件数 (例、{{group_count}}) や軸タイトル (例、yaxis_title) など、グラフ全体に適用される値を返すマクロです。

構文　アクセシビリティのタイトルおよび説明の定義

```
"accessibility": {
   "title": "string",
   "description": "string"
      }
```

説明

`"title": "string"`

グラフ上にマウスポインタを置いた際に、ツールヒントとして表示するタイトル文字列を定義します。このタイトル文字列には、グラフマクロを含めることができ、この文字列は、使用中のスクリーンリーダーで読み上げられます。

`"description": "string"`

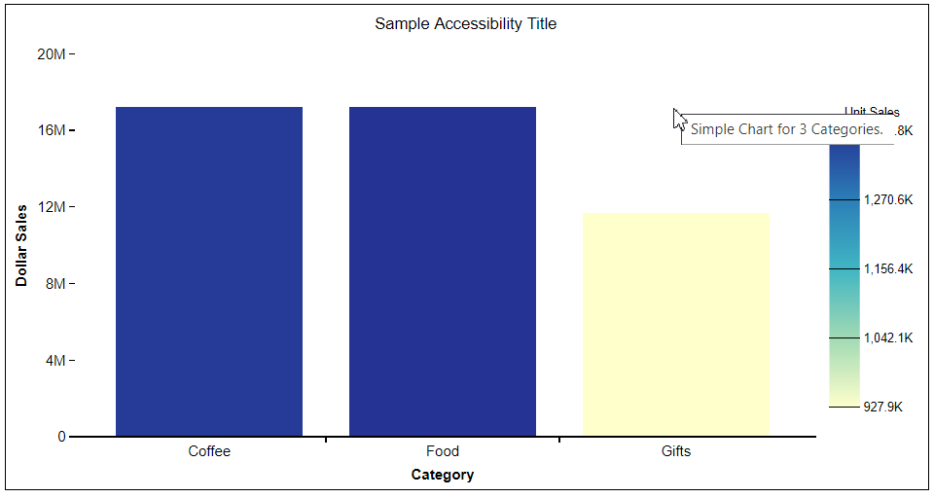
グラフ上に表示しない説明文字列を定義します。この文字列には、グラフマクロを含めることができ、この文字列は、使用中のスクリーンリーダーで読み上げられます。

例　アクセシビリティのタイトルおよび説明の定義

次のリクエストは、アクセシビリティのタイトルおよび説明を定義します。タイトルには {{group_count}} マクロが使用され、グラフ上にグループ数が返されます。説明には {{yaxis_title}} マクロ、{{xaxis_title}} マクロ、{{coloraxis_title}} マクロが使用され、グラフの表示項目が記述されます。

```
GRAPH FILE ibisamp/ggsales
HEADING CENTER
"Sample Accessibility Title"
SUM DOLLARS
UNITS
BY CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N3, BUCKET=color, $
*GRAPH_JS
"axisList":{
"y1": {"majorGrid": {"visible": false}},
"x1": {"majorGrid": {"visible": false}}
            },
"accessibility": {
            "title": "Simple Chart for {{group_count}} Categories.",
            "description": "{{yaxis_title}} by {{xaxis_title}} Colored by
{{coloraxis_title}}."
       }
*END
ENDSTYLE
END
```

下図では、アクセシビリティタイトルがツールヒントとして表示されています (Simple Chart for 3 Categories)。このテキストは、スクリーンリーダーによって読み上げられます。このグラフマクロ参照は、グラフ上のグループ数で置き換えられています。



アクセシビリティの説明の結果として、スクリーンリーダーは、次の読み上げも行います。

```
Dollar Sales by Category Colored by Unit Sales
```

グラフのライザおよびマーカーでの Tab キーナビゲーションの有効化

グラフでアクセシビリティプロパティを設定して、ライザとマーカーでの Tab キーナビゲーションを有効にすることができます。ライザとマーカーで Tab キーナビゲーションを実行する際、スクリーンリーダーが有効化されている場合は、フォーカス中のライザまたはマーカーのツールヒントテキストが読み上げられます。

ナビゲーションは、グラフにフォーカスを移動することから開始され、次に、最初のシリーズの最初のマーカーまたはライザに移動します。Tab キーを押すと、同一シリーズの次のマーカーまたはライザに移動します。最初のシリーズの最後のライザまたはマーカーに到達した時点で Tab キーを押すと、次のシリーズの最初のライザまたはマーカーに移動します。Shift +Tab を押すと、ライザまたはマーカーを逆方向に移動します。 複数のグラフが表示されている場合は、1 つのグラフですべてのライザおよびマーカーへの移動が一巡した時点で、次のグラフでの Tab ナビゲーションが開始されます。

構文　　グラフのライザおよびマーカーでの Tab キーナビゲーションの有効化

```
"accessibility": {
        "navigation": {
                        "enabled": boolean,
                        "mode": "cycleRisers",
                        "tabLimit": number
                    },
       },
```

説明

`"enabled": boolean`

ライザおよびマーカーでの Tab キーナビゲーションを、有効または無効にします。有効な値には、次のものがあります。

- true - ライザおよびマーカーでの Tab キーナビゲーションを有効にします。
- **false** - ライザおよびマーカーでの Tab キーナビゲーションを無効にします。これがデフォルト値です。

`"mode": "cycleRisers"`

Tab キーを押すことで、ライザとマーカーを巡回します。スクリーンリーダーが有効化されている場合は、ツールヒントのコンテンツとして提供されているすべての情報が読み上げられます。ナビゲーションが有効化されている場合、この設定がデフォルト値です。

`"tabLimit": number`

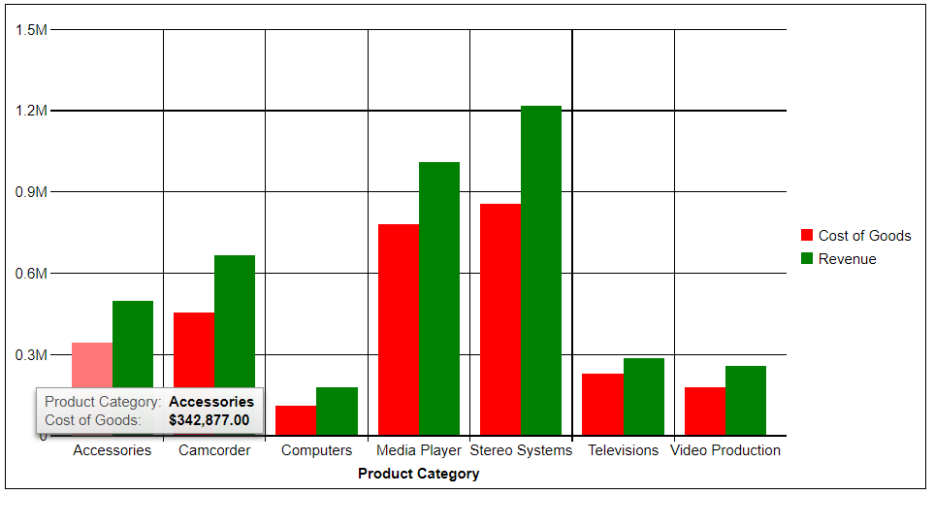
ナビゲーション対象のライザおよびマーカーの最大数です。この数値を超えるライザおよびマーカーは、ナビゲーションから除外されます。値 undefined は、無制限を指定します。デフォルト値は 100 です。

例　棒グラフでの Tab キーナビゲーションの有効化

次のリクエストは、棒グラフで Tab ナビゲーションを有効化します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON TABLE SET LOOKGRAPH BAR
ON TABLE PCHOLD FORMAT JSCHART
ON TABLE SET STYLE *
*GRAPH_JS
"accessibility": {
    "navigation": {
                   "enabled": true
                   }
                 },
*END
  TYPE=HEADING, STYLE=BOLD, $
  TYPE=REPORT, ORIENTATION=LANDSCAPE, $
  TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
  TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
  TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
ENDSTYLE
END
```

下図は、Tab キーを 2 回押した後の出力結果を示しています。



フォーカスは、最初のグループの最初のシリーズに置かれ、ツールヒントが表示されています。スクリーンリーダーが有効化されている場合は、ツールヒントのコンテンツが読み上げられます。

12

その他の機能

ここでは、HTML5 グラフの特別な機能について説明します。

トピックス

- アニメーション (introAnimation)
- モーフアニメーションの継続時間 (morphAnimation)
- データページスライダの速度制御
- JSON を使用したグラフイベント登録
- グラフ上の注釈
- PDF、Excel、PowerPoint、PNG フォーマットで使用する HTML5 グラフの作成
- 色モード (colorMode)
- 色スケールの定義 (colorScale)
- ピクトグラムのサポート
- エラーバーの描画 (errorBars)
- HTML ツールヒントの生成 (htmlToolTip)
- グラフでのユーザ操作の定義 (interaction)
- マウスオーバーインジケータのフォーマット設定 (mouseOverIndicator)
- 参照線の描画 (referenceLines)
- バブルグラフおよび散布図での傾向線の描画 (trendline)

アニメーション (introAnimation)

introAnimation プロパティを使用して、グラフ描画時のアニメーションを有効または無効にしたり、アニメーションの継続時間を定義したりします。このプロパティを有効にすると、グラフの描画時にライザおよびマーカーがアニメーション化されます。棒ライザ、折れ線ライザ、面ライザのアニメーションはフレームの底辺から開始され、割り当て値に到達するとアニメーションが静止します。円形のライザ (例、バブルマーカー) は、内側から外枠の方向にアニメーション化されます。円項目は、時計回りに描画されます。アニメーションは、3D グラフおよび 2.5D グラフ (depth プロパティで奥行きが適用されたグラフ) を除いて、すべてのグラフタイプに適用することができます。

構文　　グラフ描画時のアニメーションの有効化または無効化

```
"introAnimation": {
   "enabled": boolean,
   "duration": number},
```

説明

`"enabled": boolean`

グラフ描画時にアニメーションを有効または無効にします。有効な値には、次のものがあります。

- true - グラフ描画時にアニメーションを有効にします。
- false - グラフ描画時にアニメーションを無効にします。これがデフォルト値です。

`"duration": number`

アニメーションの継続時間を定義する数値をミリ秒で指定します。数値を大きくすると、アニメーションの動作が遅くなります。デフォルト値は 1400 (1.4 秒) です。

モーフアニメーションの継続時間 (morphAnimation)

ビジュアライゼーションモードでユーザが複数のフィルタを選択した場合、ライザおよびマーカーが滑らかに変形 (モーフィング) しながら、フィルタで生成された新しい値への変化がアニメーション表示されます。このモーフアニメーション効果の継続時間は、グラフ全体のプロパティ "morphAnimation": "duration" を使用して制御することができます。

構文　モーフアニメーションの継続時間の制御

```
"morphAnimation": {
  "duration": number
}
```

説明

`"duration": number`

モーフィング効果をアニメーション表示する時間の長さです。ミリ秒単位で指定します。デフォルト値は 1500 です。

データページスライダの速度制御

slider 属性カテゴリにソートフィールドを追加した場合、またはツールの animate カテゴリにソートフィールドを追加した場合、グラフ出力結果にデータページスライダが表示されます。再生ボタンをクリックすると、スライダコントロールがスライダバー上をデフォルト遅延 1000 ミリ秒で移動します。このアニメーション速度は速くしたり遅くしたりすることができます。

グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

構文　データページスライダの速度制御

```
"dataPageSlider": {
    "animateButton": {
      "delay": number                    }
                              },
```

説明

`"delay": number`

スライダがフレーム間を移動する遅延時間をミリ秒単位で指定します。デフォルト値は 1000 です。

JSON を使用したグラフイベント登録

HTML5 グラフでは、DOM (Document Object Model) イベントを検知し (例、グラフ上のオブジェクトのクリック)、そのイベントを使用してコールバック関数の呼び出しや URL への移動などのレスポンスを発動することができます。

構文　JSON を使用したグラフイベント登録

イベントディスパッチャには、イベントの配列を記述することができます。イベント定義に列記されたプロパティに一致する DOM ノードでイベントが発生すると、各イベントがそれぞれに対応するレスポンスを発動します。

```
"eventDispatcher": {
 "events" = [
  {response, event: 'string', object: 'string',
   userInfo: {custom: 'string'}, series: number, group: number,
   row: number, col: number,
   misc: 'string', userInfo: {custom: 'string'},
   context:  context, target:  'string' },
   .
   .
   .

 ]
}
```

説明

`response, ...`

必須

イベントに対応するレスポンスです。このイベントのパラメータは、イベント配列のこのエントリで定義されます。有効なレスポンスには次のものがあります。

- コールバック関数の呼び出し

```
callback: function(parm){function_code;}
```

説明

`function(parm){function_code;}`

関数によって呼び出される関数パラメータおよびコードです。

- URL への移動

```
"url": "link_to_url"
```

説明

`"url": "link_to_url"`

移動先の URL です。

`event: 'string'`
必須

DOM イベントを文字列で指定します (例、'mouseover'、'click'、'keyup')。イベント配列で定義されたこのイベントの発生時にレスポンスが発動されます。

`object: 'string'`
必須

イベントを登録するグラフオブジェクトのオブジェクト ID を文字列で指定します (例、'riser')。

`series: number`
オプション

ライザシリーズのシリーズ ID 番号です。この値を定義しない場合、すべてのシリーズが対象になります。

`group: number`
オプション

ライザグループのグループ ID 番号です。この値を定義しない場合、すべてのグループが対象になります。

`row: number`
オプション

マトリックスの行 ID です。この値を定義しない場合、すべての行が対象になります。

`col: number`
オプション

マトリックスの列 ID です。この値を定義しない場合、すべての列が対象になります。

`misc: 'string'`
オプション

その他のオブジェクト ID です (例、'marker')。この値を定義しない場合、その他のすべてのオブジェクト ID が対象になります。

`userInfo: {custom: 'string'}`
オプション

イベントコールバックに返すユーザ定義カスタムオブジェクトです。

`context: context`
オプション

イベントコールバックで 'this' を定義します。この値を定義しない場合、'this' は呼び出し元グラフインスタンスを表します。

`target: 'string'`
オプション

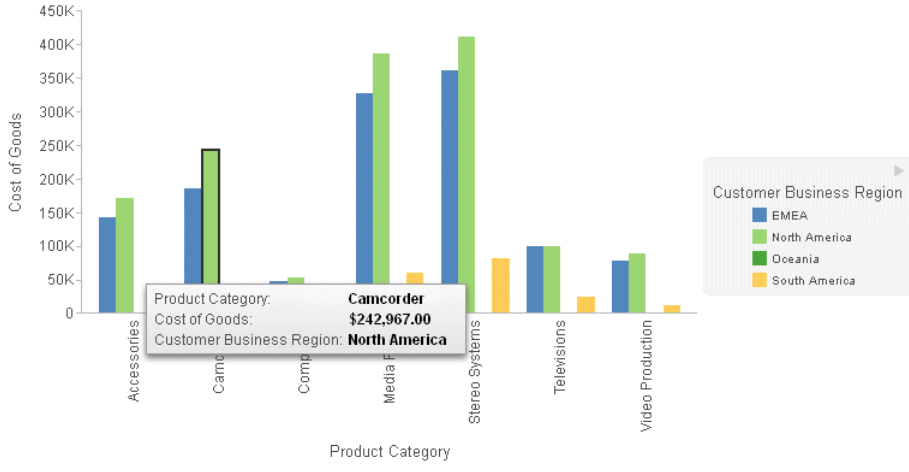
"url" プロパティに関連付けるターゲットです。URL クリックイベントを発動する際に使用します。デフォルト値は '_blank' です。この設定では、ユーザがリンクをクリックした際に、ブラウザに新しいブラウザタブまたはウィンドウが作成されます。

例　　グラフイベントの登録

次のリクエストは、2 つのイベントを登録します。1 つ目のイベントは、ユーザがグループ 1 のシリーズ 1 をクリックした際に、イベントパラメータのログをブラウザの Web コンソールに記録します。2 つ目のイベントは、ユーザがシリーズ 2 のライザをクリックした際に、TIBCO の Web サイトに移動します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis,$
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=color,$
TYPE=DATA, COLUMN=cogs_us, BUCKET=y-axis,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"eventDispatcher": {
"events": [
    {callback: function(a){console.log(a);},
   event: 'click', object: 'riser',
   userInfo: {custom: 'Custom1'}, series: 1, group:1},
 {"url": "http://www.tibco.com", target: '_blank',
    event: 'click', object: 'riser', series: 2},
    ]
}
*END
ENDSTYLE
END
```

下図は、グラフ出力でグループ 1 のシリーズ 1 をクリックした際の結果を示しています。



下図は、クリックイベントに応答して、パラメータのログがブラウザの Web コンソールに記録された結果を示しています。

グラフ上の注釈

注釈は、グラフ上の任意の位置に配置可能なテキスト、マーカー、またはその両方です。複数の注釈は、annotations 配列内で定義することができます。

グラフの注釈では、任意の SVG オブジェクトを指定し、その SVG オブジェクトのイベントとツールヒントを定義する機能がサポートされます。この機能を使用して、グラフキャンバス上のボタンやアイコンにユーザ独自のインタラクティブ性を付加することができます。

注釈にインタラクティブ性を付加するには、リクエストに別の eventDispatcher オブジェクトを追加する必要があります。詳細は、771 ページの「JSON を使用したグラフイベント登録」を参照してください。

annotations プロパティおよびそのデフォルト値は次のとおりです。

```
"annotations": [
 "position": {
   "x": undefined,
   "y": undefined,
   "points": undefined,
   "parent": "background"
        },
 "marker": {
   "visible": true,
   "color": "red",
   "size": 8,
   "shape": "circle",
   "rotation": 0,
   "border": {
     "width": 1,
     "color": "black",
     "dash": ""
           }
     "tooltip": undefined
        },
 "line": {
              "width": 1,
              "color": undefined,
              "dash: ""
          },
 "label": {
    "visible": false,
    "text": "Annotation Label",
    "font": "10pt Sans-Serif",
    "color": "black",
    "position": "center"
     }
 },
]
```

構文　　グラフ上への注釈の配置

```
"annotations": [
 "position": {
   "x": x,   "y": y,
   "points": [{"x":x, "y":y},...},
   "parent": "string"
    },
 "background": {
     "color" : "string",
     "border" : {"width": number, "color": "string"},
 "marker": {
   "visible": boolean,
   "color": "string",
   "size": number,
   "shape": "string",
   "position": "string",
   "rotation": number,
   "border": {
     "width": number,
     "color": "string",
     "dash": "string"
         }
     "tooltip": "string"
     },
 "line": {
               "width": number,
               "color": "string",
               "dash": ""
           },
 "label": {
        "visible": boolean,
        "text": "string",
        "font": "string",
        "color": "string",
        "position": "string"
    }
 },
   .
   .
   .
]
```

説明

"position":

注釈マーカーの位置を定義します。

"x": x

注釈アイコンの横方向の位置です。"x" の同義語は "left" です。この場合、フレームの左端を基点として横方向の位置を計測します。"right" を使用することもできます。この場合、フレームの右端を基点として位置を計測します。

デフォルト値は undefined です。 次のいずれかの値を指定します。

- 数値。"parent" で指定した親グラフコンポーネントが "chart" の場合、この数値は、数値軸の値を表します。それ以外の場合、"x" または "left" では親コンポーネントの左端からのピクセル数、"right" では右端からのピクセル数を表します。
- 横方向グラフのシリーズおよびグループラベル ({series:n,group:m} 形式)。注釈は、そのシリーズまたはグループに配置されます。この場合、親コンポーネントを "chart" にする必要があります。
- パーセント文字列 (例、"40%")。注釈は、"x" または "left" の場合は親コンポーネントの左端から横方向にパーセント指定された位置、"right" の場合は右端から横方向にパーセント指定された位置を表します。
- ピクセル文字列 (例、"40px")。注釈は、"x" または "left" の場合は親コンポーネントの左端から横方向にピクセル指定された位置、"right" の場合は右端から横方向にピクセル指定された位置を表します。

複数の指定方法で横方向の位置プロパティを設定した場合、"x" が "right" および "left" より優先され、"left" が "right" より優先されます。

"y": y

注釈アイコンの縦方向の位置です。"y" の同義語は "bottom" です。この場合、フレームの下端を基点として縦方向の位置を計測します。"top" を使用することもできます。この場合、フレームの上端を基点として位置を計測します。

デフォルト値は undefined です。 次のいずれかの値を指定します。

- 数値。"parent" で指定した親グラフコンポーネントが "chart" の場合、この数値は、数値軸の値を表します。それ以外の場合、"y" または "bottom" では親コンポーネントの下端からのピクセル数、"top" では上端からのピクセル数を表します。
- 縦方向グラフのシリーズおよびグループラベル ({series:n,group:m} 形式)。注釈は、そのシリーズまたはグループに配置されます。この場合、親コンポーネントを "chart" にする必要があります。

- パーセント文字列 (例、"40%")。注釈は、"y" または "bottom" の場合は親コンポーネントの下端から縦方向にパーセント指定された位置、"top" の場合は上端から縦方向にパーセント指定された位置を表します。
- ピクセル文字列 (例、"40px")。注釈は、"y" または "bottom" の場合は親コンポーネントの下端から縦方向にピクセル指定された位置、"top" の場合は上端から縦方向にピクセル指定された位置を表します。

複数の指定方法で縦方向の位置プロパティを設定した場合、"y" が "bottom" および "top" より優先され、"bottom" が "top" より優先されます。

`"points": [{"x":x, "y":y},...]`

任意の線または形状を描画するために使用される {x, y} 座標の配列です。"position": "x" および "y" が undefined の場合のみ使用されます。

`"parent": "string"`

次のグラフコンポーネントのいずれかです。

- "background" - グラフフレームの左下を基点として位置を計測し、注釈をグラフの背後に配置します。これがデフォルト値です。
- "chart" - グラフ描画領域の左下を基点として位置を計測し、注釈をグラフ上に配置します。
- "legend" - 凡例領域の左下を基点として位置を計測し、注釈を凡例領域に配置します。
- "xaxis" - X 軸に沿って位置を計測します。
- "yaxis" - Y 軸に沿って位置を計測します。
- "series": *number* - ライザのエンドキャップ上に形状を追加することができます。この場合、*number* は整数のシリーズ ID または文字列 "all" のいずれかです。設定した場合、選択したシリーズの各ライザまたはグラフ内のすべてのライザ ("all" の場合) にこの注釈が描画されます。

"background"
注釈の周囲に長方形を描画し、次の塗りつぶし色および境界プロパティを使用します。

"color": "string",
形状の周囲に描画される長方形の塗りつぶし色を指定する色指定文字列です。デフォルト値は "red" です。

"border"
形状の周囲に描画される長方形の境界プロパティを定義します。

"width": number
形状の周囲に描画される長方形の境界線の幅をピクセル単位で指定します。デフォルト値は 1 です。

"color": "string"
形状の周囲に描画される長方形の境界線の色を定義する色指定文字列です。デフォルト値は "black" です。

"marker":
注釈マーカーのプロパティを定義します。

"visible": boolean
注釈マーカーを表示するかどうかを定義します。有効な値には、次のものがあります。

- true - 注釈マーカーを表示します。これがデフォルト値です。
- false - 注釈マーカーを表示しません。

"color": "string"

注釈マーカーの色を定義します。色名または数値指定の文字列で定義された色、または文字列で定義されたグラデーションを指定します。デフォルト値は "red" です。

"size": number

注釈マーカーの直径をピクセル単位で指定します。"shape" が SVG パスを表す場合は無視されます。デフォルト値は 8 です。

"shape": "string"

注釈マーカーの形状です。有効なマーカー形状がすべてサポートされます。

"axisArrow" は、親である "xaxis" または "yaxis" にアンカー表示される注釈固有のマーカー形状です。軸に沿って矢印が描画され、軸の正の方向を指すように自動的に回転されます。axisArrow の色とスタイルは境界のプロパティを使用して設定できます。axisArrow マーカーに境界線が描画されない場合は、関連する軸線のスタイルが適用されます。

このマーカーには、アイコンの形状とサイズを定義する SVG パスデータ文字列 ('d' 属性) を指定することもできます。デフォルト形状は "circle" です。

マーカー形状のリストについては、427 ページの 「 シリーズ固有のプロパティ 」 を参照してください。

`"scale": number`

マーカーを拡大 (scale > 1 の場合)、または縮小 (0 < scale < 1 の場合) します。これは "size" とは異なり、ビルトインのマーカー形状だけでなく、イメージおよび任意の SVG パスなどすべてのマーカー形状で使用できます。

`"centerOffset"`

次のプロパティを使用して、SVG 形状がアンカー表示される位置を定義します。

`"x": value`

SVG 形状の幅に関連するピクセル数またはパーセント文字列です。

`"y": value`

SVG 形状の高さに関連するピクセル数またはパーセント文字列です。

`"position": "string"`

ブレットグラフの場合に、マーカーの位置をデータテキストラベルの相対位置として定義し、"top"、"bottom"、"middle" のいずれかを指定します。その他のグラフタイプではサポートされません。

`"rotation": number`

マーカーの回転角度を 0 (ゼロ) から 360 までの数値で定義します。

`"border":`

マーカー境界線のプロパティを定義します。

`"width": number`

境界の幅を定義する数値をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string"`

マーカー境界色を色名または数値指定の文字列で定義します。デフォルト値は "black" です。

`"dash": "string"`

境界線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。デフォルト値は「""」です (実線)。

`tooltip`

注釈マーカーのツールヒントです。デフォルト値は undefined です。

"line":
座標が 1 組の xy 値以外で指定された場合に、描画される線のプロパティを定義します。"position" に複数の座標が含まれる場合のみ使用されます。次のプロパティは、これらの座標を接続するために使用する線のスタイルを指定します。

"width": number
線幅をピクセル単位で指定します。デフォルト値は 1 です。

"color": "string"
線の色を指定する文字列です。デフォルト値は undefined です。

"dash": "string"
線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。デフォルト値は「""」です (実線)。

"label":
注釈ラベルのプロパティを定義します。

"visible": boolean
注釈ラベルを表示するかどうかを定義します。有効な値には、次のものがあります。

- true - 注釈ラベルを表示します。これがデフォルト値です。
- false - 注釈ラベルを表示しません。

"text": "string"
注釈ラベルのテキストを定義します。ラベルテキストのデフォルト値はありません。

"font": "string"
注釈ラベルのフォントを定義します。デフォルト値は "10pt Sans-Serif" です。

"color": "string"
注釈ラベルの色を色名または数値指定の文字列で定義します。デフォルト値は "black" です。

"position": "string"
注釈ラベルの位置を、注釈マーカーに対する相対位置で定義します。有効な値には、次のものがあります。

- "top" - 注釈ラベルを注釈マーカーの上側に表示します。
- "bottom" - 注釈ラベルを注釈マーカーの下側に表示します。
- "left" - 注釈ラベルを注釈マーカーの左側に表示します。
- "right" - 注釈ラベルを注釈マーカーの右側に表示します。

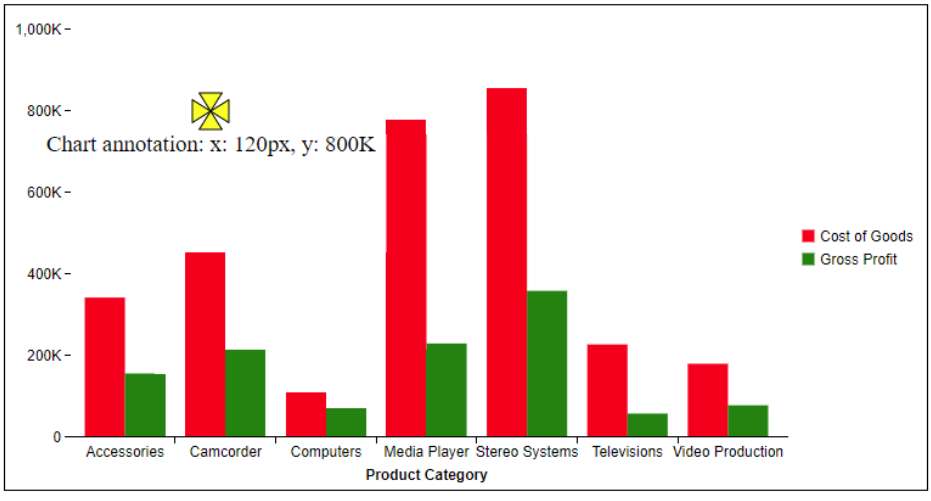
- "center" - 注釈ラベルを注釈マーカーの中央に表示します。これがデフォルト値です。

例　注釈の追加

次のリクエストでは、1 つの注釈を指定します。注釈は、グラフコンポーネント上に配置します。この注釈のマーカーは、黒色の境界で囲まれた黄色のパイレーツクロスで、X 軸の左端から横方向に 120 ピクセル、Y 軸 (数値) の 800K の位置に配置されます。このラベルは、マーカーの下側に配置されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *

ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS_FINAL
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}},
"annotations": [
    {"position": {"x": "120px", "y": 800000, "parent": "chart"},
        "marker": {"visible": true, "color": "yellow",
            "size": 30, "shape": "piratecross", "rotation": 0,
            "border": {"width": 1, "color": "black", "dash": ""}},
        "label": {"visible": true, "text": "Chart annotation: x: 120px, y: 800K",
            "font": "14pt Times New Roman", "color": "black", "position": "bottom"}},
        ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

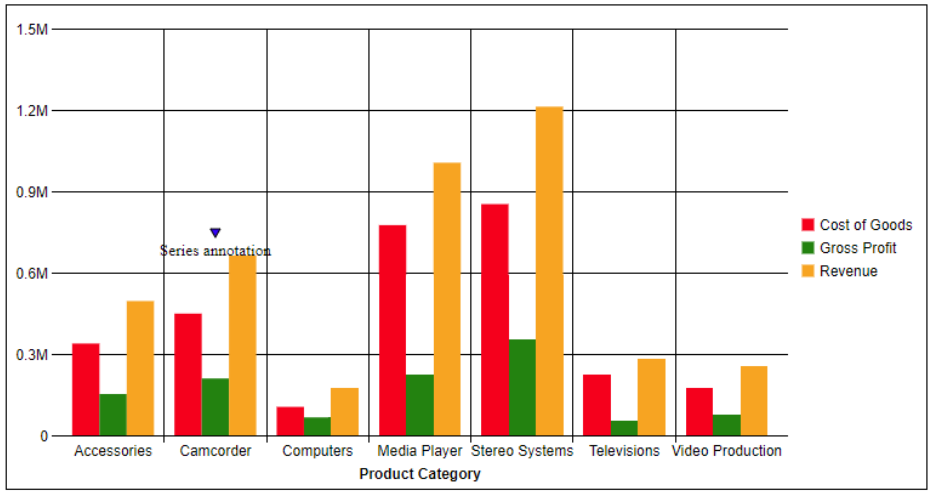


例　シリーズおよびグループに関連する注釈の配置

次のリクエストは、シリーズ 1 のグループ 1 に対して注釈を 1 つ配置します。y 値は、Y 軸上の 50% の位置です。マーカーは、青色の三角形です。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
*GRAPH_JS
"annotations": [{
   "position": {"x": {"series": 1, "group": 1},
      "y": "50%", "parent": "chart"},
   "marker": {"visible": true, "color": "blue",
      "shape": "triangle", "rotation": 0,
      "border": {"width": 1, "color": "black", "dash": ""}},
   "label": {"visible": true, "text": "Series annotation",
      "font": "10pt Times New Roman", "color": "black", "position":
"bottom"}
   }]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

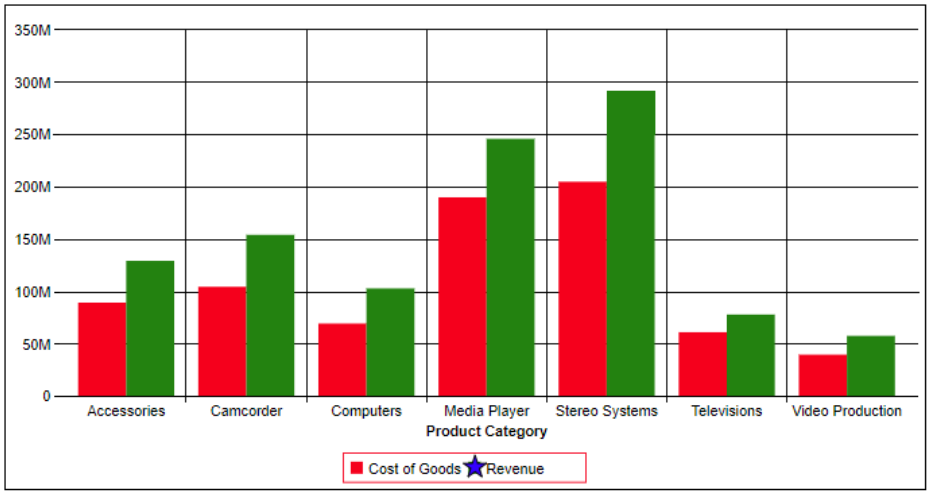


例　凡例領域への注釈の配置

次のリクエストは、凡例領域の緑色の凡例マーカーの手前 (前面) に青色の星印を配置します。

```
GGRAPH FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"legend": {"position": "bottom", "lineStyle": {"color": "red"}},
"annotations": [{
   "position": {
      "x": 109,
      "y": 12,
      "parent": "legend"},
   "marker": {"visible": true, "color": "blue", "size": 20,
      "shape": "fiveStar", "rotation": 0,
      "border": {"width": 1, "color": "black", "dash": ""}}
   }]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

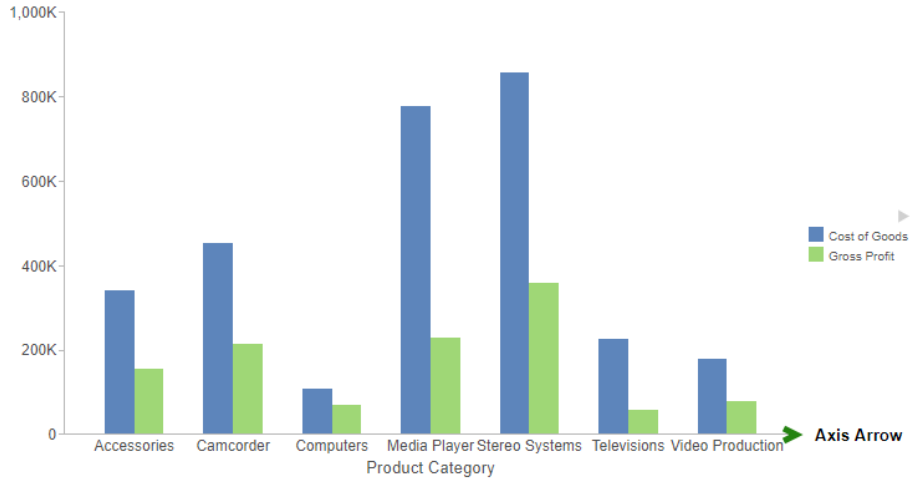


例　軸注釈アンカーの使用

次のリクエストは、注釈を X 軸上にアンカー表示します。注釈マーカーは、緑色、20 ピクセル、境界幅 5 ピクセルの axisArrow マーカーで、X 軸の終点に配置されます。ラベルテキストの「Axis Arrow」は、マーカーの右側に表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=gross_profit_us, bucket=y-axis,$
type=data, column=n1, bucket=x-axis,$
*GRAPH_JS
"annotations": [{
  "position": {
     "x": "100%",
     "parent": "xaxis"},
  "marker":
    {"visible": true,
    "shape": "axisArrow", "rotation": 0,
    "size": 20,
    "border": {"width": 5, "color": "green", "dash": ""}
},
  "label": {"visible": true, "text": "Axis Arrow",
    "font": "bold 10pt Arial", "color": "black", "position": "right"},
}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

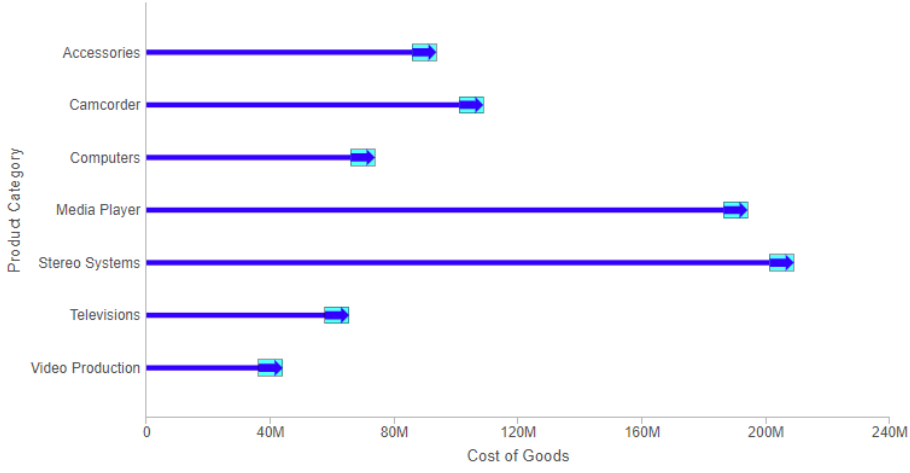


例　棒グラフライザのエンドキャップとしての標準マーカー形状の使用

次のリクエストは、横棒グラフを生成し、ライザのエンドキャップとして矢印のマーカー形状を使用します。また、ライザとマーカーの色を青色に変更し、マーカー周囲にシアンの長方形を描画します。マーカーは、グラフの向きに合わせて 90 度回転されます。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-ORIENTATION=horizontal,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"blaProperties": {
"barGroupGapWidth": 0.9,
          },
"series": [ {"series": 0,"color": "blue"}],
"annotations": [{
  "position": {
   "y": "100%",
   "parent": "series 0"
    },
  "background": {"color": "cyan", "border": {"color": "black", "width":
2}},
  "marker": {
   "visible": true,
   "color": "blue",
   "size": 100,
   "scale": 0.2,
   "shape": "arrow",
   "rotation": 90,
  }
 }],
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



例　棒グラフライザのエンドキャップとしての SVG イメージの使用

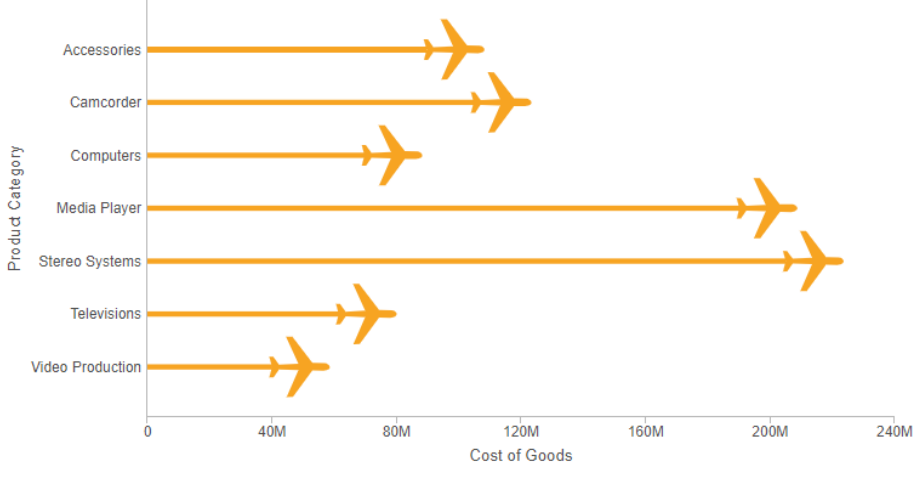
次のリクエストは、横棒グラフを生成し、ライザのエンドキャップとして SVG 形状を使用します。また、ライザとマーカーの色をオレンジ色に変更します。マーカーは、グラフの向きに合わせて 90 度回転されます。

注意：このマニュアルでは、マージン内に収まるようマーカー形状の定義が複数行に分割されています。このリクエストを実行する場合は、改行を削除する必要があります。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-ORIENTATION=horizontal, $
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"blaProperties": {
"barGroupGapWidth": 0.9,
},
 "series": [ {"series": 0,"color": "orange"}],
annotations: [{
  "position": {
   "y": "100%",
   "parent": "series 0"
  },
  "marker": {
   "visible": true,
   "color": "orange",
   "size": 100,
   "scale": 0.2,
   "shape": "M 247.51404,152.40266 139.05781,71.800946 c 0.80268,-12.451845
            1.32473, -40.256266 0.85468,-45.417599 -3.94034,-43.266462
            -31.23018,-24.6301193 -31.48335, -5.320367 -0.0693,5.281361
            -1.01502,32.598388 -1.10471,50.836622 L 0.2842717,
            154.37562 0,180.19575 l 110.50058,-50.48239 3.99332,80.29163
            -32.042567,22.93816 -0.203845,16.89693 42.271772,
            -11.59566 0.008,0.1395 42.71311,10.91879 -0.50929,
            -16.88213 -32.45374,-22.39903 2.61132,-80.35205 111.35995,
            48.50611 -0.73494,-25.77295 z",
   "rotation": 90,
   "centerOffset": {
    "x": "-50%",
    "y": "-95%"
   },
```

```
"border": {
    "width": 0,
    "color": "transparent",
    "dash": ""
   },
  }
 }],
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



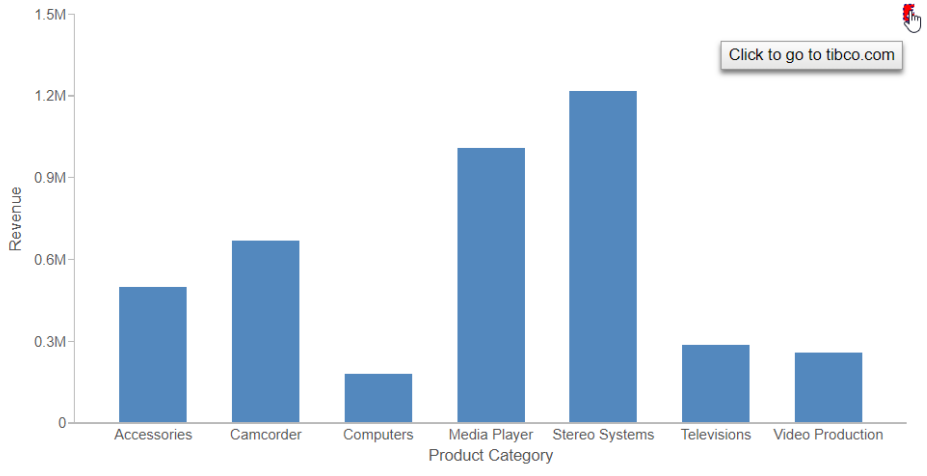
例　グラフのインタラクティブ注釈の定義

次のリクエストは、赤色のアイコンおよび紺色の破線境界で構成された注釈を作成します。このアイコンのクリック操作に関連付けられたイベントにより、TIBCO ホームページが開きます。ツールヒントには、次のテキストが表示されます。

```
Click to go to tibco.com
```

```
SET PAGE-NUM=NOLEAD
SET HTMLENCODE=OFF
SET EMBEDHEADING=ON
GRAPH FILE wf_retail_lite
SUM REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
*GRAPH_JS
   "eventDispatcher": {
        "events": [
            {
                // This will add a click  event handler to the annotation
                "event": "click", object: "annotation", series: 0, url:
"http://tibco.com", target: "_blank"
            }
        ]
    },

annotations:[{
      "position": {
            "x": 750,
            "y": 12,
        "parent": "background"
    },
    "marker": {
    "visible": true,
    "color": "red",
    "border": {"width": 1, "color": "navy", "dash": "2 2"},
       "shape": "m 12.666667,8.291667 0,-4.375 c 0,-0.319084
-0.26425,-0.583334 -0.583334,-0.583334 l -4.3749997,0 C 7.4715,3.333333
7.2615,3.479167 7.1705,3.688583 7.0795,3.907333 7.125,4.16225
7.29825,4.32675 l 1.3125,1.3125 -4.8673333,4.867333 c -0.2280834,0.228084
-0.2280834,0.592667 0,0.820167 L 4.67325,12.256583 c 0.2280833,0.228084
0.5926667,0.228084 0.8201667,0 L 10.36075,7.38925 l 1.3125,1.3125 c
0.109083,0.118417 0.2555,0.17325 0.410083,0.17325 0.07292,0
0.155167,-0.01808 0.228084,-0.0455 0.209416,-0.091 0.35525,-0.301
0.35525,-0.537833 z M 15,3.625 l 0,8.75 C 15,13.824 13.824,15 12.375,15 l
-8.75,0 C 2.176,15 1,13.824 1,12.375 L 1,3.625 C 1,2.176 2.176,1 3.625,1 l
8.75,0 C 13.824,1 15,2.176 15,3.625 Z",
        "tooltip": "Click to go to tibco.com"
    }
}]
*END
 ENDSTYLE
END
```

出力結果は次のとおりです。



PDF、Excel、PowerPoint、PNG フォーマットで使用する HTML5 グラフの作成

HTML5 グラフ出力は通常ブラウザで生成され、高度なインタラクティブ機能を提供します。ただし、以前のバージョンでは、このタイプの出力は .png または .pdf など他のフォーマットで生成することができませんでした。そのため、ユーザが静的なフォーマットでグラフ出力を生成する場合は、マップや他のグラフ拡張機能など HTML5 グラフで利用可能な新機能をサポートしない、レガシーグラフ API エンジンを使用する必要がありました。

サーバサイドグラフ機能には、PDF、PNG、DHTML、XLSX、PPTX のフォーマットで HTML5 グラフ出力を生成する機能が追加されています。これらの静的なフォーマットでは HTML5 グラフのインタラクティブ機能は使用できませんが、マップおよびグラフ拡張機能はサポートされます。生成された出力ファイルは、Reporting Server によって別のフォーマットで生成されない限り、ブラウザに表示されたとおりにグラフを表示します。

参照　サーバサイド HTML5 グラフ出力の使用上の注意

- 現在のバージョンでは、HTML5 グラフのサーバサイドグラフ機能は、Intel x86-64 CPU で動作する Windows および Linux 環境の Reporting Server でのみサポートされます。

- サーバにカスタム PDF フォント (PFA または PFB および AFM ファイルを使用) をインストールし、fontuser.xml で参照する場合、このフォントはサーバサイド HTML 5 グラフ出力で使用することができません。これは、グラフが、Reporting Server の Headless ブラウザ (PhantomJS) により PNG ファイルとして生成され、ビットマップイメージとして PDF 出力に埋め込まれるためです。このため、サーバサイド HTML5 グラフで使用するフォントは、システムフォントとしてインストールしておく必要があります。たとえば、Reporting Server が Windows 上で実行されている場合、TTF (TrueType フォント) を Windows フォントとして Reporting Server マシンにインストールする必要があります。この場合、Windows のエクスプローラで .ttf ファイルのコンテキストメニューから [インストール] オプションを設定します。
- グラフ出力のサイズを指定する HAXIS、VAXIS、UNITS パラメータがサポートされます。
- サーバサイド HTML5 グラフ出力は静的です。ズームなどのボタンが表示される場合も、イメージのみで動作しません。
- サーバサイドグラフ出力は、アンチエイリアシング効果 (輪郭部のギザギザなど) を回避するため、デフォルト設定の解像度 192 PPI (インチ当たりピクセル数) を使用して生成されます。解像度の設定は、SET GRAPH-PPI コマンドを発行することで変更できます。構文は次のとおりです。

  ```
  SET GRAPH-PPI = n
  ```

  グラフリクエストでは次のとおりです。

  ```
  ON GRAPH SET GRAPH-PPI n
  ```

  説明

  `n`

  出力の解像度をインチ当たりのピクセル数で指定する 72 から 1200 までの整数です。デフォルト値は 192 です。数字が大きいほど出力ファイルのサイズが大きくなります。
- HTML5 グラフは、複合レポートのコンポーネントとして使用できます。

構文　サーバサイドグラフ出力の有効化

サーバは、次の設定が有効な場合に、従来のグラフ出力ではなく HTML5 出力を生成する必要があると認識します。

```
SET GRAPHENGINE = JSCHART
```

デフォルト設定では、グラフリクエストでグラフ属性構文が使用された場合にこの設定が実行されます。グラフ属性構文を使用しないグラフリクエストでは、リクエストまたはサポート対象のプロファイルでこのコマンドを手動で発行する必要があります。発行しない場合は、レガシーグラフ API を使用して出力が生成されます。グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

構文　その他のフォーマットでの HTML5 グラフ出力の生成

インタラクティブな HTML5 グラフを生成する場合は、グラフリクエストに次のコマンドを追加します。

```
ON GRAPH PCHOLD FORMAT JSCHART
```

その他のフォーマットで HTML5 グラフ出力を生成する場合は、グラフリクエストに次のコマンドを追加する必要があります。

```
ON GRAPH PCHOLD FORMAT format
```

説明

`format`

次の出力フォーマットのいずれかです。

- PDF
- PNG
- DHTML
- XLSX
- PPTX

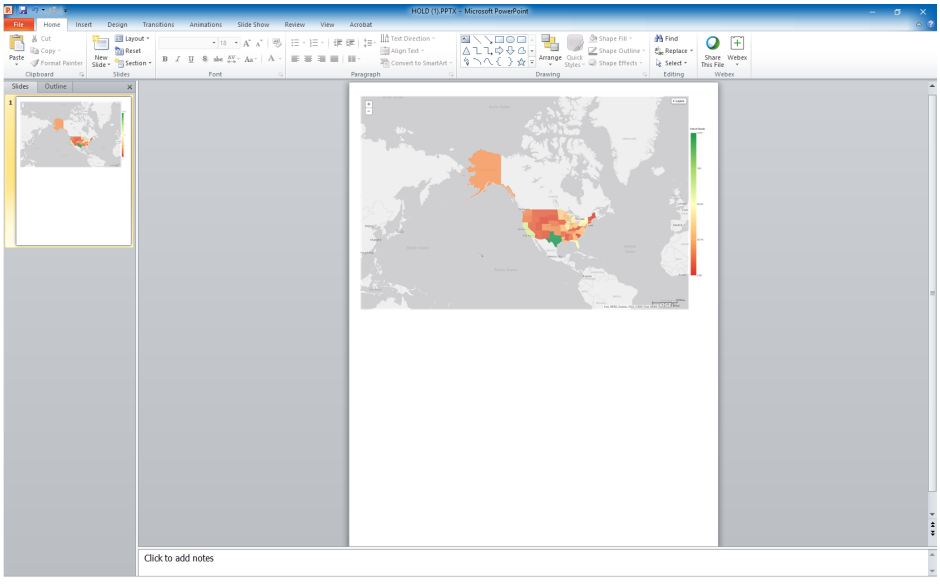
また、SET GRAPHENGINE = JSCHART コマンドを有効にする必要があります。この場合、デフォルト設定 (リクエストでグラフ属性構文を使用) で有効化することも、プロシジャ (FOCEXEC) またはプロファイルでこのコマンドを発行して有効化することもできます。

例 PowerPoint ファイルとしての ESRI マップグラフの生成

次のグラフリクエストは、ESRI マップを生成し、PowerPoint ファイルとしてこれをエクスポートします。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT PPTX
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions":
{
  "com.esri.map":
  {
    "overlayLayers":
     [
      {
      "ibiDataLayer": {
         "map-metadata": {"map_by_field": "STATE_PROV_NAME"}
                       }
      }
    ],
  "baseMapInfo":
    {"customBaseMaps": [{"ibiBaseLayer": "gray"}]}
  }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

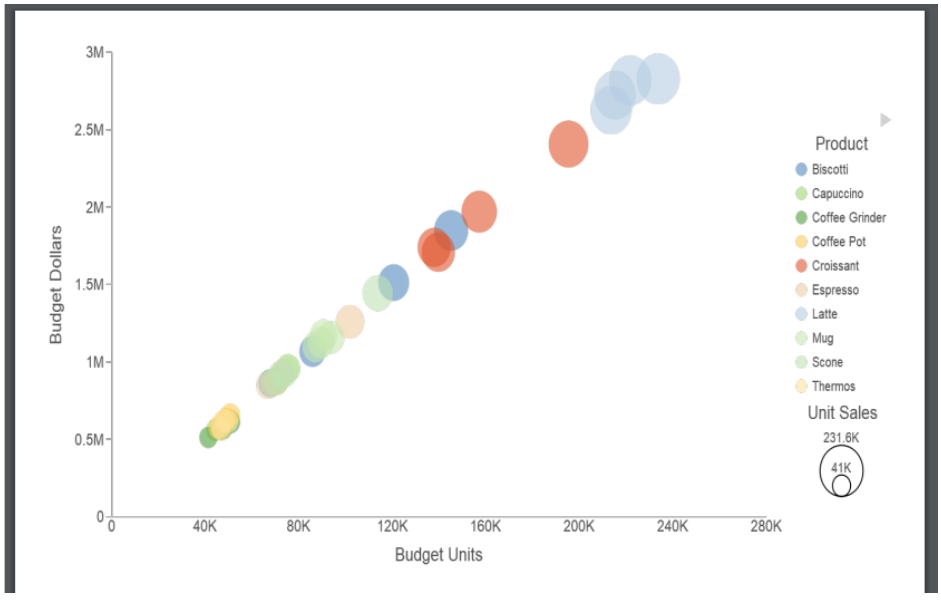
例

PDF フォーマットでの HTML5 グラフ出力の解像度の変更

次のリクエストは、デフォルト設定の解像度を使用し、PDF フォーマットでバブルグラフを生成します。

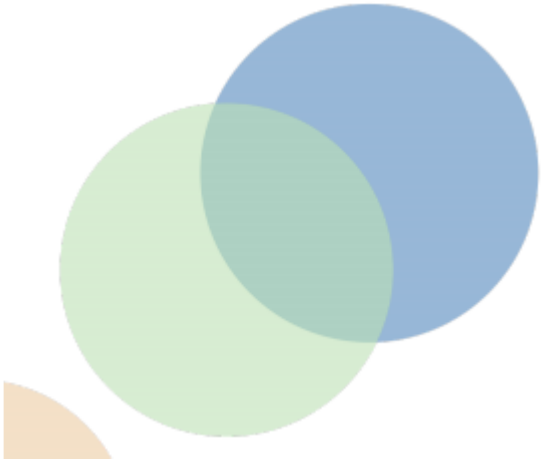
```
GRAPH FILE ibisamp/ggsales
SUM GGSALES.SALES01.BUDDOLLARS
GGSALES.SALES01.BUDUNITS
GGSALES.SALES01.UNITS
BY GGSALES.SALES01.PRODUCT
BY GGSALES.SALES01.REGION
ON GRAPH PCHOLD FORMAT PDF
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET AUTOFIT OFF
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-ORIENTATION=VERTICAL, CHART-SERIES-LAYOUT=side-by-side,
ARGRAPHENGINE=JSCHART, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
TYPE=DATA, COLUMN=N1, BUCKET=color, $
TYPE=DATA, COLUMN=N4, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N5, BUCKET=size, $
TYPE=DATA, COLUMN=N2, BUCKET=detail, $
ENDSTYLE
END
```

出力結果は次のとおりです。



ON GRAPH SET GRAPH-PPI 1200 コマンドを追加することで、鮮明なマーカーを生成することができます。違いを明確にするため、下図は、デフォルト設定の解像度を使用した場合の 2 つのバブルの拡大イメージを示しています。

下図は、リクエストに ON GRAPH SET GRAPH-PPI 1200 を追加した場合の、同じ 2 つのバブルの拡大イメージを示しています。

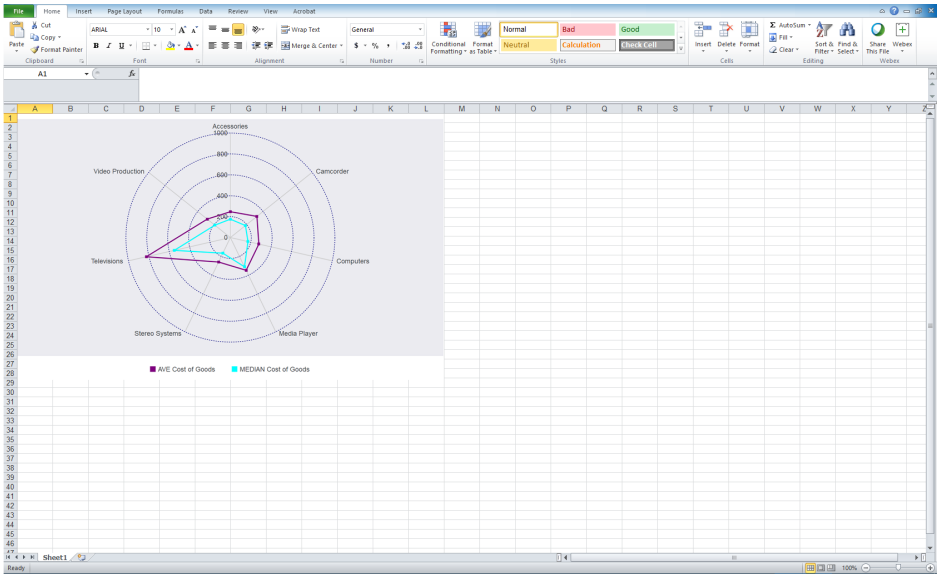


例　グラフ属性構文を使用しない HTML5 グラフの Excel 出力ファイルとしての生成

次のリクエストは、Excel 出力ファイルとしてレーダーグラフを生成します。リクエストではグラフ属性構文を使用しないため、ON GRAPH SET GRAPHENGINE JSCHART コマンドがリクエストに追加されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM AVE.COGS_US MDN.COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT XLSX
ON GRAPH SET LOOKGRAPH RADARL
ON GRAPH SET GRAPHENGINE JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,
$*GRAPH_JS
"legend": {"visible": true},
"polarProperties": {
"straightGridLines": false, "extrudeAxisLabels": false, "drawAsArea":
false},
"yaxis": {"numberFormat": "##","majorGrid": {"lineStyle": {
"width": 1, "color": "navy", "dash": "2 2"}}},
"series": [
{"series": 0, "color": "purple", "border": {"width": 2}},
{"series": 1, "color": "cyan", "border": {"width": 2}},
{"series": 2, "color": "grey", "border": {"width": 2}},
{"series": 3, "color": "teal", "border": {"width": 2}}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



色モード (colorMode)

colorMode プロパティは、グラフのライザおよびマーカーを描画する際の色モードを定義します。デフォルト構成では (bySeries)、シリーズのライザおよびマーカーには、"series": "color" プロパティで指定された値に基づいて色が適用されます。デフォルト値は、series: 0, color: red、series: 1, color: green、series: 2, color: orange です。このプロパティを使用して、グラフのライザおよびマーカーにさまざまな色モードを適用することができます。グラフタイプによっては、一部の色モードが適用されない場合や、グラフに適合しない場合があります。たとえば、滝型グラフのプロパティでは、正の値のライザと負の値のライザの色を定義します。

構文　　色モードの定義

```
"colorMode": {
   "mode": "string",
   "colorList": ["string",..."string"]
 },
```

説明

`"mode": "string"`

グラフの色モードを定義する文字列です。有効な値には、次のものがあります。

- "byGroup" - グループのライザおよびマーカーは、"series": "color" プロパティで指定された値に基づいて色分けされます。
- "byHeight" - colorList 配列で定義された色の組み合わせにより、グラデーションが作成されます。最初の色が数値軸の最小値の位置に使用され、最後の色が数値軸の最大値の位置に使用されます。残りの色は、最小値と最大値の間で補間されます。ライザの色には、このグラデーションが使用されます。byHeight モードは、X 軸と単一の数値軸で構成されたグラフでのみ有効です。
- "byInterpolation" - 最初のシリーズには colorList 配列の最初の色が使用され、最後のシリーズには colorList 配列の最後の色が使用されます。中間のシリーズでは色が補間されます。
- "byInterpolationAlt" - byInterpolation と同一ですが、このモードでは、指定されたシリーズ色の順序の入れ替えが行われます。このモードでは、隣接するライザに異なる色が使用されるため、ライザを区別しやすくなります。
- "bySeries" - シリーズのライザおよびマーカーは、"series": "color" プロパティで指定された値に基づいて色分けされます。 これがデフォルト値です (円グラフを除く)。
- "continuous" - ヒートマップおよびコロプレスマップでのデフォルト色モードです。このモードを使用すると、色スケールで複数の色が混合され、グラデーションになります。
- "discrete" - ヒートマップおよびコロプレスマップでこのモードを使用することで、不連続な色バンドに色スケールを表示することができます。色バンドの配列を定義し、色ごとに開始値と終了値を指定するには、このモードを colorScale オブジェクトで使用します。詳細は、819 ページの 「 色スケールの色モードの定義 」 を参照してください。
- "bin" - コロプレスマップおよびバブルマップでこのモードを使用することで、マーカーを使用して色バンドの凡例を表示することができます。色バンドの配列を定義し、色ごとに開始値と終了値を指定するには、このモードを colorScale オブジェクトで使用します。詳細は、819 ページの 「 色スケールの色モードの定義 」 を参照してください。

```
"colorList": ["string",..."string"]
```

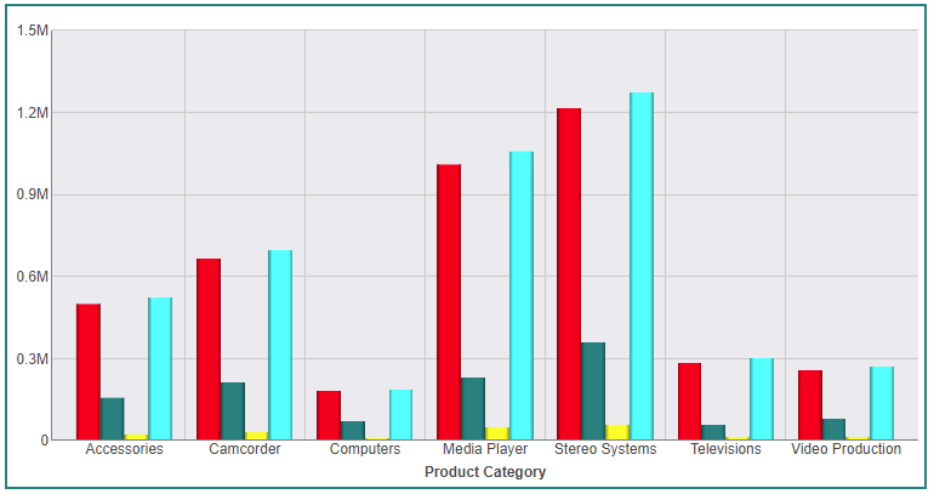
byHeight、byInterpolation、byInterpolationAlt 色モードの場合、色の配列を色名または数値指定の文字列で定義します。byHeight、byInterpolation、byInterpolationAlt で指定された色が 1 つの場合、その色の薄い色が、暗黙的な 2 つ目の色として使用されます。デフォルト値は undefined です。

例　色モードの設定

次のリクエストでは、デフォルトの色モード (bySeries) を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"legend": {"visible": false},
"series": [
   {"series": "all", "border": {"width": 1, "color": "grey"}},
   {"series": 0, "color": "red"},
   {"series": 1, "color": "teal"},
   {"series": 2, "color": "yellow"},
   {"series": 3, "color": "cyan"}]
*END
ENDSTYLE
END
```

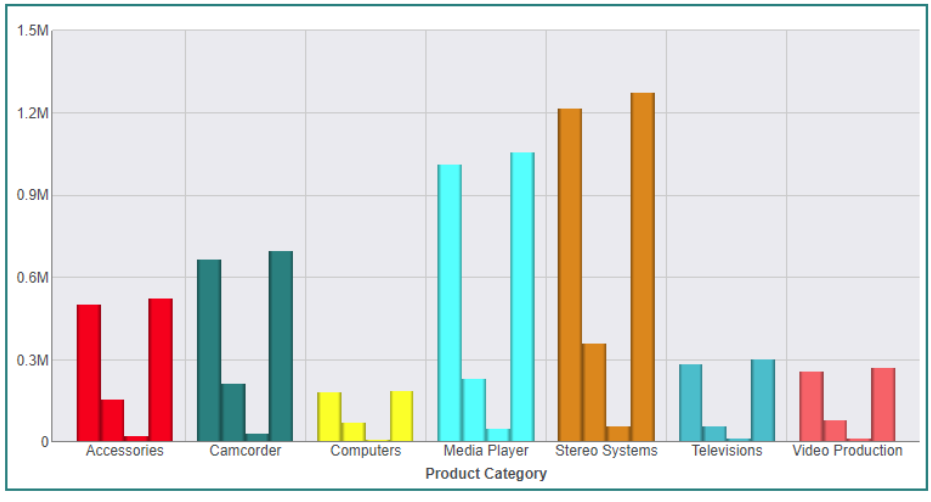
出力結果では、各シリーズにそれぞれ独自の色が適用され、その色が各グループ (PRODUCT_CATEGORY) で繰り返されています。



次のリクエストでは、色モードを byGroup に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"legend": {"visible": false},
"colorMode": {"mode": "byGroup"},
"series": [
   {"series": "all", "border": {"width": 1, "color": "grey"}},
   {"series": 0, "color": "red"},
   {"series": 1, "color": "teal"},
   {"series": 2, "color": "yellow"},
   {"series": 3, "color": "cyan"}]
*END
ENDSTYLE
END
```

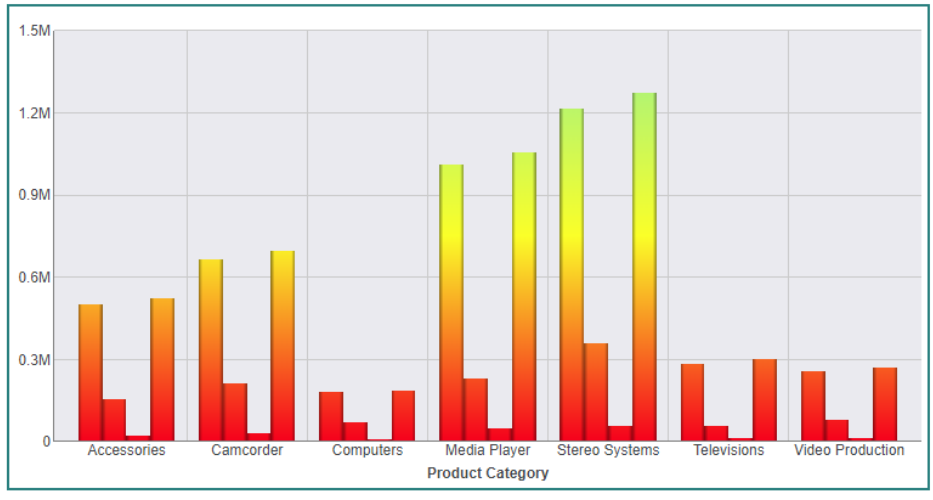
出力結果では、各グループ (PRODUCT_CATEGORY) のすべてのシリーズが、グループごとに同一色で表示されています。たとえば、グループ 0 にはシリーズ 0 の色セットが使用され、グループ 1 にはシリーズ 1 の色セットが使用されています。シリーズの色を使い切ると、デフォルトの色が使用されます。



次に色モードを byHeight に変更し、色リストを追加します。

```
"colorMode": {
    "mode": "byHeight",
    "colorList": ["red","yellow","lightgreen"]}
```

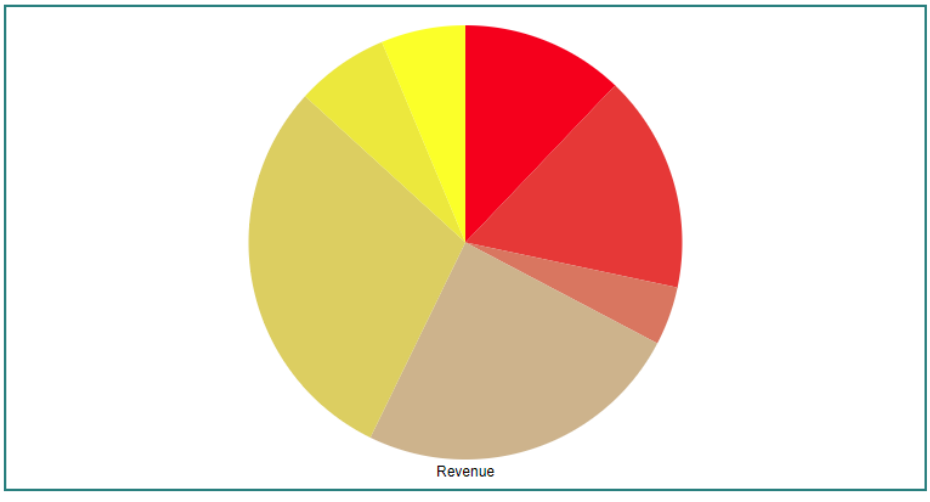
出力結果では、ライザの色にグラデーションが使用されています。最初の色が数値軸の最小値の位置、最後の色が数値軸の最大値の位置に使用され、残りの色が最小値と最大値の間で補間されます。



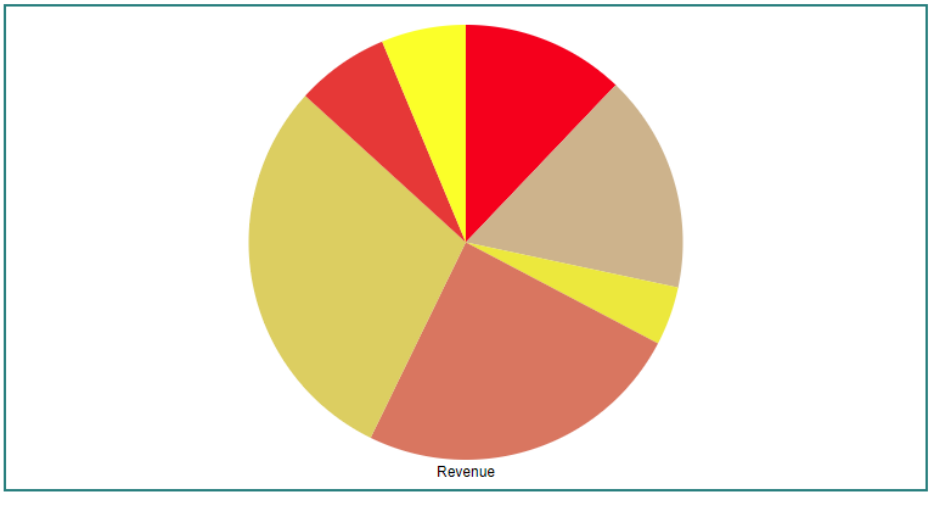
次のリクエストは、円グラフを生成し、byInterpolation 色モードで色リストを指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"legend": {"visible": false},
"colorMode": {
   "mode": "byInterpolation",
   "colorList": ["red", "tan", "yellow"]}
*END
ENDSTYLE
END
```

出力結果では、最初のシリーズに colorList 配列の最初の色、最後のシリーズに colorList 配列の最後の色が使用され、中間のシリーズでは色が補間されています。



色モードを byInterpolationAlt に変更すると、下図のグラフが生成されます。中間のシリーズでは色が補間され、色の順序が入れ替えられています。



色モードの色の定義

colorMode プロパティを "byInterpolation" または "byHeight" に設定した場合、colorModeColors プロパティを使用して、ライザおよびマーカーに適用する色の範囲を定義します。

- colorMode プロパティを "byInterpolation" に設定した場合、最初のシリーズにはこの配列で定義された最初の色、最後のシリーズには最後の色が使用され、中間のシリーズでは色が補間されます。
- colorMode プロパティを "byHeight" に設定した場合、ここで定義された色の組み合わせにより、グラデーションが作成されます。最初の色が数値軸の最小値の位置、最後の色が数値軸の最大値の位置に使用され、残りの色は最小値と最大値の間で補間されます。ライザの色には、このグラデーションが使用されます。

構文　色モードの色の定義

```
"colorModeColors": ["string",...,"string"]
```

説明

`["string",...,"string"]`

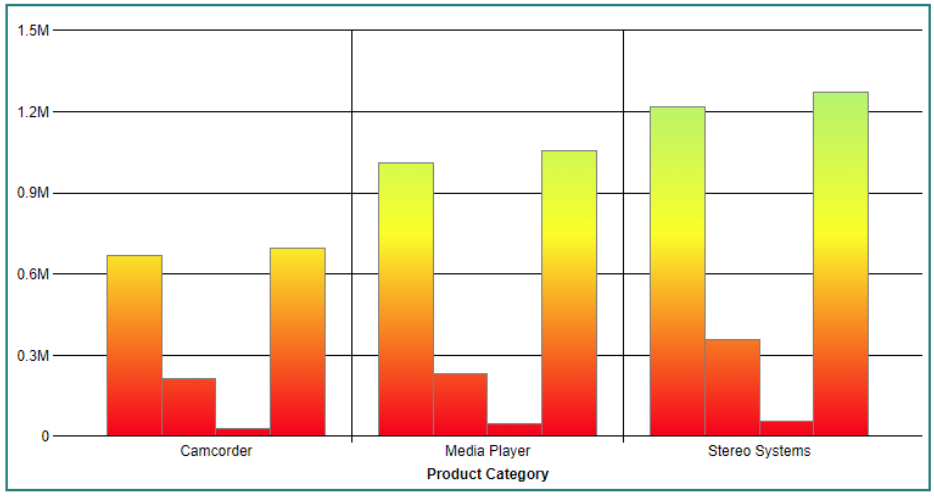
色名または数値指定の文字列で色の配列を定義します。デフォルト値は undefined です。

例　色モードの色の定義

次のリクエストは、縦棒グラフを生成し、colorMode を "byHeight"、colorModeColors を赤色、黄色、黄緑色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Camcorder' OR 'Media Player' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"legend": {"visible": false},
"colorMode": "byHeight",
"colorModeColors": ["red", "yellow", "lightgreen"],
"series": [
    {"series": "all", "border": {"width": 1, "color": "grey"}},
    {"series": 0, "color": "red"},
    {"series": 1, "color": "teal"},
    {"series": 2, "color": "yellow"},
    {"series": 3, "color": "cyan"}]
*END
ENDSTYLE
END
```

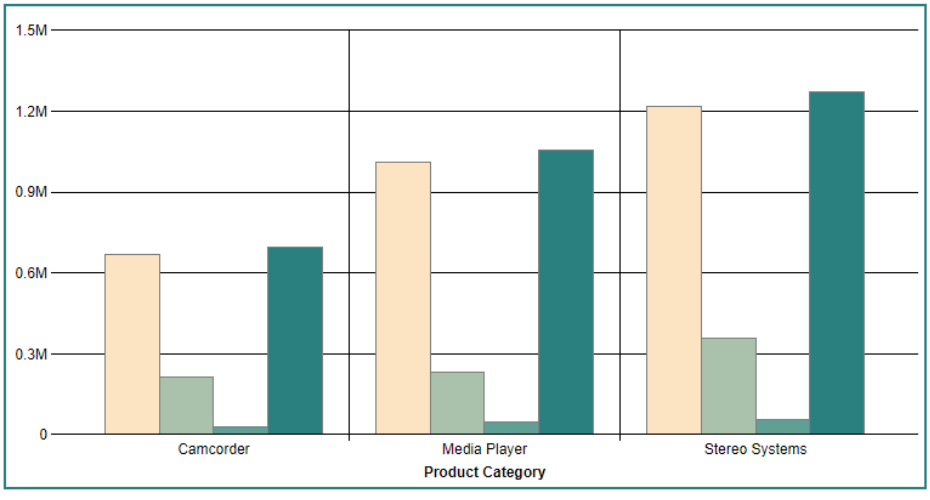
出力結果では、指定した色に基づいてライザにグラデーションが生成されています。最初の色が数値軸の最小値の位置、最後の色が数値軸の最大値の位置に使用され、残りの色が最小値と最大値の間で補間されています。



次のリクエストでは、colorMode が 'byInterpolation'、colorModeColors がビスクおよび青緑色に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US DISCOUNT_US MSRP_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Camcorder' OR 'Media Player' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=y-axis,$
type=data, column=n5, bucket=y-axis,$
*GRAPH_JS
"border": {"width": 2, "color": "teal"},
"legend": {"visible": false},
"colorMode": "byInterpolation",
"colorModeColors": ["bisque", "teal"],
"series": [
   {"series": "all", "border": {"width": 1, "color": "grey"}},
   {"series": 0, "color": "red"},
   {"series": 1, "color": "teal"},
   {"series": 2, "color": "yellow"},
   {"series": 3, "color": "cyan"}]
*END
ENDSTYLE
END
```

出力結果では、最初のシリーズに配列の最初の色、最後のシリーズに配列の最後の色が使用され、中間のシリーズでは色が補間されています。



色スケールの定義 (colorScale)

colorScale プロパティは、color 属性カテゴリにフィールドが割り当てられているすべてのグラフで色スケールの描画に使用する色を定義します。

ヒートマップまたはコロプレスマップでは、色の変化を不連続な色バンドとして表示することもできます。マップグラフの場合、ビンマーカーを使用してマップグラフの凡例を表示することもできます。詳細は、819 ページの 「 色スケールの色モードの定義 」 を参照してください。

構文　色スケールの定義

```
  "colorScale": {
    "title": {
            "text": "string",
                  "visible": boolean,
                  "font": "string",
                  "color": "string",
            "tooltip": "string"
                  "wrap": number
           },
    "colors": ["string", "string", ..., "string"]
},
```

または

```
"colorScale": {
  "title": {
            "text": "string",
                  "visible": boolean,
                  "font": "string",
                  "color": "string",
            "tooltip": "string"
                  "wrap": number
           },
  "colors":[
           {
            "start": value,
            "stop": value,
            "color": "string"
            },
            .
            .
            .
           ]
         }
```

説明

`"title"`

色スケール軸上のタイトルのフォーマットを設定します。次のプロパティを使用します。

`"text": "string"`

色スケール軸上に表示するテキストです。デフォルト値は "Color Axis Title" です。

`"visible": boolean`

次のいずれかの値を指定します。

- true - 色スケールタイトルを表示します。
- false - 色スケールタイトルを表示しません。これがデフォルト値です。

- "auto" - 色スケールタイトルの表示と配置をグラフエンジンが自動的に決定します。

"font": "string"

色スケールタイトルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "7.5pt Sans-Serif" です。

"color": "string"

色名または数値指定の文字列で色スケールタイトルの色を定義します。デフォルト値は "black" です。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"tooltip": "string"

マウスポインタを色スケールタイトル上に置いたときに表示する文字列、または文字列を返す関数を指定します。

"wrap": number

長いタイトルの折り返し行数を定義します。整数を指定します。必要に応じて、長いタイトルをこの行数に折り返すか、"auto" を指定して、長いタイトルを必要な行数に折り返します。デフォルト値は 1 です。

"colors"

色の配列または開始値、終了値、色の配列を指定します。

["string","string", ..., "string"]

色名または数値指定の文字列で色の配列を定義します。デフォルト値は ["#253494", "#2C7FB8", "#41B6C4", "#A1DAB4"] です。

"start": value or "pin": value

色のグラデーションの開始値を定義する数値またはパーセント文字列です。

"pin" は "start" の同義語です。

"stop": value

色のグラデーションの終了値を定義する数値またはパーセント文字列です。

終了値はオプションです。指定しない場合、開始値の色スケールが終了色として定義されます。この色から次の開始値またはピン値の色に滑らかに変化します。

2 つの色エントリの開始値と終了値が一致している場合は、連続モードであっても不連続の単色で描画されます。たとえば、次のプロパティでは、色をグラデーションで表示せず、0 (ゼロ) から 50 までの値では単色の赤、50 から 100 の値では単色の緑を指定します。

```
[{"start": 0, "stop": 50, "color": "red"},
{"start": 50, "stop": 100, "color": "green"} ]
```

```
"color": "string"
```

指定した間隔の色を定義します。

例　ヒートマップの色スケールの定義

次のリクエストは、ヒートマップを生成し、黄緑色、水色、青緑色、緑色で構成された色スケールを定義します。

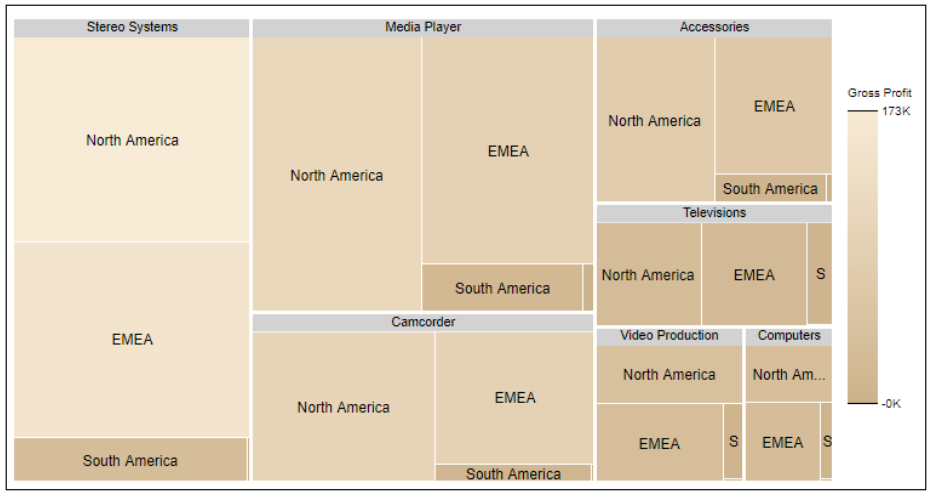
```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US AS 'Revenue'
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH HEATMAP
ON GRAPH SET STYLE *
type=data, column=n1, bucket=y-axis,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=color,$
*GRAPH_JS
"colorScale": {"colors": ["limegreen", "cyan", "teal", "green"]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

次のリクエストは、ツリーマップを生成し、黄褐色とアンティークホワイトで構成された色スケールを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM GROSS_PROFIT_US COGS_US
BY PRODUCT_CATEGORY
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
type=report, column=COGS_US, bucket=size, $
type=report, column=GROSS_PROFIT_US, bucket=color, $
type=report, column=PRODUCT_CATEGORY, bucket=detail, $
type=report, column=BUSINESS_REGION, bucket=detail, $
*GRAPH_JS
"legend": {"visible": true},
"colorScale": {"colors": ["tan", "antiquewhite"]}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

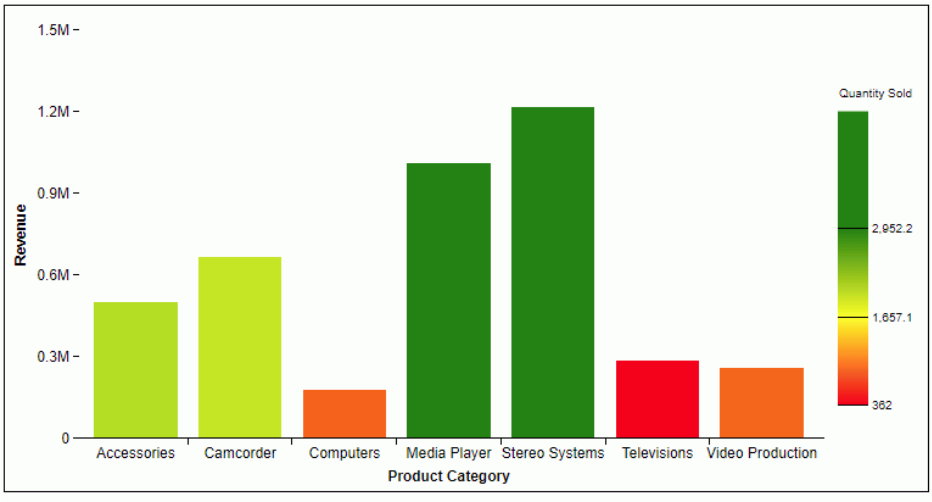


例　棒グラフの色スケールの定義

次のリクエストは、連続モード (デフォルト設定) で色スケールを定義します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US  QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis,$
TYPE=DATA, COLUMN=N2, BUCKET=y-axis,$
TYPE=DATA, COLUMN=N3, BUCKET=color,$
*GRAPH_JS_FINAL
"yaxis": {"majorGrid": {"visible": false}},
"xaxis": {"majorGrid": {"visible": false}},
"colorScale": {
    "colors": [
         {"pin": 0, "color": "red"},
         {"start": "30%", "color": "yellow"},
         {"pin": "60%", "color": "green"}
     ] }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

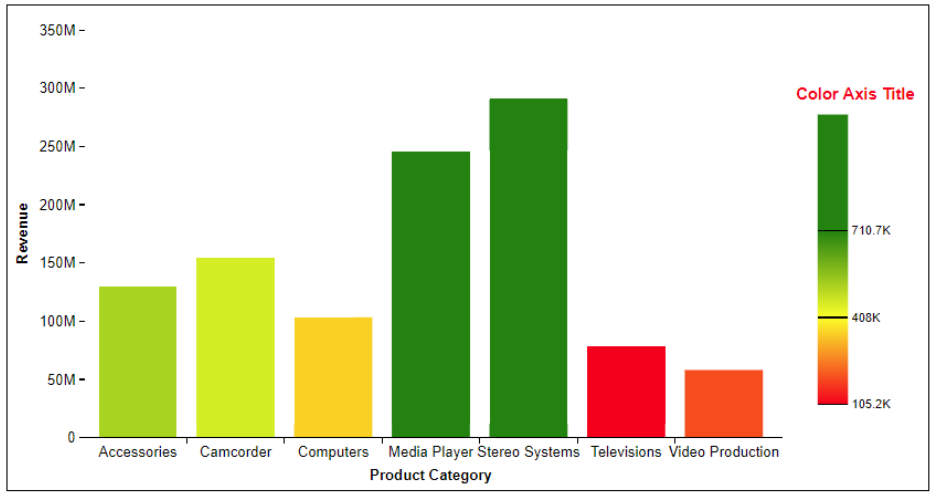


例　色スケールタイトルのフォーマット設定

次のリクエストは、色スケールタイトルを生成し、"Color Axis Title" というテキストを赤字で表示します。フォントのプロパティには "bold 10pt Arial" を使用します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US  QUANTITY_SOLD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N1, BUCKET=x-axis,$
TYPE=DATA, COLUMN=N2, BUCKET=y-axis,$
TYPE=DATA, COLUMN=N3, BUCKET=color,$
*GRAPH_JS_FINAL
"yaxis": {"majorGrid": {"visible": false}},
"xaxis": {"majorGrid": {"visible": false}},
"colorScale": {
    "title": {"text": "Color Axis Title", "font": "bold 10pt Arial",
"color": "red"},
    "colors": [
         {"pin": 0, "color": "red"},
         {"start": "30%", "color": "yellow"},
         {"pin": "60%", "color": "green"}
     ] }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



色スケールの色モードの定義

3 つの色モードを使用して、グラフおよび凡例の視覚的な色分け方法を定義することができます。

- デフォルトの色分け方法は、連続モードです。連続モードでは、色スケールの色は混合グラデーションで表示されます。
- 不連続モードでは、色スケールの色は不連続な色バンドで表示されます。それぞれの色の開始値および終了値を、色バンドの配列として定義します。色バンドは、colorScale オブジェクト内で定義する必要があります。
- ビンモードでは、colorScale 配列で指定されたエントリごとに、凡例に不連続なマーカーが作成されます。このモードは、色バンドで生成される凡例が不明瞭で判読困難な場合に役立ちます。

構文　　不連続な色スケールの定義

不連続モードの構文は次のとおりです。

```
"colorScale": {
   "colorMode":"discrete",
   "invert": boolean,
   "colors":[
                {
                 "start": value,
                 "stop": value,
                 "color": "string"
                 },
                 .
                 .
                 .
               ]
         }
```

説明

`"invert": boolean`

色バンドの順序を定義します。有効な値には、次のものがあります。

- true - 色バンドを高い値から低い値の順序で配列します。
- false - 色バンドを低い値から高い値の順序で配列します。これがデフォルト値です。

`"colors":`

各色バンドの開始値、終了値、色の配列です。

`"start": value`

色バンドの開始値を定義する数値またはパーセント文字列です。

`"stop": value`

色バンドの終了値を定義する数値またはパーセント文字列です。

`"color": "string"`

バンドの色を定義します。

連続モードの構文は次のとおりです。

```
"colorScale": {
  "colorMode": "continuous",
  "colors": ["string","string", ..., "string"]
        }
```

説明

`"colors": ["string","string", ..., "string"]`

混合する色を指定します。

注意：色モードを "continuous" に設定した場合でも、色バンドを定義すると、グラフは不連続モードで色分けされます。

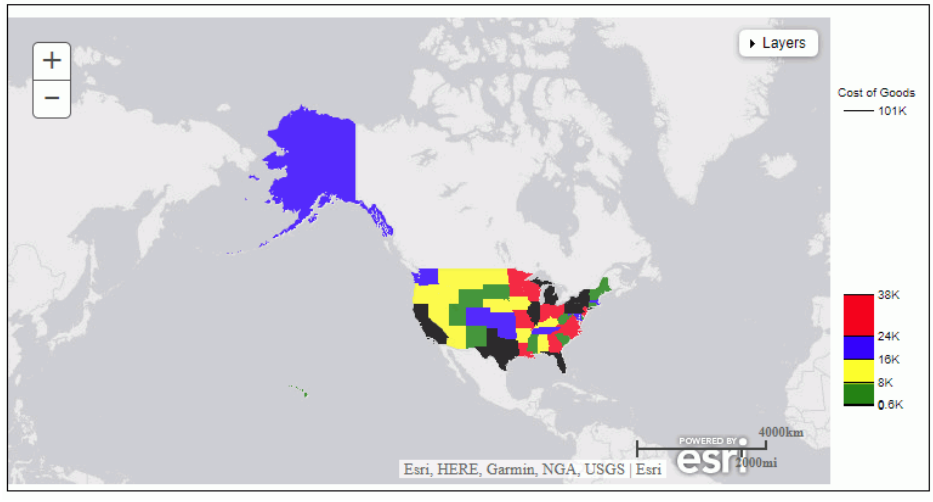
例　コロプレスマップでの不連続な色バンドの定義

次のリクエストは、コロプレスマップを生成し、不連続な色バンドを定義します。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN= COGS_US, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
"com.esri.map": {
"overlayLayers":
[
 {
 "ibiDataLayer": {
   "map-metadata": {
      "map_by_field": "STATE_PROV_NAME"
                    }
                 }
 }
],
"baseMapInfo":
 {
   "customBaseMaps":
   [
    {
     "ibiBaseLayer": "gray"
    }
   ]
   }
 }
},
"legend":{"visible":true},
"colorScale":
{
 "colorMode":"discrete",
  "colors": [
   {"start": 0, "stop": 8000,"color":"green"},
   {"start": 8000, "stop": 16000,"color":"yellow"},
   {"start": 16000, "stop": 24000,"color":"blue"},
   {"start": 24000, "stop": 38000,"color":"red"}
        ]
}
*END
ENDSTYLE
END
```

下図のように、グラフおよび凡例が色スケールで定義された不連続な色バンドで表示されます。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

構文　ビン色モードの定義

ビン色モードについては、次の構文を使用します。

```
"colorScale":
  {
    "colorMode": "bin",
    "invert": boolean,
      "binMarkers":
         {
           "visible": boolean,
           "concatSymbol": "string",
           "size": number,
           "shape": "string",
           "rotation": number,
           "position": "string",
           "border":
                {
                  "width": number,
                  "color": "string",
                  "dash": "string"
                }
         }
  }
```

説明

`"invert": boolean`
色バンドの順序を定義します。有効な値には、次のものがあります。

- true - 色バンドを高い値から低い値の順序で配列します。
- false - 色バンドを低い値から高い値の順序で配列します。これがデフォルト値です。

`"visible": boolean`
ビンマーカーを表示するかどうかを指定します。有効な値には、次のものがあります。

- true - ビンマーカーを表示します。これがデフォルト値です。
- false - ビンマーカーを表示しません。

`"concatSymbol": "string"`
各凡例ラベルでビンの最小値と最大値を連結する記号を文字列で定義します。デフォルトの連結記号は " ... " です。

`"size": number`
ビンマーカーの直径をピクセル単位で指定します。デフォルト値は 8 です。

`"shape": "string"`
標準のマーカー形状を指定する文字列です。デフォルト値は "square" です。

`"rotation": number`
マーカーの回転角度 (度数) を定義する数値です。デフォルト値は 0 (ゼロ) です。

`"position": "string"`
ラベルを基準にしたマーカーの相対位置を定義します。有効な値には、次のものがあります。

- "left" - マーカーをラベルの左側に配置します。これがデフォルト値です。
- "right" - マーカーをラベルの右側に配置します。
- "bottom" - マーカーをラベルの下側に配置します。
- "top" - マーカーをラベルの上側に配置します。

`"border":`
マーカー境界線のプロパティを定義します。

`"width": number`
境界の幅を定義する数値をピクセル単位で定義します。デフォルト値は 1 です。

`"color": "string",`
マーカー境界線の色を定義する文字列です。デフォルト値は "black" です。

"dash": "string"
マーカー境界線の破線スタイルを定義する文字列です。破線長さと破線間隔をピクセル単位で定義する数値を列記します。たとえば、「"dash": "1 1"」と指定すると、点線が生成されます。デフォルト値は「" "」です。この設定では、実線が生成されます。

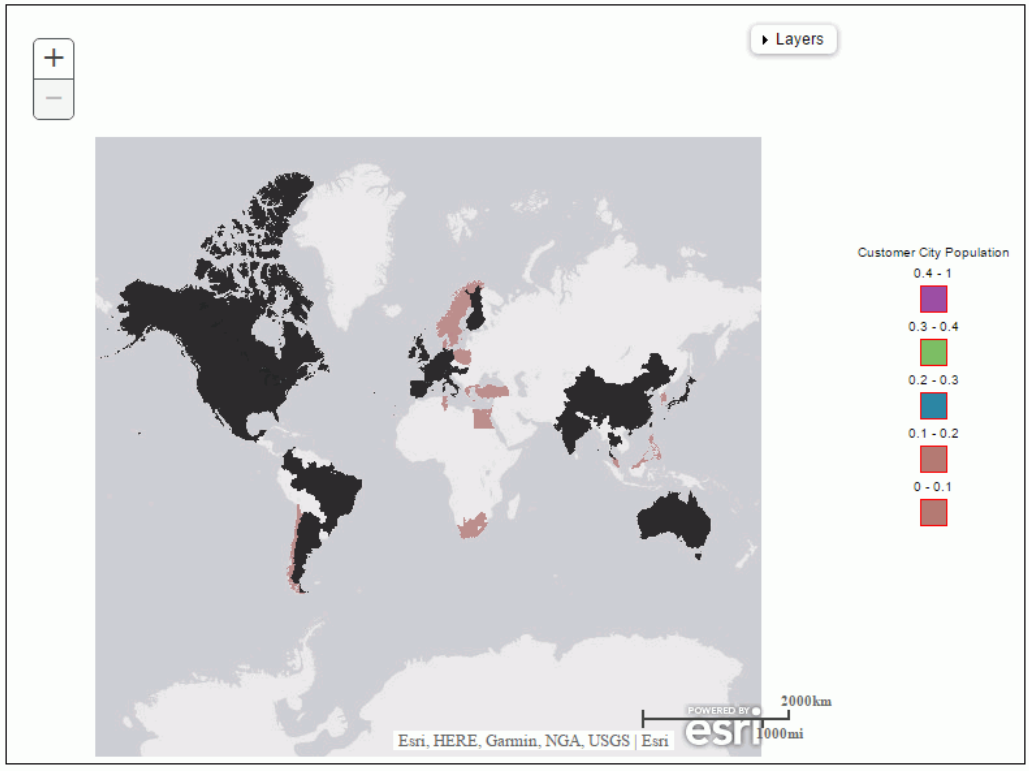
例 コロプレスマップでのビン色モードの使用

次のリクエストでは、ビン色モードを使用します。ビンマーカーは、20 ピクセルの正方形です。開始値と終了値を連結する記号は " - " (ハイフン) です。ビンマーカーは、ビンラベルの下側に配置され、1 ピクセル幅の赤色の境界線で囲まれます。色バンドは、逆の順序で配列され、最大のパーセントを表すバンドが最上位に表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM CITY_POPULATION
BY COUNTRY_NAME
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=CITY_POPULATION, BUCKET=color, $
```

```
*GRAPH_JS
"legend": {"visible": true},
"colorScale":
      {
      "colorMode": "bin",
      "invert": true,
      "binMarkers":
           {
              "visible": true,
              "concatSymbol": " - ",
              "size": 20,
              "shape": "square",
              "rotation": 0,
              "position": "bottom",
                 "border": {
                     "width": 1,
                     "color": "red",
                           }
           },
      "colors":[
            {"start":"0%",  "stop":"10%",  "color":"#B57877"},
            {"start":"10%", "stop":"20%",  "color":"#2887A6"},
            {"start":"20%", "stop":"30%",  "color":"#7ABD66"},
            {"start":"30%", "stop":"40%",  "color":"#984EA3"},
            {"start":"40%", "stop":"100%", "color":"#4DAF4A"},
                  ]
      },

"extensions": {
"com.esri.map": {
"overlayLayers":
   [
    {
     "title": "WebFOCUS Countries Request",
     "layerType": "choropleth",
     "geometrySourceType": "esri",
     "geometryLocateField": ["Country"],
     "geometryDataField": "name",
     "url": "http://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
    }
   ],
              }
            }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

ピクトグラムのサポート

グラフタイプに "pictogram" (ピクトグラム) を指定して、イメージをグラフのマーカーとして使用することができます。グラフに表示される各シリーズの棒には、棒の高さの最上端までイメージを反復的に積み上げる方法と、棒の高さに合わせて単一イメージを伸縮させる方法のいずれかが使用されます。各シリーズのイメージは、"marker": "shape" プロパティで、URL 参照または標準マーカー形状として指定します。

構文　　グラフマーカーとしてのピクトグラムの使用

```
*GRAPH_JS
"chartType": "pictogram",
"pictogramProperties": {
       "fillMode": "string",
       "backFill": "string",
       "lockToGrid": boolean
},
"series":
[
  {
      "series": number,
      "color": "string",
      "marker":
     {
            "shape": "string",
            "color": "string",
            "border": {"width": number, "color": "string"}
         }
   }
]
```

説明

`"pictogramProperties"`
ピクトグラムイメージのプロパティを定義します。

`"fillMode": "string"`
棒の塗りつぶし方法を制御します。有効な値には、次のものがあります。

- "repeatClip" - 棒の高さの最上端までイメージを反復的に積み上げます。これがデフォルト値です。
- "stretch" - 棒の高さに合わせてイメージを伸縮します。

`"backFill": "string"`
標準マーカー形状および「"fillMode": "repeatClip"」プロパティとともに使用する色指定の文字列を定義します。マーカー形状を軸領域の最上端まで (棒の高さを超えて) 反復的に積み上げますが、棒の高さを超えるマーカーをこの色で塗りつぶします。デフォルト値は "undefined" です。

`"lockToGrid": boolean`
true の場合、Y 軸罫線の間隔ごとにピクトグラムグリフ 1 つが正確に収まるようグラフが収縮されます。デフォルト値は false です。

"series"
イメージで表現されるシリーズのプロパティを定義します。

"series": number
ピクトグラムで表現されるシリーズの番号です。

"color": "string"
シリーズに適用する色指定の文字列です。シリーズ棒を表示せずにピクトグラムを表示するには、「"color": "transparent"」プロパティを使用します。

"marker"
シリーズに適用するマーカーのプロパティを定義します。

"shape": "string"
イメージファイルへの URL 参照または標準マーカー形状を指定します。URL 参照は、次のフォーマットで指定する必要があります。

"url(url_to_image_file)"

サポートされるイメージタイプは、.gif、.jpg、.png、.svg です。

"color": "string"
標準マーカー形状を使用する場合、マーカーの色を定義します。イメージファイルを使用する場合、このプロパティは無視されます。

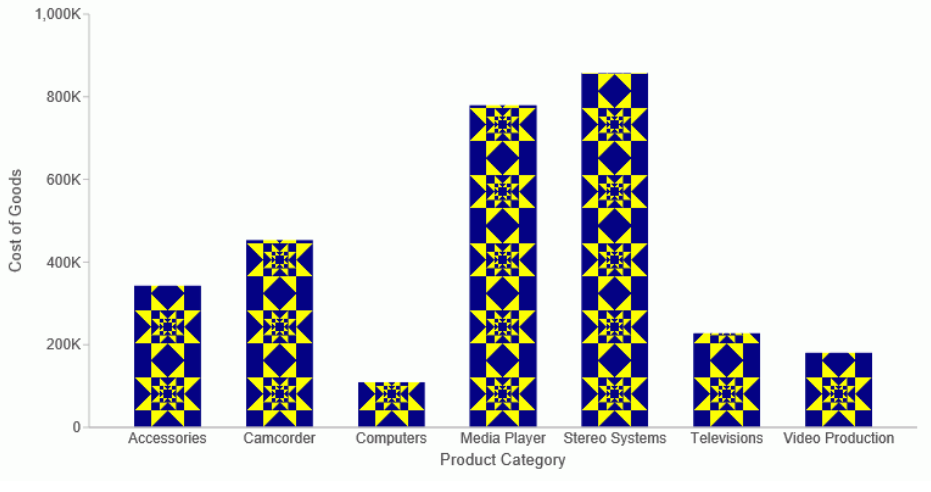
"border"
マーカー境界の幅 (ピクセル単位) および色指定の文字列を定義します。マーカー形状がイメージファイルの場合、境界プロパティは無視されます。

例　反復イメージファイルをピクトグラムとして使用

次のリクエストでは、キルトブロックイメージを反復的に使用してグラフの棒を生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=N1, bucket=x-axis, $
type=data, column=n2, bucket=y-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS
"chartType": "pictogram",
"pictogramProperties": {
    "fillMode": "repeatClip",
    "backFill": "undefined",
},
"series": [{
 "series": 0,
  "color": "transparent",
  "marker": {
  "shape": "url(http://ecl.informationbuilders.com/jschart/quilt.gif)",
  "color": "undefined",
  "border": {"width": 1, "color": "#DDDDDD"}
    }
}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

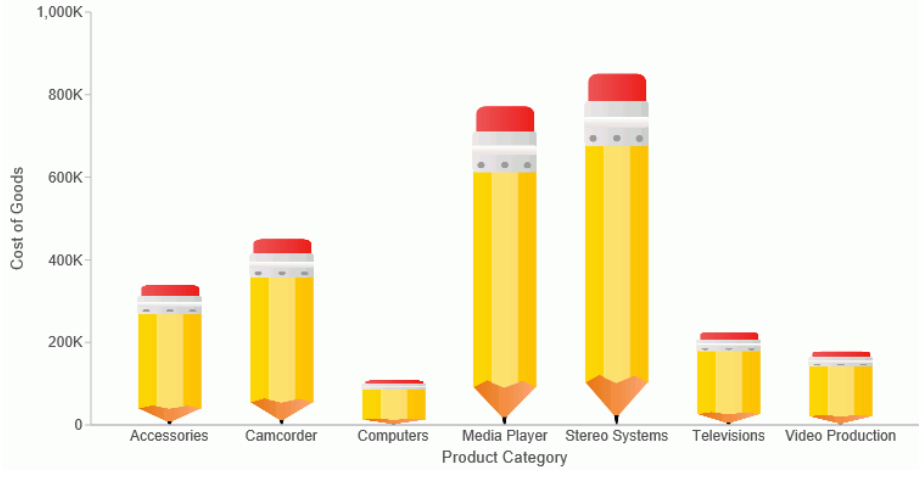


例　伸縮イメージファイルをピクトグラムとして使用

次のリクエストでは、イメージファイルを伸縮させてグラフの棒を生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=N1, bucket=x-axis, $
type=data, column=n2, bucket=y-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS_FINAL
"chartType": "pictogram",
"pictogramProperties": {
    "fillMode": "stretch",
    "backFill": "undefined",
},
"series": [{
 "series": 0,
    "color": "transparent",
    "marker": {
    "shape": "url(http://ecl.informationbuilders.com/jschart/pencil.gif)",
    }
}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



例　標準マーカー形状 (塗りつぶしあり) をピクトグラムとして使用

次のリクエストでは、5 つ星のマーカー形状および薄い灰色の背景塗りつぶしを使用してグラフを生成します。

```
GRAPH FILE MOVIES
SUM COPIES
BY RATING
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=N1, bucket=x-axis, $
type=data, column=n2, bucket=y-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
*GRAPH_JS
"chartType": "pictogram",
"pictogramProperties": {
    "fillMode": "repeatClip",
    "backFill": "lightgray",
},
"series": [{
    "series": 0,
    "color": "transparent",
    "marker": {
        "shape": "fivestar",
        "color": "gold",
        "border": {"width": 1, "color": "#FECF01"}
    }
}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

例　ピクトグラムでの角丸四角形マーカー形状の使用

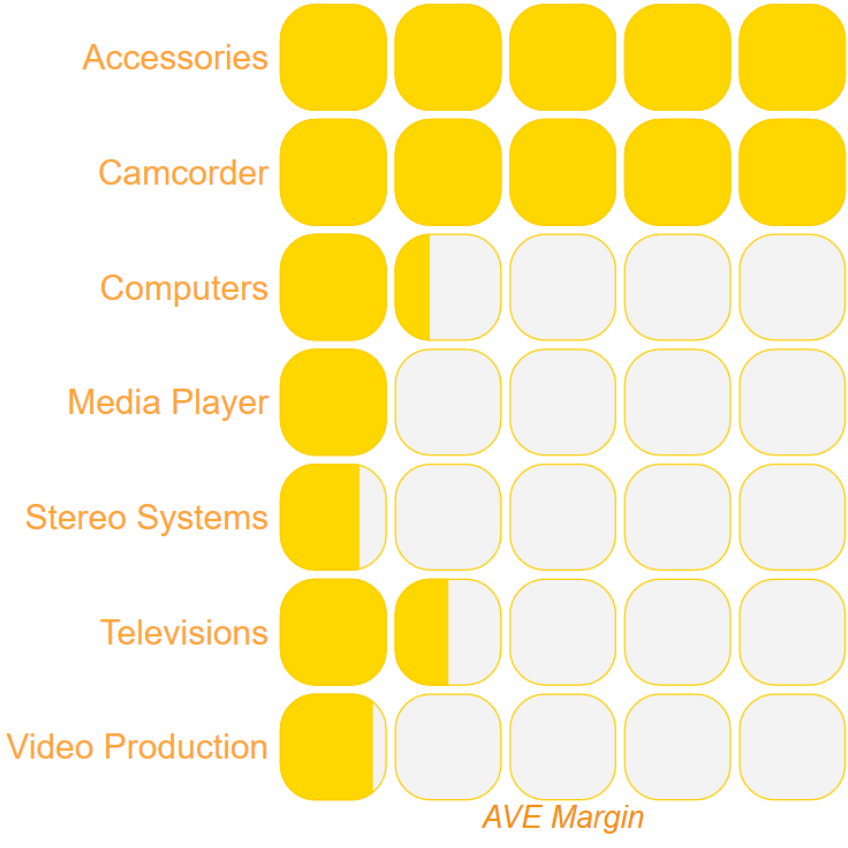
標準マーカー形状の "roundedSquare" は、角に丸みを帯びた四角形を描画し、ピクトグラムで使用することができます。

次のリクエストは、roundedSquare マーカー形状を適用したピクトグラムを作成します。

```
DEFINE FILE wf_retail_lite
 Margin/D12.2= IF PRODUCT_CATEGORY LE 'COMPUTERS' THEN (GROSS_PROFIT_US ) *
100
 ELSE IF PRODUCT_CATEGORY GT 'COMPUTERS' AND PRODUCT_CATEGORY LE 'STEREO'
THEN (GROSS_PROFIT_US/200) * 100
 ELSE (GROSS_PROFIT_US/300) * 100;
END

GRAPH FILE wf_retail_lite
SUM AVE.Margin
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET AUTOFIT ON
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $
```

```
*GRAPH_JS_FINAL
"chartType": "pictogram",
"pictogramProperties": {
    "fillMode": "repeatClip",
    "backFill": "#F3F3F3",
    "lockToGrid": true
},
"blaProperties":{"orientation": "horizontal", "barGroupGapWidth": 0},
"yaxis":{"max":250, "intervalMode": "interval", "intervalValue":50,
"labels":{"visible":false},
"baseLineStyle":{"width":0},
"bodyLineStyle":{"width":0},
"majorGrid":{"ticks":{"visible":false}},
"title":{"font": "16pt Sans-Serif italic","color": "#F28E02"},
},
"xaxis":{
"title":{"visible":false},
"baseLineStyle":{"width":0},
"bodyLineStyle":{"width":0},
"majorGrid":{"ticks":{"visible":false}},
"labels":{"font": "18pt Sans-Serif", "color": "#FDA52B"},
},

"series": [{
    "series": 0,
    "color": "transparent",
    "marker": {
        "shape": "roundedSquare",
        "color": "gold",
        "border": {"width": 1, "color": "#FECF01"}
    }
}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



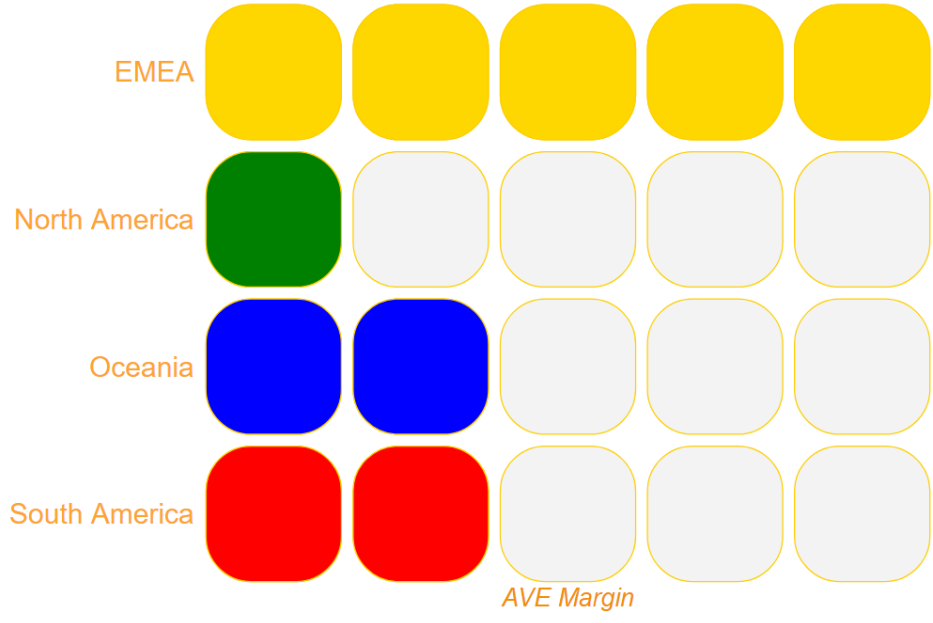
例　roundedSquare ピクトグラムマーカー形状によるグループの色分け

グループごとに異なる色を表示するには、シリーズ番号とともにグループ番号を指定した上で、シリーズとグループの組み合わせごとに適用する色を指定します。次のリクエストは、シリーズ 0 (ゼロ) の各グループに色を指定します。

```
DEFINE FILE wf_retail_lite
 Margin/D12.2= IF BUSINESS_REGION EQ 'EMEA' THEN 1000
 ELSE IF BUSINESS_REGION GT 'EMEA' AND BUSINESS_REGION LT 'Oceania' THEN 50
 ELSE 100;
END

GRAPH FILE wf_retail_lite
SUM AVE.Margin
BY BUSINESS_REGION
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET AUTOFIT ON
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/
ENWarm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, $

*GRAPH_JS_FINAL
"chartType": "pictogram",
"pictogramProperties": {
    "fillMode": "repeatClip",
    "backFill": "#F3F3F3",
    "lockToGrid": true
},
"blaProperties":{"orientation": "horizontal", "barGroupGapWidth": 0},
"yaxis":{"max":250, "intervalMode": "interval", "intervalValue":50,
"labels":{"visible":false},
"baseLineStyle":{"width":0},
"bodyLineStyle":{"width":0},
"majorGrid":{"ticks":{"visible":false}},
"title":{"font": "16pt Sans-Serif italic","color": "#F28E02"},
},
"xaxis":{
"title":{"visible":false},
"baseLineStyle":{"width":0},
"bodyLineStyle":{"width":0},
"majorGrid":{"ticks":{"visible":false}},
"labels":{"font": "18pt Sans-Serif", "color": "#FDA52B"},
},
```

```
"series": [{
    "series": 0, "group": 0,
    "color": "transparent",
    "marker": {
        "shape": "roundedSquare",
        "color": "gold",
        "border": {"width": 1, "color": "#FECF01"}
    }
},
{
    "series": 0, "group": 1,
    "color": "transparent",
    "marker": {
        "shape": "roundedSquare",
        "color": "green",
        "border": {"width": 1, "color": "#FECF01"}
    }
},
{
    "series": 0, "group": 2,
    "color": "transparent",
    "marker": {
        "shape": "roundedSquare",
        "color": "blue",
        "border": {"width": 1, "color": "#FECF01"}
    }
},
{
    "series": 0, "group": 3,
    "color": "transparent",
    "marker": {
        "shape": "roundedSquare",
        "color": "red",
        "border": {"width": 1, "color": "#FECF01"}
    }
}
]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



エラーバーの描画 (errorBars)

errorBars プロパティは、棒ライザ、折れ線ライザ、面ライザ、バブルマーカー、散布図マーカー上に描画されるエラーバーを定義します。エラーバーを描画することで、ライザが期待値を上回っているか、下回っているかを視覚的に表現することができます。

次のコードは、各プロパティおよびデフォルト値を示しています。

```
"errorBars": {
   "yData": undefined,
   "xData": undefined,
   "hatWidth": '50%',
   "line": {"color": "black","width": 1,"dash": ""},
   "marker": {
      "color": "grey",
      "shape": "diamond",
      "size": "undefined",
      "border": {"width": 1,"color": "black","dash": ""}
   }
},
```

構文　　エラーバーの描画

```
"errorBars": {
"yData": [highValue, markerPosition, lowValue, ...]
   "xData": [highValue, markerPosition, lowValue, ...],
   "hatWidth": value,
   "line": {"color": "string",
          "width": number,
          "dash": "string"},
   "marker": {
      "color": color,
      "shape": "string",
      "size": number,
      "border": {
               "width": number,
               "color": "string",
               "dash": "string"
}
   }
}
```

説明

`"yData": [highValue, markerPosition, lowValue, ...]` または

`"yData": [highValue, lowValue, ...]`

各ライザの最高値、マーカー位置 (オプション)、最低値の配列を定義します。Y 軸データにエラーバーを描画しない場合、この値は undefined にする必要があります。各配列には、次のいずれかのフォーマットを指定することができます。

- [highValue,lowValue] - ライザにエラーバーを描画しますが、マーカーは描画しません。
- [highValue, markerPosition, lowValue] - ライザにエラーバーとマーカーの両方を描画します。

`"xData": [highValue, markerPosition, lowValue, ...]` または

`"xData": [highValue, lowValue, ...]`

各ライザの最高値、マーカー位置 (オプション)、最低値の配列を定義します。X 軸データにエラーバーを描画しない場合、この値は undefined にする必要があります。各配列には、次のいずれかのフォーマットを指定することができます。

- [highValue,lowValue] - ライザにエラーバーを描画しますが、マーカーは描画しません。
- [highValue, markerPosition, lowValue] - ライザにエラーバーとマーカーの両方を描画します。

"hatWidth": value

エラーバーの幅をピクセル単位の数値で指定するか、ライザ幅に対する割合を示すパーセント文字列で指定します。文字列は、パーセント記号 (%) を含めて二重引用符 (") で囲みます。

"line":

線のプロパティを定義します。

"color": "string"

色名または数値指定の文字列で線およびハットの色を定義します。

色の定義についての詳細は、63 ページの 「 色およびグラデーション 」 を参照してください。

"width": number

エラーバーの線幅を定義する数値をピクセル単位で指定します。デフォルト値は 1 です。

"dash": "string"

線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。実線の場合、「""」を使用します。

"marker":

シリーズに適用するマーカーのプロパティを定義します。

"color": color

シリーズのマーカーの色を色名、数値指定の文字列、グラデーション文字列のいずれかで定義します。文字列は二重引用符 (") で囲みます。また、JSON オブジェクトでグラデーションを定義することもできます。デフォルト値は "grey" です。

"shape": "string"

シリーズのマーカー形状を定義する文字列です。マーカー形状のリストについては、427 ページの 「 シリーズ固有のプロパティ 」 を参照してください。デフォルト値は "diamond" です。bar、circleMinus、circlePlus、cross、tick マーカーを使用する場合は、境界幅と色を指定する必要があります。

"size": number

マーカーの直径を定義する数値を指定します。この値を undefined にすると、グラフエンジンがマーカーサイズを選択します。デフォルト値は undefined です。

"border":
マーカー境界線のプロパティを定義します。

"width": number

境界の幅を定義する数値をピクセル単位で定義します。デフォルト値は 1 です。

"color": "string"

マーカー境界色を色名または数値指定の文字列で定義します。デフォルト値は "black" です。

"dash": "string"

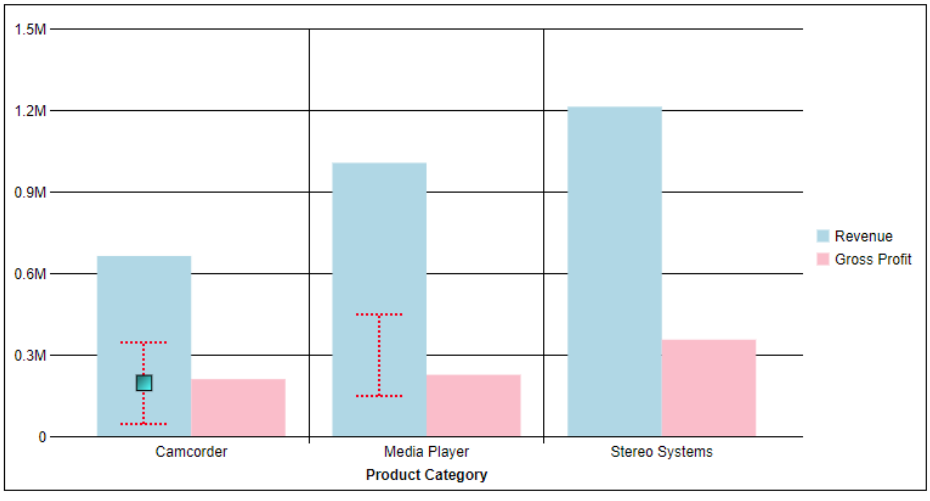
マーカー境界線の破線スタイルを定義する文字列です。デフォルト値は「""」です (実線)。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。

例　エラーバーの生成

次のリクエストは、最初の 2 本のライザにエラーバーを生成し、赤色の破線で表示します。最初のライザのエラーバーには、色のグラデーションが適用された正方形のマーカーを表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Camcorder' OR 'Media Player' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS_FINAL
"errorBars": {
   "hatWidth": "50%",
   "yData": [[[50000, 200000, 350000], [150000, 450000]]],
   "line": {"color": "red", "width": 2, "dash": "2 2"},
   "marker": {
      "color": "linear-gradient(0,0,100%,100%, 20% teal, 95% cyan)",
      "shape": "square", "size": 25}},
"series": [
   {"series": 0, "color": "lightblue"},
   {"series": 1, "color": "pink"}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



HTML ツールヒントの生成 (htmlToolTip)

1 つまたはすべてのシリーズでツールヒントが定義されている場合、htmlToolTip プロパティを使用して、グラフのすべてのツールヒントに HTML ベース (div) スタイルのツールヒントを有効または無効にすることができます。ツールヒントに表示するテキストは、シリーズ固有の tooltip プロパティで定義します。

次のコードは、各プロパティおよびデフォルト値を示しています。

```
"htmlToolTip": {
    "enabled": true,
    "mouseMargin": 10,
    "style": "undefined",
    "autoTitleFont": "bold 12pt Sans-Serif",
    "autoContentFont": "10pt Sans-Serif",
    "snap": false,
 "sticky": "auto",
 "fill": 'linear-gradient(to bottom, rgba(250,250,250,0.98)
8%,rgba(230,230,230,0.98) 95%)',
 "border": {
                "width": 1,
             "color": 'rgba(150,150,150,0.95)',
                 "cornerRadius": 0},
 "cascadeMenuStyle": {
                "label": {
                         "font": undefined,
                         "color": undefined
                       },
                "nameValue": {
                         "name": {
                                  "font": "9pt sans-serif",
                                  "color": "#505050"
                                },
                         "value": {
                                  "font": "bold 9pt sans-serif",
                                  "color": "black"
                                 }},
      "hover": {
                         "fill": "rgba(220,220,220,0.9)",
                         "labelColor": "black"
                       }
                  }
                 }
```

構文　　HTML ツールヒントの生成

```
"htmlToolTip": {
            "enabled": boolean,
   "mouseMargin": number,
   "style": "string",
   "autoTitleFont": "string",
   "autoContentFont": "string",
   "snap": boolean,
   "sticky": boolean ,
   "fill": "string",
   "border": {
               "width": number,
             "color": "string",
               "cornerRadius": {"x": value, "y": value}
                           },
   "cascadeMenuStyle": {
               "label": {
                        "font": "string",
                        "color": "string"
                      },
               "nameValue": {
                        "name": {
                                 "font": "string",
                                 "color": "string"
                               },
                        "value": {
                                 "font": "string",
                                 "color": "string"
                                }},
      "hover": {
                        "fill": "string",
                        "labelColor": "string"
                      }


           }
             }
```

説明

`"enabled": boolean`

HTML ベースのツールヒントを有効または無効にします。有効な値には、次のものがあります。

- true - グラフのすべてのツールヒントに HTML ベースのツールヒントを使用します。これがデフォルト値です。
- false - グラフのすべてのツールヒントに標準スタイルのツールヒントを使用します。

"mouseMargin": number

マウスポインタの上端からツールヒントの下端までの距離をピクセル単位で指定します。

"style": "string"

ツールヒントのスタイルを定義します。有効な値には、次のものがあります。

- "seriesFill" - ツールヒントの背景をシリーズ色より薄い色で塗りつぶします。
- CSS プロパティを定義するオブジェクトまたは文字列
- undefined - これがデフォルト値です。

"autoTitleFont": "string"

"series": "tooltip" が "auto" に設定されている場合、CSS フォント文字列を使用して、自動ツールヒントのタイトルテキストのフォーマットを定義します。デフォルト値は "bold 12pt Sans-Serif" です。

"autoContentFont": "string"

"series": "tooltip" が "auto" に設定されている場合、CSS フォント文字列を使用して、自動ツールヒントのコンテンツテキストのフォーマットを定義します。デフォルト値は "10pt Sans-Serif" です。

"snap": boolean

吹き出し型ツールヒントを有効または無効にします。有効な値には、次のものがあります。

- true - 吹き出し型ツールヒントを有効にします。グラフの各ライザ横のハードコードされたスポットにツールヒントを描画します。
- false - 吹き出し型ツールヒントを無効にします。ツールヒントはカーソルから数ピクセル上に描画されます。これがデフォルト値です。

"sticky": boolean

付箋型ツールヒントを有効または無効にします。有効な値には、次のものがあります。

- true - 付箋型ツールヒントを有効にします。マウスポインタをグラフフレームの外側へ移動した場合にのみツールヒントが非表示になります。
- false - 付箋型ツールヒントを無効にします。マウスポインタをライザの外側へ移動すると、ツールヒントが非表示になります。これがデフォルト値です。

`"fill": "string"`

ツールヒントコンテナの塗りつぶし色またはグラデーションを定義する文字列、または undefied を指定します。undefined を指定した場合、グラフエンジンでコーディングされたデフォルトの塗りつぶし色が適用されます。デフォルト値は "linear-gradient(to bottom, rgba(250,250,250,0.98) 8%,rgba(230,230,230,0.98) 95%)" です。

`"border"`

ツールヒントコンテナの境界線のプロパティを定義します。

`"width": number`

境界線の幅をピクセル単位で指定するか、undefined を指定します。undefined を指定した場合、グラフエンジンでコーディングされたデフォルト値が適用されます。デフォルト値は 1 です。

`"color": "string"`

境界線の色を定義する色指定文字列、または undefined を指定します。undefined を指定した場合、グラフエンジンでコーディングされたデフォルト値が適用されます。デフォルト色は "rgba(150,150,150,0.95)" です。

`"cornerRadius": {"x": value, "y": value}`

ツールヒントコンテナの角形状のプロパティを定義します。

cornerRadius プロパティには、1 つまたは 2 つの値を指定することができます。単一値の 0 (ゼロ) を指定すると、角形状は直角になります。これがデフォルト角形状です。曲線の角形状は、円形または楕円形にすることができます。x および y 値は、円形の半径、または楕円形の長半径と短半径を表します。ここで x および y 値はグラフの向きに対応します。x は序数軸を表し、y は数値軸を表します。デフォルト値は 0 (ゼロ) です。

x および y の有効値は次のとおりです。

- 数値 - 角形状の半径をピクセル単位で指定します。
- パーセント文字列 (例、"20%") - ライザの序数軸または数値軸のパーセントを表します。
- CSS 長さ文字列 (例、"5em"、"10px") - 角形状の半径をピクセル単位で指定します。CSS 長さ文字列についての詳細は、https://developer.mozilla.org/en-US/docs/Web/CSS/length を参照してください。

- "x" および "y" プロパティで定義されるオブジェクト (ここで "x" および "y" には上記のいずれかを指定可能) - ライザの序数軸 (x) および数値軸 (y) の角形状の半径を指定します。

"cascadeMenuStyle"

ツールヒントメニューのプロパティを定義します。

"label"

ツールヒントのラベルテキストのプロパティを定義します。

"font": "string"

ラベルテキストのフォントを定義する文字列です。デフォルト値は undefined です。

"color": "string"

ラベルテキストの色を定義する文字列です。デフォルト値は undefined です。

"nameValue":

ツールヒントのフィールド名とフィールド値のプロパティを定義します。

"name"

ツールヒントのフィールド名のプロパティを定義します。

"font": "string"

フィールド名のフォントを定義する文字列です。デフォルト値は "9pt sans-serif" です。

"color": "string"

フィールド名の色を定義する文字列です。デフォルト値は "#505050" です。

"value"

ツールヒントのフィールド値のプロパティを定義します。

"font": "string"

フィールド値のフォントを定義する文字列です。デフォルト値は "bold 9pt sans-serif" です。

"color": "string"

フィールド値の色を定義する文字列です。デフォルト値は "black" です。

"hover"

ツールヒントの上にマウスポインタを置いたときのプロパティを定義します。

"fill": "string"

マウスポインタを置いたときの塗りつぶし色を定義します。デフォルト色は "rgba(220,220,220,0.9)" です。

"labelColor": "string"

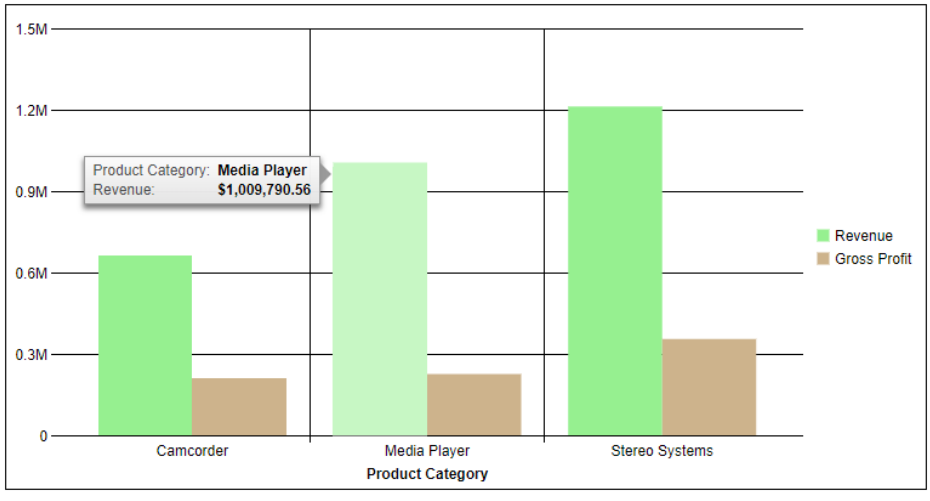
マウスポインタを置いたときのサブメニューのラベル色を定義します。デフォルト値は "black" です。

例　HTML ベースのツールヒントの生成

次のリクエストは、HTML ベースのツールヒントを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Camcorder' OR 'Media Player' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "lightgreen"},
   {"series": 1, "color": "tan"}],
"htmlToolTip": {"enabled": true, "snap": true}
*END
ENDSTYLE
END
```

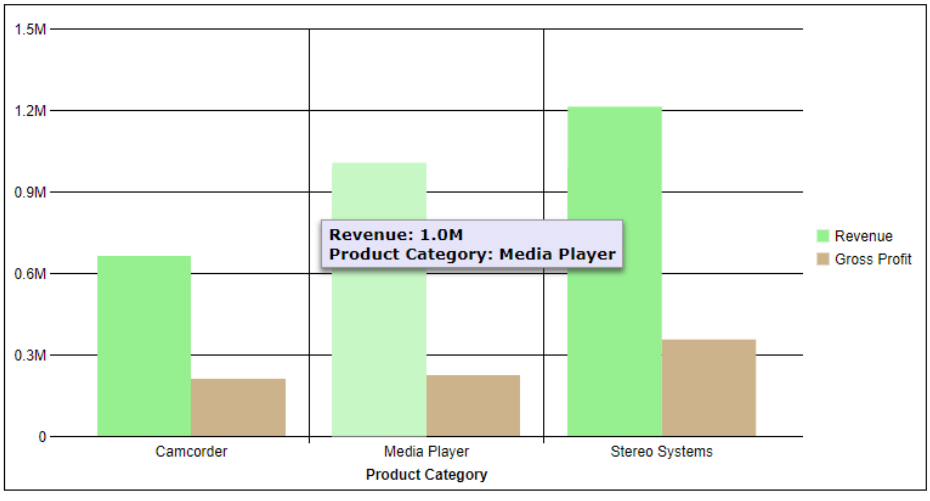
出力結果では、snap プロパティが true に設定されているため、吹き出し型ツールヒントが表示されます。



次のリクエストでは、ツールヒントの背景色を設定し、"tooltip": "auto" が設定されたシリーズ 0 のツールヒントのフォントを指定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
WHERE PRODUCT_CATEGORY EQ 'Camcorder' OR 'Media Player' OR 'Stereo Systems'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS_FINAL
"series": [
   {"series": 0, "color": "lightgreen", "tooltip": "auto"},
   {"series": 1, "color": "tan"},
   ],
"htmlToolTip": {
   "enabled": true,
   "style": {"background": "lavender"},
   "autoContentFont": "bold 10pt Verdana"}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



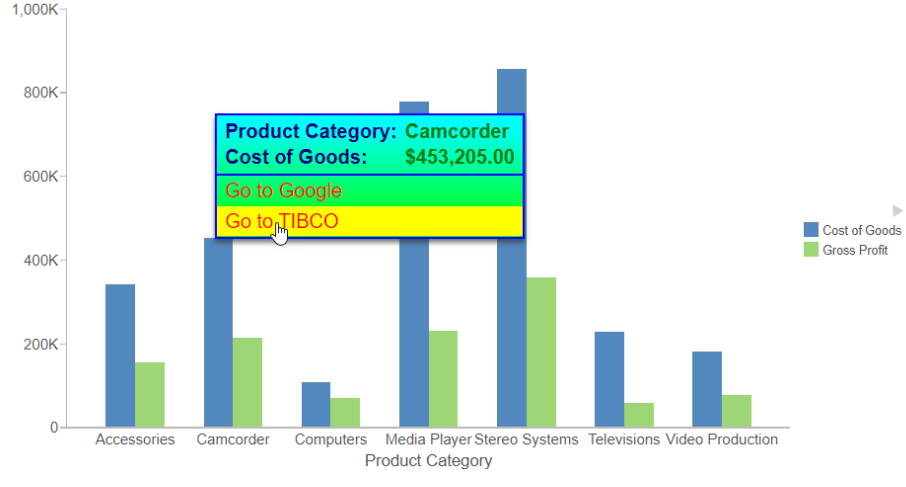
例 HTML ツールヒントのカスケードメニューのフォーマット設定

次のリクエストでは、シリーズ 0 にはスタイルシートで定義されたドリルダウン付きデフォルトツールヒントを使用し、シリーズ 1 にはサブメニュー付きカスタムツールヒントを定義します。フィールド名は紺色、フィールド値は緑色です。ラベルは赤色です。マウスポインタを置いたときの塗りつぶし色は黄色、サブメニューのラベル色は茶色です。

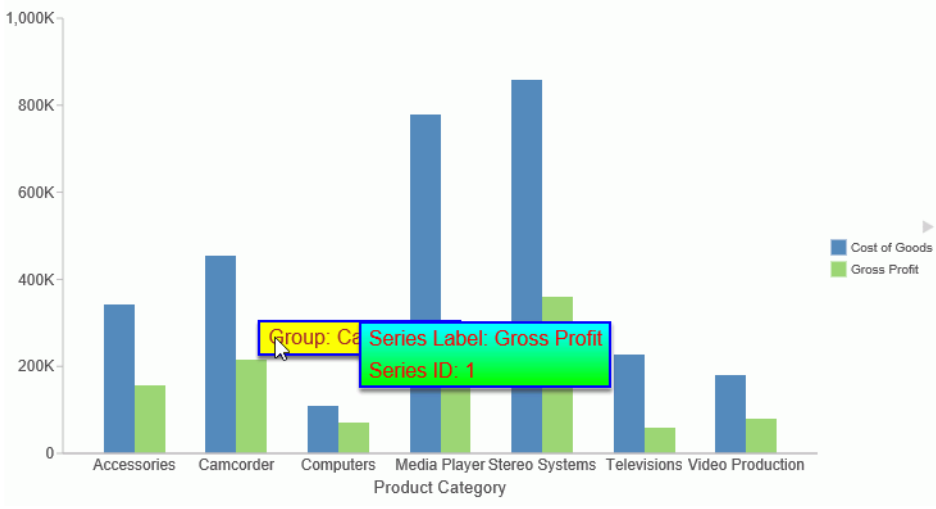
```
GRAPH FILE wf_retail_lite
SUM COGS_US
GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=x-axis, $
TYPE=DATA, COLUMN=N2, BUCKET=y-axis, DRILLMENUITEM='Go to Google',
TARGET='_blank', URL=http://www.google.com,
DRILLMENUITEM='Go to TIBCO', TARGET='_blank',
   URL=http://www.tibco.com, $
TYPE=DATA, COLUMN=N3, BUCKET=y-axis, $
*GRAPH_JS_FINAL
"series": [{
    "series": "reset"},
    {"series": 1, "tooltip": [{
        "entry": "Group: {{group_label}}",
        "children": ["Series Label: {{series_label}}",
               "Series ID: {{series_id}} "]
                          }]
            }],
"htmlToolTip": {
    "enabled": true,
    "fill": "linear-gradient(to bottom, cyan 8%, lime 95%)",
    "border": {
                "width": 2,
                "color": "blue"
                      },
```

```
    "cascadeMenuStyle": {
          "label": {
          "font": "12pt Arial",
          "color": "red"
                  },
          "nameValue": {
                  "name": {
                  "font": "bold 12pt Arial",
                  "color": "navy"
                    },
                 "value": {
                  "font": "bold 12pt sans-serif",
                  "color": "green"
                     }
                       },
            "hover": {
                  "fill": "yellow",
                  "labelColor": "brown"
                    }
                        }
                   }
*END
ENDSTYLE
END
```

下図は、シリーズ 0 の最初のドリルダウンにマウスポインタを置いたときのツールヒントを示しています。



下図は、シリーズ 1 のサブメニューにマウスポインタを置いたときのツールヒントを示しています。



グラフでのユーザ操作の定義 (interaction)

interaction プロパティは、グラフでのユーザ操作を定義します。

- 棒グラフ、折れ線グラフ、面グラフ、バブルグラフ、散布図の場合、「"mousedrag": "pan"」を指定すると、ユーザがグラフフレーム内でマウスをドラッグした際に、ライザ、マーカー、軸ラベル、データラベル (表示されている場合) が移動します。
- 折れ線グラフ、面グラフ、散布図の場合、「"mousemove": "nearestNeighbor"」を指定すると、ユーザがマウスポインタをデータポイントから離した場合でも、ツールヒントが表示された状態になります。
- 3D グラフの場合、「"mousedrag": "rotate"」を指定すると、ユーザがグラフフレームをドラッグした際にグラフフレームが回転します。
- 円グラフで [その他] 円項目が定義されている場合 ("pieProperties": "otherSlice" を使用)、「"click": "otherSliceDrillDown"」を指定すると、ユーザが [その他] 円項目をクリックした際に、その他の円項目コンポーネントにドリルダウンします。この場合、ドリルダウン先の円グラフに、[その他] 円項目を構成する個々の円項目が表示されます。

構文　　グラフでのユーザ操作の定義

```
"interaction": {
   "click": "string",
   "mousedrag": "string",
   "dblclick": "string",
   "mousemove": "string"
},
```

説明

`"click": "string"`
マウスのクリックによるユーザ操作を定義する文字列です。有効な値には、次のものがあります。

- "otherSliceDrillDown" - [その他] 円項目を構成する個々の円項目を表示します。
- undefined - これがデフォルト値です。

`"mousedrag": "string"`
マウスのドラッグによるユーザ操作を定義する文字列です。

- "select" - グラフのライザとマーカーを選択します。
- "pan" - マウスのドラッグでデータセット内を移動し、他の値を表示します。
- "rotate" - 3D キューブを回転します (3D グラフの場合のみ)。
- undefined - すべての動作を無効にします。これがデフォルト値です。

`"dblclick": "string"`
マウスのダブルクリックによるユーザ操作を定義する文字列です。有効な値には、次のものがあります。

- "resetView" - "pan" および "rotate" の動作と連動して機能します。ダブルクリックで、グラフが元の表示に戻ります。
- undefined - これがデフォルト値です。

`"mousemove": "string"`
折れ線グラフ、散布図、面グラフでのツールヒントの検知方法を指定します。有効な値には、次のものがあります。

- "over" - マウスポインタをデータポインタの真上に置いた場合にのみツールヒントを表示します。これがデフォルト値です。

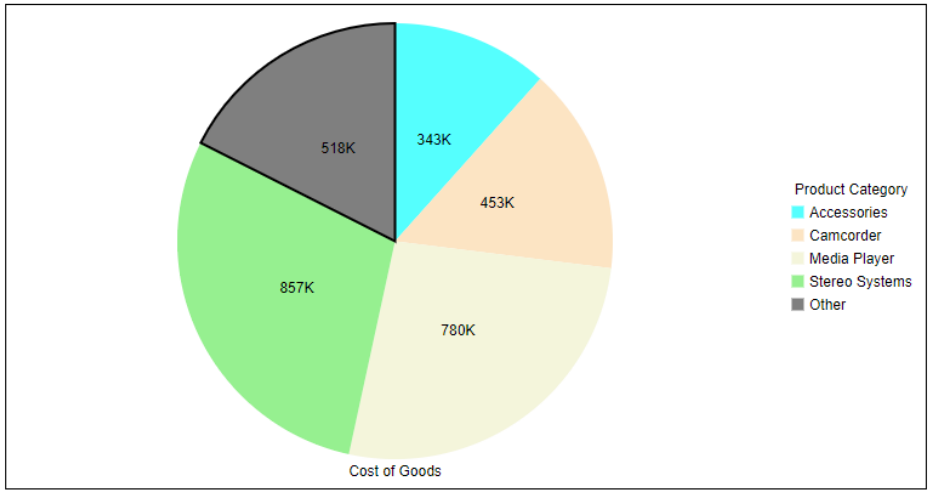
- "nearestNeighbor" - 折れ線グラフ、散布図、面グラフでマウスポインタをデータポイント上から離した場合でも、近傍データポイントのツールヒントを表示します。

例　円グラフでのユーザ操作の有効化

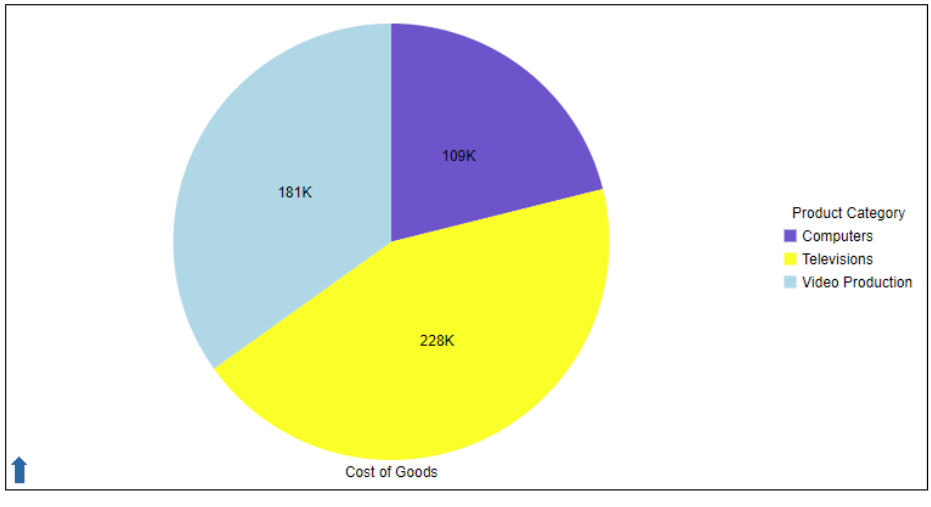
次のリクエストは、円グラフを生成し、[その他] 円項目 (otherSlice) のしきい値を 8% に設定します。interaction 設定により、ユーザが [その他] 円項目をクリックした際に、[その他] 円項目を構成する個々の円項目がグラフに表示されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=measure,$
*GRAPH_JS
"dataLabels": {"visible": true},
"pieProperties": {"otherSlice": {"threshold": "8%"}},
"interaction": {"click": "otherSliceDrillDown"},
"series": [
    {"series": 0, "color": "cyan", "showDataValues": true},
    {"series": 1, "color": "bisque", "showDataValues": true},
    {"series": 2, "color": "slateblue", "showDataValues": true},
    {"series": 3, "color": "beige", "showDataValues": true},
    {"series": 4, "color": "lightgreen", "showDataValues": true},
    {"series": 5, "color": "yellow", "showDataValues": true},
    {"series": 6, "color": "lightblue", "showDataValues": true},
    {"series": 7, "color": "lavender", "showDataValues": true},
    {"series": 8, "color": "limegreen", "showDataValues": true},
    {"series": 9, "color": "red", "showDataValues": true}]
*END
ENDSTYLE
END
```

出力結果では、パーセント値がしきい値に満たない円項目がすべて統合され、1 つの円項目として灰色で表示されています。



[その他] 円項目をクリックすると、下図のように [その他] 円項目のドリルダウンが生成され、[その他] 円項目を構成する各コンポーネントおよびそのパーセント値が表示されます。左下の青色の矢印をクリックすると、元の表示に戻ります。

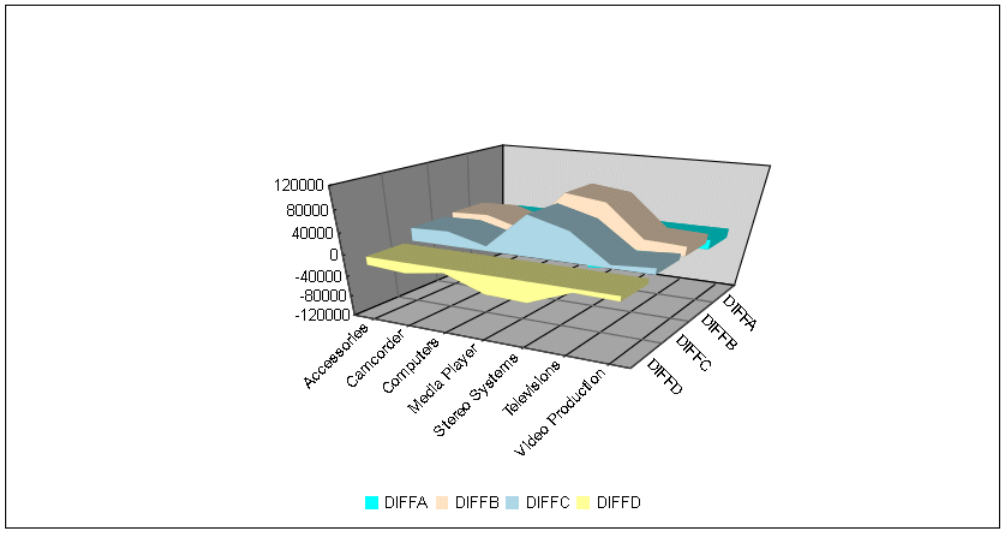


例　3D グラフの回転の有効化

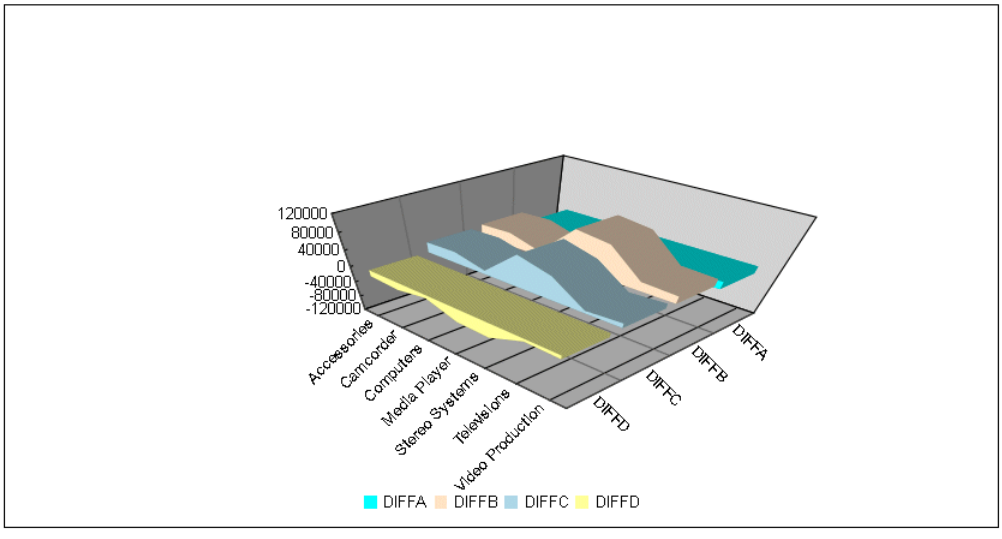
次のリクエストは、3D 面グラフを生成します。interaction プロパティを使用すると、マウスのドラッグでグラフを回転し、ダブルクリックで元に戻す操作を有効にすることができます。

```
DEFINE FILE WF_RETAIL_LITE
DIFFA = GROSS_PROFIT_US - REVENUE_US;
DIFFB = REVENUE_US - GROSS_PROFIT_US;
DIFFC = COGS_US - (COGS_US * DISCOUNT_US)/100;
DIFFD = COGS_US - MSRP_US;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFFA DIFFB DIFFC DIFFD
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH 3DAREAS
ON GRAPH SET STYLE *
*GRAPH_JS
  interaction:
      {mousedrag: 'rotate',
      dblclick: 'resetView'},
series: [
    {series: 0, color: 'cyan'},
    {series: 1, color: 'bisque'},
    {series: 2, color: 'lightblue'},
]
*END
ENDSTYLE
END
```

下図は、元のグラフ表示を示しています。



グラフの外側をドラッグして回転すると、下図のようになります。



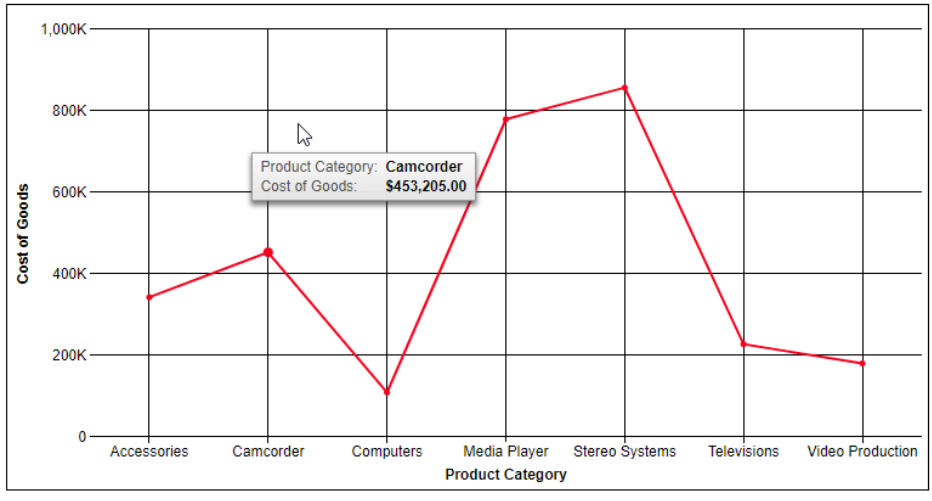
グラフの外側をダブルクリックすると、元のグラフ表示に戻ります。

例　折れ線グラフでのツールヒントの近傍データポイント検知の有効化

次のリクエストは、折れ線グラフを生成し、ツールヒントの近傍データポイント検知を有効にします。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY  PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
*GRAPH_JS
"interaction": {"mousemove": "nearestNeighbor"}
*END
END
```

下図のように、マウスポインタをグラフのデータポイント上から離した場合でもツールヒントが表示されます。



マウスオーバーインジケータのフォーマット設定 (mouseOverIndicator)

mouseOverIndicator プロパティは、マウスポインタをライザ上に置いたときにライザをハイライト表示する色を定義します。折れ線グラフ、複合グラフ、パレートグラフの場合、マウスポインタを折れ線ライザ上に置いたときに描画するマーカーを定義するには、marker プロパティを使用します。

- マーカーを表示しない折れ線グラフの場合 ("series": #, "marker": {"visible": false})、マウスポインタに最も近い線のポイントの位置にマーカーが 1 つ追加されます。このマーカーは、シリーズ固有のマーカーではなく、"mouseOverIndicator": "marker" で定義されたプロパティに基づいて作成されます。
- マーカーを表示する折れ線グラフ ("series": #, "marker": {"visible": true}) および散布図の場合、マウスポインタに最も近いマーカー (シリーズで定義されたマーカー) が拡大表示されます。この境界線の外観を制御することはできません。制御する方法は、"mouseOverIndicator": "enabled" を false に設定することで、マーカーを無効にすることのみです。
- バブルグラフのマーカーは、灰色の境界線で描画されます。マーカーは拡大表示されません。マーカーサイズはデータに応じて変化します。
- その他のグラフタイプでは (例、棒グラフ、面グラフ、円グラフ、タグクラウド、ツリーマップ、ヒートマップ)、"mouseOverIndicator": "color" で定義された標準のハイライト表示が使用され、マーカーは表示されません。

次のコードは、インジケータのプロパティおよびデフォルト値を示しています。

```
"mouseOverIndicator": {
    "enabled": true,
    "marker": {
        "color": "red",
        "size": 25,
        "shape": "square",
        "rotation": 0,
        "border": {"width": 1, "color": "darkblue"}
        }}
```

構文　マウスオーバーインジケータのフォーマット設定

```
"mouseOverIndicator": {
    "enabled": boolean,
   "color": "string",
   "marker": {
      "color": "string",
      "size": number,
      "shape": "string",
      "rotation": number,
      "border": {
         "width": number,
         "color": "string",
         "dash": "string"
      }
   }
},
```

説明

`"enabled": boolean`

マウスオーバーインジケータを有効または無効にします。有効な値には、次のものがあります。

- true - マウスオーバーインジケータを有効にします。これがデフォルト値です。
- false - マウスオーバーインジケータを無効にします。

`"color": "string"`

マウスポインタをライザ上に置いたときに適用する色を定義します。色は、色名または数値指定の文字列で定義することができます。また、色を "-100%" から "100%" までのパーセント文字列として定義することで、指定した割合の濃淡をシリーズ色に適用することができます。デフォルト値は undefined または「""」です。

"marker":
折れ線ライザのマーカーのプロパティを定義します。

"color": "string"

折れ線ライザのマーカー色を定義します (折れ線グラフ、複合グラフ、パレートグラフの場合)。色は、色名または数値指定の文字列で定義することができます。デフォルト値は "lightblue" です。

"size": number

マーカーのサイズを定義する数値です。デフォルト値は 5 です。

"shape": "string"

マーカー形状を定義する文字列です。デフォルト値は "square" です。

マーカー形状についての詳細は、427 ページの「 シリーズ固有のプロパティ 」 を参照してください。

"rotation": number

マーカーの回転角度 (度数) を定義する 0 (ゼロ) から 360 までの数値です。デフォルト値は 0 (ゼロ) です。

"border":
マーカー境界線のプロパティを定義します。

"width": number

マーカー境界線の幅を定義する数値をピクセル単位で指定します。デフォルト値は 1 です。

"color": "string"

色名または数値指定の文字列でマーカー境界色を定義します。デフォルト値は "darkblue" です。

"dash": "string"

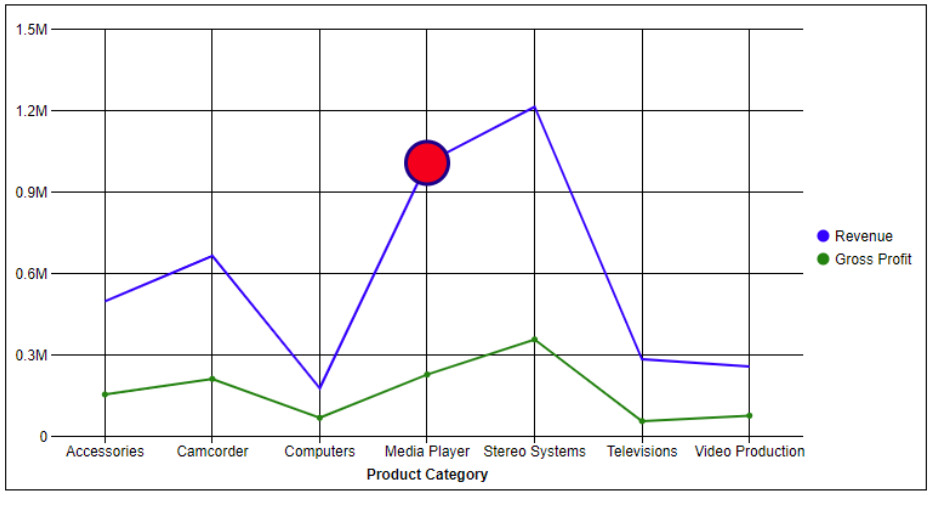
境界線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。デフォルト値は「""」です。この場合、実線が描画されます。

例　マウスオーバーインジケータのフォーマット設定

次のリクエストは、縦折れ線グラフを生成し、2 つのシリーズを表示します。シリーズ 0 にはマーカーを表示せず、シリーズ 1 にはデフォルトマーカーを表示します。マウスオーバーインジケータを赤色に設定し、マーカーを表示しない線のマーカー形状を丸、境界線を濃い青色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"series": [
   {"series": 0, "color": "blue", "marker": {"visible": false}},
   {"series": 1, "color": "green"}],
"mouseOverIndicator": {
   "enabled": true,
   "marker": {
      "color": "red", "size": 35, "shape": "circle", "rotation": 0,
      "border": {"width": 3, "color": "darkblue"} }
      }
*END
ENDSTYLE
END
```

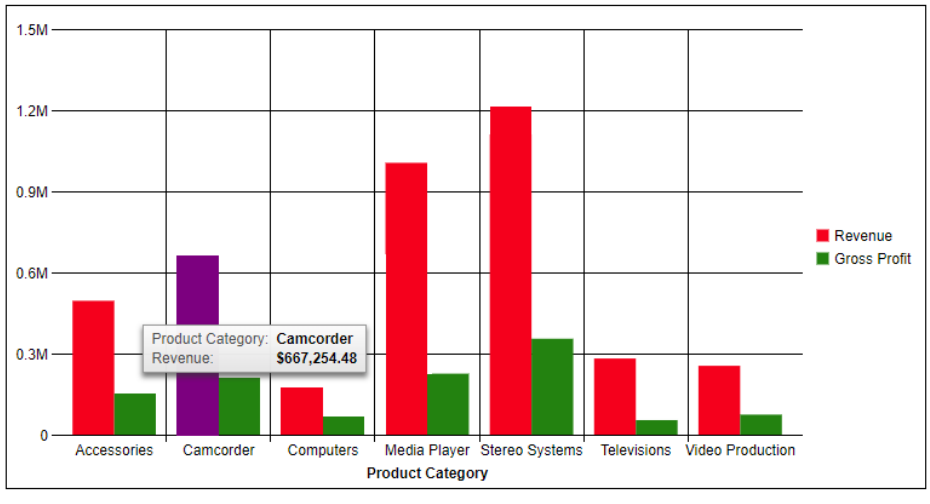
下図は、マーカーが表示されていない線でのマウスオーバーインジケータを示しています。



次のリクエストは、縦棒グラフを生成し、マウスオーバーインジケータを紫色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"mouseOverIndicator": {"enabled": true, "color": "purple"}
*END
ENDSTYLE
END
```

下図にように、マウスポインタをライザ上に置くと、ライザが紫色に変わります。



参照線の描画 (referenceLines)

referenceLines プロパティは、軸に描画する参照線を定義します。

注意：軸に自動目盛りが割り当てられている場合で、表示された目盛りの最小値より小さい、または最大値より大きい値で参照線を定義した場合は、参照線が表示されるよう目盛りの最小値または最大値が調整されます。

構文　　参照線の描画

```
"referenceLines": [
   {
   "value": number,
   "axis": "string",
   "line": {
      "color": "string",
      "width": number,
      "dash": "string"
   },
   "label": {
      "text": "string",
      "font": "string",
      "color": "string"
   },
   "anchor": "string",
   "showValue": boolean   }
]
```

説明

`"value": number`

軸に描画する線の位置を数値または文字列で定義します。

- 数値軸の場合、数値は軸に表示される値にする必要があります。文字列は "0%" から "100%" までのパーセントで指定する必要があります。
- X 軸の場合、文字列は軸に表示されるグループラベルにする必要があります。数値は 0 (ゼロ) から 1 までの値で指定する必要があります。たとえば、0.5 と指定すると、参照線がグラフの中央に描画されます。

`"axis": "string"`

線を描画する軸を定義する文字列です。"x"、"y"、"y2"、undefined (線を描画しない) のいずれかを指定します。

`"line":`

参照線のプロパティを定義します。

`"color": "string"`

色名または数値指定の文字列で線の色を定義します。デフォルト値は "black" です。

`"width": number`

線の幅をピクセル単位で定義します。デフォルト値は 1 です。

"dash": "string"

線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。実線の場合、「""」を使用します。

"label":

参照線ラベルのプロパティを定義します。

"text": "string"

参照線の横に描画するラベルを定義する文字列です (オプション)。「""」を使用するか (未定義のブランク文字列)、ラベルを描画しない場合は undefined を指定します。

"font": "string"

ラベルのサイズおよびタイプフェースを定義する文字列です。デフォルト値は "7.5pt Sans-Serif" です。

"color": "string"

色名または数値指定の文字列でラベルの色を定義します。

"anchor": "string"

参照線を基準に参照ラベルの描画位置を定義する文字列です。有効な値は、"start" (軸ラベルと同じ側) または "end" (軸ラベルと反対側) です。

- "start" - 軸ラベルと同じ側に参照ラベルを描画します。
- "end" - 軸ラベルと反対側に参照ラベルを描画します。これがデフォルト値です。

"showValue": boolean

参照線の値を表示するかどうかを指定します。有効な値には、次のものがあります。

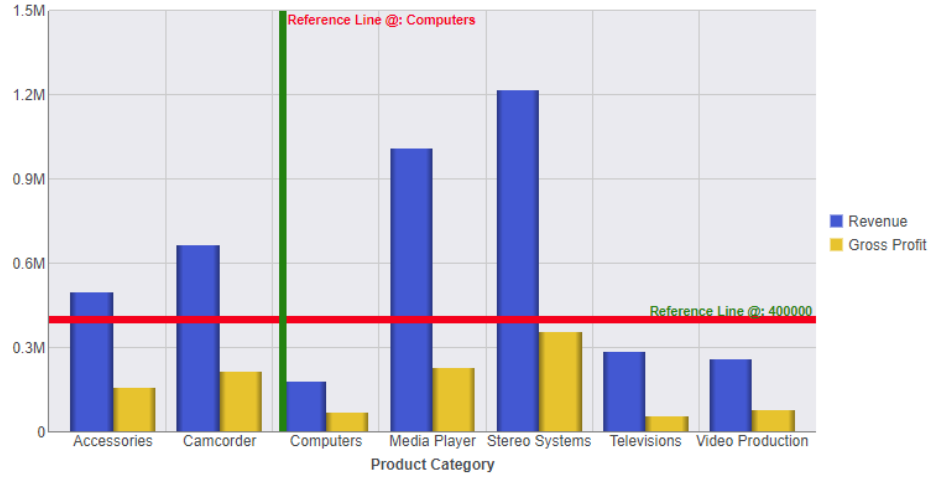
- true - 参照線を基準に指定された位置に値を表示します。
- false - 値を表示しません。これがデフォルト値です。

例　参照線の描画

次のリクエストは、縦棒グラフを生成し、2 本の参照線を描画します。Y 軸の参照線は、値が 400,000 の位置に赤色で描画し、その値を表示するラベルを緑色に設定します。X 軸の参照線は、値が Computers の位置に緑色で描画し、その値を表示するラベルを赤色に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=x-axis,$
type=data, column=n2, bucket=y-axis,$
type=data, column=n3, bucket=y-axis,$
*GRAPH_JS
"referenceLines": [
   {"value": 400000, "axis": "y",
      "line": {"color": "red", "width": 6, "dash": ""},
      "label": {"text": "Reference Line @:",
         "font": "bold 8pt Sans-Serif", "color": "green"},
      "anchor": "end", "showValue": true},
   {"value": "Computers", "axis": "x",
      "line": {"color": "green", "width": 6, "dash": ""},
      "label": {"text": "Reference Line @:",
         "font": "bold 8pt Sans-Serif", "color": "red"},
      "anchor": "end", "showValue": true}]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



バブルグラフおよび散布図での傾向線の描画 (trendline)

バブルグラフおよび散布図では、trendline プロパティを使用して傾向線および数式ラベルを描画します。

注意：特定のシリーズのみの傾向線を描画することもできます。詳細は、427 ページの 「 シリーズ固有のプロパティ 」 を参照してください。

次のコードは、傾向線のプロパティおよびデフォルト値を示しています。

```
"trendline": {
    "enabled": false,
    "mode": "undefined",
    "lineStyle": {
        "width": 1,
        "color": "black",
        "dash": ""
        },
    "equationLabel": {
        "visible": false,
        "font": "8pt Sans-Serif",
        "color": "black",
        "mode": "equation"
        }
    }
```

構文　　バブルグラフおよび散布図での傾向線の描画

```
"trendline": {
    "enabled": boolean,
    "mode": "string"
    "lineStyle": {
         "width": number,
         "color": "string",
         "dash": "string"
     },
    "equationLabel": {
        "visible": boolean,
        "font": "string",
        "color": "string",
        "mode": "string"
     }
   }
```

説明

`"enabled": boolean`
傾向線を有効または無効にします。有効な値には、次のものがあります。

- true - 傾向線を有効にします。
- false - 傾向線を無効にします。これがデフォルト値です。

`"mode": "string"`
傾向線モードを定義する文字列です。傾向線モードは、傾向線の生成に使用する数式のタイプです。有効な値には、次のものがあります。

- "exponential"
- "geometric"
- "hyperbolic"
- "linear"
- "logarithmic"
- "logQuadratic"
- "modExponential"
- "modHyperbolic"
- "polynomial"
- "quadratic"

- "rational"
- undefined - これがデフォルト値です。

"lineStyle":
傾向線のプロパティを定義します。

"width": number

傾向線の幅を定義する数値をピクセル単位で指定します。デフォルト値は 1 です。

"color": "string"

色名または数値指定の文字列で傾向線の色を定義します。デフォルト値は "black" です。

"dash": "string"

傾向線の破線スタイルを定義する文字列です。破線長さと破線間隔を定義する数値を順に列記し、文字列として指定します。たとえば、「"dash": "1 1"」と指定すると、破線長さ 1 ピクセル、破線間隔 1 ピクセルの点線が描画されます。デフォルト値は「""」です。この場合、実線が描画されます。

"equationLabel":
数式ラベルのプロパティを定義します。

"visible": boolean
数式ラベルの表示を制御します。有効な値には、次のものがあります。

- true - 数式ラベルを表示します。
- false - 数式ラベルを表示しません。これがデフォルト値です。

"font": "string"

数式ラベルのサイズ、スタイル、タイプフェースを定義する文字列です。デフォルト値は "8pt Sans-Serif" です。

"color": "string"

色名または数値指定の文字列で数式ラベルの色を定義します。デフォルト値は "black" です。

"mode": "string"
数式ラベルのモードを定義する文字列です。有効な値には、次のものがあります。

- "equation" - 生成される傾向線のタイプに応じて、最適なフォーマットで数式を表示します。直線の傾向線の場合、傾きと切片で表現するフォーマットで数式が表示されます。これがデフォルト値です。

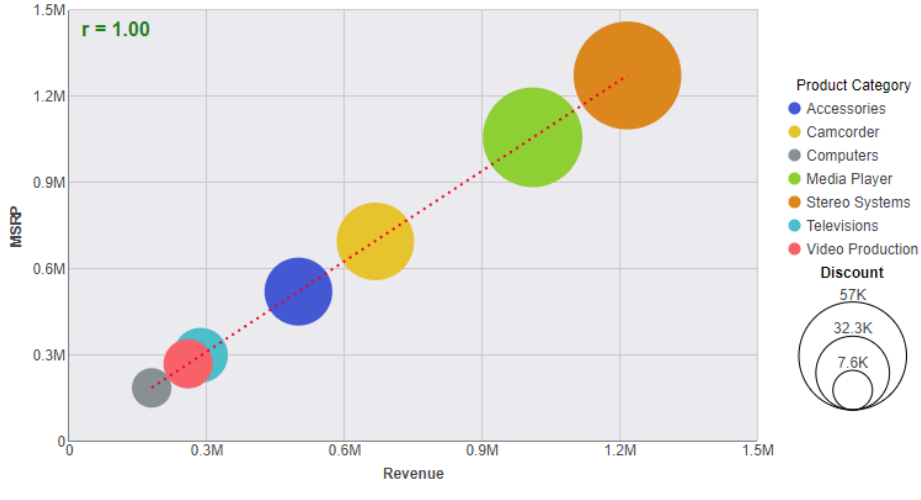
- "r" - 相関係数を表示します。

例　バブルグラフおよび散布図での傾向線の描画

次のリクエストは、バブルグラフを生成し、直線の傾向線を定義します。傾向線は赤色の破線で、相関係数を示す数式ラベルを緑色で表示します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US MSRP_US DISCOUNT_US
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
type=data, column=n1, bucket=color,$
type=data, column=n2, bucket=x-axis,$
type=data, column=n3, bucket=y-axis,$
type=data, column=n4, bucket=size,$
*GRAPH_JS
"trendline": {
   "enabled": true, "mode": "linear",
   "lineStyle": {"width": 2, "color": "red", "dash": "2 4"},
   "equationLabel": {
      "visible": true, "font": "bold 12pt Sans-Serif",
      "color": "green", "mode": "r"}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



13

マップサポート

WebFOCUS には、基本的な GIS 機能が組み込まれています。州、国、郵便番号などの位置情報または経度と緯度の値に関連付けられたデータは、強力なマップビューアに統合されたシンボルレイヤとして表示することができます。一般にサポートされるフォーマットには、地図上にマーカーが描画されるプロポーショナルシンボルマップ (バブルマップとも呼ばれる) やコロプレス (地図上の領域が多角形で色分けされる) があります。

注意：マップリクエストには、厳格な JSON 構文を使用する必要があります。つまり、すべてのプロパティ名およびオブジェクト名と、すべての文字列値は、二重引用符 (") で囲む必要があります。

トピックス

- マップサポートの概要
- マップグラフの TIBCO WebFOCUS パラメータおよび属性カテゴリ
- マッププロパティの概要
- ESRI マップグラフの作成
- Leaflet マップグラフの生成
- マップグラフでのグラフプロパティの統合

マップサポートの概要

WebFOCUS の地図作成アーキテクチャは、ESRI (Environmental Systems Research Institute, Inc.) と Leaflet の 2 つの HTML5 マップビューアエンジンを提供します。どちらのエンジンも、ベースマップとして ESRI マップタイルを使用します。また、どちらのエンジンも 19 段階の詳細ズームレベルを備えたズーム、パン、縮尺コントロール機能を提供します。

- ESRI マップビューアでは、Leaflet マップビューアより正確なジオメトリと豊富なシンボル生成が可能です。ESRI マップビューアを使用すると、ESRI サーバ上で利用可能なデータ (人口密度データなど) による人口統計レイヤや、地図上の地域の外枠を示すフィーチャレイヤをグラフに組み込むことができます。また、API によるカスタマイズオプションを多数提供します。ただし、ファイアウォールやセキュリティポリシーの問題が発生した場合に、ESRI ビューアでは内部リダイレクトが難しいのに対し、Leaflet マップビューアでは内部リダイレクトが可能です。また、ESRI のオプションには有料のものもあります。有料のサービスまたはデータが必要な場合は、ESRI から追加サービスまたはローカルサーバのライセンスを取得することができます。

  ESRI の HTML5 グラフは、WebFOCUS グラフ属性構文でのみサポートされます。詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

- Leaflet のマップビューアには、ESRI が提供するデフォルト設定のマッピングサーバが含まれます。さらに、Google、UrbanMaps など他のマップサービスからマップタイル (ビットマップ) を簡単にインポートすることができます。Leaflet エンジンでサポートされるマップサービスおよびその他の機能についての詳細は、http://leafletjs.com/ を参照してください。

  Leaflet の HTML5 グラフは、WebFOCUS グラフ属性構文と従来の WebFOCUS 構文の両方をサポートします。グラフ属性構文を使用する場合は、属性カテゴリは ESRI グラフと同じです。

すべての HTML5 ビジュアライゼーションと同様に、地図上でハイライトされたマーカーおよび領域では、追加のグラフプロパティおよび WebFOCUS リクエスト機能 (例、ドリルダウン、マルチドリルダウン、情報ツールヒント) がサポートされます。また、マップサーバを構成し、GeoJSON および JavaScript ファイルを使用することで、ユーザ独自の位置情報およびマップ定義を追加することができます。

マップグラフの TIBCO WebFOCUS パラメータおよび属性カテゴリ

グラフ属性構文を使用する場合、WebFOCUS リクエストの各フィールドには適切な属性カテゴリを割り当てる必要があります。使用するカテゴリはグラフタイプにより異なります。

構文　マップグラフのグラフタイプ指定

コロプレスマップを生成するには、WebFOCUS リクエストで次のコマンドを指定します。

```
ON GRAPH SET LOOKGRAPH CHOROPLETH
```

プロポーショナルシンボルマップ (バブルマップ) を生成するには、WebFOCUS リクエストで次のコマンドを指定します。

```
ON GRAPH SET LOOKGRAPH BUBBLEMAP
```

説明

CHOROPLETH

地図上の特定の地域を色コード値または不連続の色指定値で塗りつぶします。

BUBBLEMAP

地図上にプロポーショナルシンボル (バブル) を描画します。

参照　マップグラフの属性カテゴリ

ESRI マップエンジンを使用するためには、WebFOCUS グラフリクエストでグラフ属性構文を使用する必要があります。Leaflet マップは、従来のグラフ構文のほかグラフ属性構文もサポートします。

WebFOCUS グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

下表は、マップグラフの各タイプで必要な属性カテゴリを示しています。

コロプレス	バブルマップ	説明
location **注意** ❑ コロプレスでは、位置が多角形でなく単一の点で示される場合は、バブルマーカー (円) を描画してその位置を示します。その場合、マーカーはすべて、WebFOCUS スタイルシートで定義されたデフォルト設定のマーカーサイズに統一されます。また、color カテゴリに割り当てられたフィールドがマーカーの色指定に使用されます。 ❑ Reporting Server 構文を使用して ESRI マップグラフを生成する場合、ソートフィールドは、location 属性カテゴリではなく ESRI プロパティに割り当てられます。	location、point、coordinates	コロプレスマップ上で色分けされる領域、またはバブルマップ上のシンボルの配置を定義します。これらはリクエスト内のソートフィールドを使用します。
color	color (オプション)	コロプレス上の領域の色またはバブルマップ上のシンボルの色は、color 属性カテゴリに割り当てられたフィールドに基づきます。フィールドがメジャーの場合、色はグラデーションとして表示されます。フィールドがディメンションの場合、色は不連続の色で表示されます。

コロプレス	バブルマップ	説明
	size	バブルマップ上のシンボルのサイズは、size 属性カテゴリに割り当てられたメジャー値に基づきます。

たとえば、次のリクエストは、バブルマップの緯度と経度のソートフィールドおよびサイズメジャーを指定し、適切な属性カテゴリにフィールドを割り当てます。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=STATE_PROV_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=STATE_PROV_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
...
```

シンボルおよびベースマップを表示するには、WebFOCUS スタイルシートにマッププロパティを追加する必要があります。詳細は次のセクションを参照してください。

マッププロパティの概要

WebFOCUS では、フレーム内のマップの範囲、デフォルト設定の最大ズームレベル、最小ズームレベルなど多数のマップ属性にデフォルトプロパティ値を提供します。また使用されるジオメトリの中央位置も特定します。ただし、ESRI マッププロパティはすべて API を使用したカスタマイズが可能です。

また、WebFOCUS グラフの全般的なプロパティを使用して、マップグラフを強化することもできます。

マップレイヤの理解

マップグラフでは、ベースレイヤがマップタイルを表示します。オーバーレイレイヤはベースマップ上にデータを追加します。

WebFOCUS リクエストの出力は、色やシンボルが重ねて表示される領域のメジャー値および地名または緯度と経度の値を提供します。色分けされた領域またはシンボルは、1 つのオーバーレイレイヤを形成します。

ESRI マップエンジンは、マップグラフに追加可能なその他のオーバーレイレイヤを提供します。その他のオーバーレイレイヤは次のいずれかです。

- **タイル (人口統計) レイヤ** - これらのレイヤは、ESRI サーバ (場合によってはローカルサーバ) 上で利用可能な事前集計された人口統計データで構成されます。
- **ダイナミックフィーチャレイヤ** - これらのレイヤは、動的に集計される人口統計データを提供します。
- **参照フィーチャレイヤ** - これらのレイヤは、地図上にシンボルを配置します。WebFOCUS は、参照フィーチャレイヤを使用して地図上に領域の外枠を描画します。

WebFOCUS では、ベースレイヤとオーバーレイレイヤが 1 つずつサポートされます。ESRI マップエンジンを使用する場合は、複数の人口統計レイヤおよびフィーチャレイヤを追加することができます。

WebFOCUS は、ESRI からライセンスを取得して、ESRI サーバまたはローカルサーバ上に存在する ESRI ArcGIS マップにアクセスします。ユーザが別のマッププロバイダのライセンスを所有している場合は、そのプロバイダのマップにアクセスすることもできます。

また、WebFOCUS とともに無償の Mapbox OpenStreet マップがインストールされます。これらのマップは、ユーザのシステムに ESRI マップをインポートできない場合に Leaflet マップの代わりに使用することができます。

地理位置情報とデータソース内のフィールドとの接続は、Leaflet マップの場合は JSON、ESRI、または geoJSON ロケーションファイルへの URL (エンドポイントとも呼ばれる)、ESRI マップの場合は geoJSON または ESRI エンドポイントをそれぞれ指定することで実行されます。

JSON ロケーションファイルは WebFOCUS から提供されますが、マップのプロパティでファイルの参照先を調整することで、別の位置情報や別タイプのロケーションファイルを追加することもできます。

ロケーションのエンドポイントには、州や市など主要行政区分のリストのほか、これら区分の境界を定義する緯度と経度が含まれます。

ESRI マップグラフの作成

ESRI マップグラフは、WebFOCUS スタイルシートの一連の拡張機能 (外部グラフコンポーネントへのポインタ) および GRAPH_JS ブロックのプロパティを使用して定義されます。これらのスタイルシート属性を使用して、使用するベースマップ、追加する人口統計レイヤとフィーチャレイヤ、および WebFOCUS リクエスト出力により生成されるオーバーレイレイヤのプロパティを指定します。

ESRI マップグラフの作成に使用できる構文には次の 2 つのタイプがあります。

- **グラフエンジン構文**　この構文では、ESRI が正確な地理オブジェクトおよびイメージを返すために必要なプロパティをすべて追加することができます。

  この構文を使用する場合、地理位置情報、人口統計レイヤ、フィーチャレイヤ、ベースマップを ESRI から WebFOCUS スタイルシートに取得するために必要なパラメータおよびポインタをマップサービスに追加する必要があります。

  地理位置情報ベースのグラフでは、リクエストに使用されるロケーションエンドポイントの URL および位置情報フィールドを提供する必要があります。緯度と経度の値に基づくバブルマップリクエストでは、エンドポイントは必要ありません。WebFOCUS でシンボルの座標を描画することができます。

  グラフエンジン構文についての詳細は、906 ページの 「 グラフエンジン構文による ESRI マップグラフの生成 」 を参照してください。

- **Reporting Server 構文**　Reporting Server には地理構成ファイルが同梱されています。このファイルには、すべての標準地理的役割、ベースマップ、人口統計レイヤ、参照レイヤのプロパティが格納されています。この構成ファイルを使用することで、WebFOCUS スタイルシートでの指定が必要なプロパティの数を大幅に減らすことができます。Reporting Server はこの構成ファイルを使用して、グラフエンジン構文を生成します。地理構成ファイルを編集して、地理的役割リスト、ベースマップ、レイヤをカスタマイズすることができます。

  Reporting Server 構文についての詳細は、878 ページの 「 TIBCO WebFOCUS Reporting Server 構文による ESRI マップグラフの生成 」 を参照してください。

ここでは、ESRI マップグラフの必須プロパティおよびよく使用する追加プロパティについて説明します。必須プロパティのほかに、ESRI API で利用可能なプロパティを追加することもできます。これらは ESRI に直接送られ、処理されます。ESRI API についての詳細は、

https://developers.arcgis.com/javascript/jsapi/map-amd.html を参照してください。

ESRI のエンドポイントはすべて、特定のジオメトリを示します。たとえば、World Countries のエンドポイントは国境を指定します。

詳細は、https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/World_Countries_(Generalized)/FeatureServer/0 を参照してください。

このエンドポイントを使用するリクエストは、国を表す位置のみを地図上に示すことができるため、location 属性カテゴリに割り当てられるソートフィールドは country の値を含むフィールドになります。エンドポイントに URL を使用した場合、エンドポイントの地理位置情報フィールド名が表示されます。

構文　ESRI マップグラフの WebFOCUS スタイルシート属性指定

WebFOCUS スタイルシートの TYPE=REPORT 宣言では、ESRI マッププロバイダを指定する必要があります。

```
TYPE=REPORT, CHART-LOOK=com.esri.map, $
```

また、リクエスト内のフィールドの属性カテゴリへの割り当ても必要です。詳細は、873 ページの「 マップグラフの属性カテゴリ 」 および次のセクションの「ESRI マップ拡張機能の指定」を参照してください。

TIBCO WebFOCUS Reporting Server 構文による ESRI マップグラフの生成

Reporting Server は、地理構成ファイルを使用して ESRI マップグラフリクエストを処理することができます。この地理構成ファイルには、ESRI が提供する標準の地理情報、レイヤ、ベースマップに必要なパラメータおよびポインタがすべて格納されています。利用可能なオブジェクトをカスタマイズするには、この構成ファイルを編集します。

TIBCO WebFOCUS Rorting Server 構文による ESRI マップレイヤの定義

WebFOCUS ESRI マップグラフでは、背景マップイメージを表示するベースレイヤを 1 つと、オーバーレイレイヤの配列を使用します。オーバーレイレイヤの 1 つには、WebFOCUS リクエストの出力が含まれます。これは、色分けされた多角形またはシンボルを配置するための位置情報または座標を提供します。その他のタイプのオーバーレイレイヤには、人口統計レイヤおよびフィーチャレイヤなどがあります。

サポート対象の地理的役割、人口統計レイヤ、フィーチャレイヤ、参照レイヤはすべて、Reporting Server の geo_services.xml ファイルで構成されます。詳細は、893 ページの「 TIBCO WebFOCUS Reporting Server の地理構成ファイル 」 を参照してください。Reporting Server は、グラフリクエストのソートフィールドを使用して地理的役割を識別します。次に、この地理的役割を使用し、地理構成ファイルで構成された URI パラメータに基づいて必要なマップサービスを特定します。地理的役割およびレイヤのリストは、地理構成ファイルを編集することでカスタマイズできます。

構文

ESRI マップ拡張機能の指定

WebFOCUS スタイルシートの TYPE=REPORT 宣言では、ESRI マッププロバイダを指定する必要があります。

```
TYPE=REPORT, CHART-LOOK=com.esri.map, $
```

また、ベースレイヤおよびオーバーレイレイヤを定義する拡張機能オブジェクトを、WebFOCUS スタイルシートの GRAPH_JS_FINAL ブロックに任意の順序で追加する必要があります。

```
"extensions": {
     "com.esri.map": {
        "overlayLayers": [
          array_of_overlay_layers
                            ],
       "baseMapInfo": {
           basemapproperties
                    }
                    }
             }
```

各レイヤタイプの定義に必要なプロパティについては、次のセクションで説明します。

構文

TIBCO WebFOCUS Reporting Server 構文によるベースレイヤの定義

geo_services.xml ファイルのベースマップリストからベースマップを選択するには、ベースマップオブジェクトの name プロパティで値を指定する必要があります。

```
"baseMapInfo": {
"customBaseMaps": [
       {
        "ibiBaseLayer": "basemapname"
       }
                  ]
             }
```

説明

```
"ibiBaseLayer": "basemapname"
```

geo_services.xml ファイルで構成されたベースマップの名前を指定します。以下はその例です。

```
"ibiBaseLayer": "gray"
```

1 つのリクエストにつき 1 つの ibiBaseLayer のみサポートされます。ただし、複数のベースマップを含むベースマップコントロールを表示し、これらを切り替えることができます。詳細は、956 ページの 「 ESRI マップグラフへのベースマップ情報の追加 」 を参照してください。

構文 TIBCO WebFOCUS Reporting Server 構文によるコロプレスデータレイヤの定義

コロプレスマップのデータレイヤを定義するには、位置ディメンションとして使用するフィールドをグラフリクエストの BY フィールドに指定し、このフィールドにマスターファイルまたは DEFINE コマンドの地理的役割を割り当てる必要があります。割り当てられた地理的役割は、geo_services.xml ファイルの role id の値と一致する必要があります。

```
{
    "ibiDataLayer": {
              "map-metadata": {
                   "map_by_field": "fieldname"
                                 }
                    }
}
```

説明

`"map-metadata"`

リクエストの BY フィールドが地理オブジェクトを表し、このフィールドにメタデータの地理的役割が割り当てられていることを示します。

`"map_by_field": "fieldname"`

リクエストの BY フィールドの名前を指定します。グラフリクエストの BY 句と同一のフィールド名を指定する必要があります。

注意： このフィールドは、WebFOCUS スタイルシートの属性カテゴリには割り当てられません。その代わりに、"ibiDataLayer" オブジェクトの "map_by_field" プロパティに割り当てられます。

例 TIBCO WebFOCUS Reporting Server 構文によるコロプレス ESRI マップの生成

次のリクエストは、BY フィールドとして STATE_PROV_NAME フィールドを使用します。WF_RETAIL_GEOGRAPHY マスターファイルでは、このフィールドには STATE 地理的役割が割り当てられています。

```
FIELDNAME=STATE_PROV_NAME, ALIAS=STATE_PROV_NAME, USAGE=A30V, ACTUAL=A30V,
      MISSING=ON,
      TITLE='State,Province', DESCRIPTION='Name of State or Province',
      GEOGRAPHIC_ROLE=STATE,  $
```

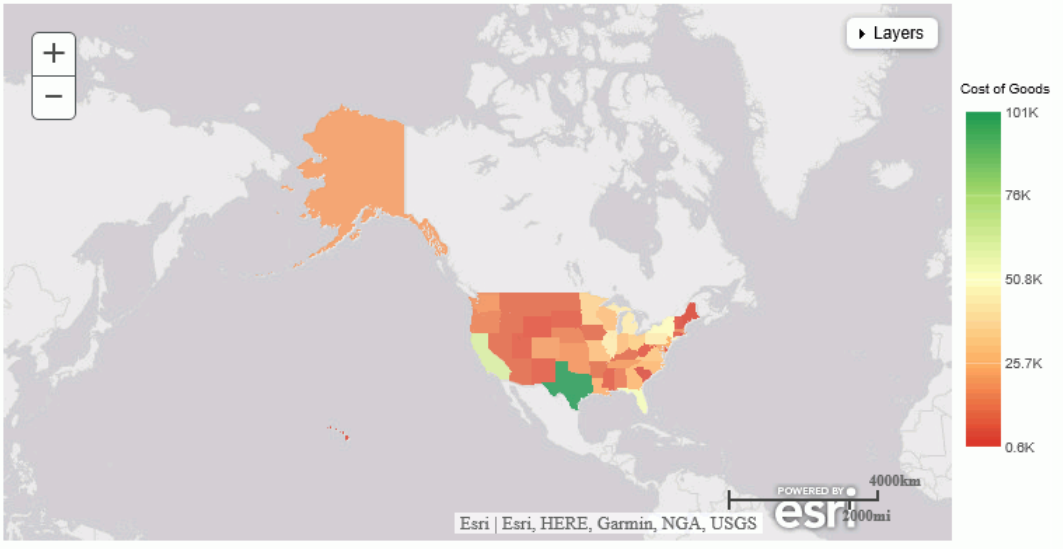
STATE 地理的役割に加えてベースマップ "gray" も、geo_services.xml ファイルで定義されています。

```
<geo_role id="STATE" type="alpha" role_name="STATE"
  role_name_title="State" role_format="NAME" role_format_title="Name"
geo_type="geography" unified="yes" > ...


<basemap custom="false" default="true" name="gray" title="Light Gray Canvas
Map" icon="qb/gray_map_108x72.png" />
```

次のリクエストでは、BY フィールドとして STATE_PROV_NAME を使用し、多角形は COGS_US フィールドで色分けされます。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "STATE_PROV_NAME"
                                            }
                                     }
        }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                          }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

構文　TIBCO WebFOCUS Reporting Server 構文によるプロポーショナルシンボルデータレイヤの定義

バブルマップのデータレイヤを定義するには、位置ディメンションとして使用するフィールドをグラフリクエストの BY フィールドおよび JSON ジオメトリポイントに指定する必要があります。GIS_POINT 関数を使用して、DEFINE コマンドでジオメトリポイントを作成することができます。この関数についての詳細は、『ibi™ WebFOCUS® 関数リファレンス』を参照してください。

```
{
    "ibiDataLayer": {
              "map-geometry": {
                   "map_by_field": "fieldname"
                              }
                    }
}
```

説明

`"map-geometry"`

リクエストの BY フィールドがジオメトリポイントを表すことを指定します。

`"map_by_field": "fieldname"`

リクエストの BY フィールドの名前を指定します。グラフリクエストの BY 句と同一のフィールド名を指定する必要があります。このフィールドは、WebFOCUS スタイルシートの属性カテゴリには割り当てられません。

例 Reporting Server 構文によるプロポーショナルシンボル ESRI マップの生成

次のリクエストは、STATE_PROV_CAPITAL_LONGITUDE フィールドおよび STATE_PROV_CAPITAL_LATITUDE フィールドを使用してジオメトリポイントを作成し、このポイントを BY フィールドとして使用します。

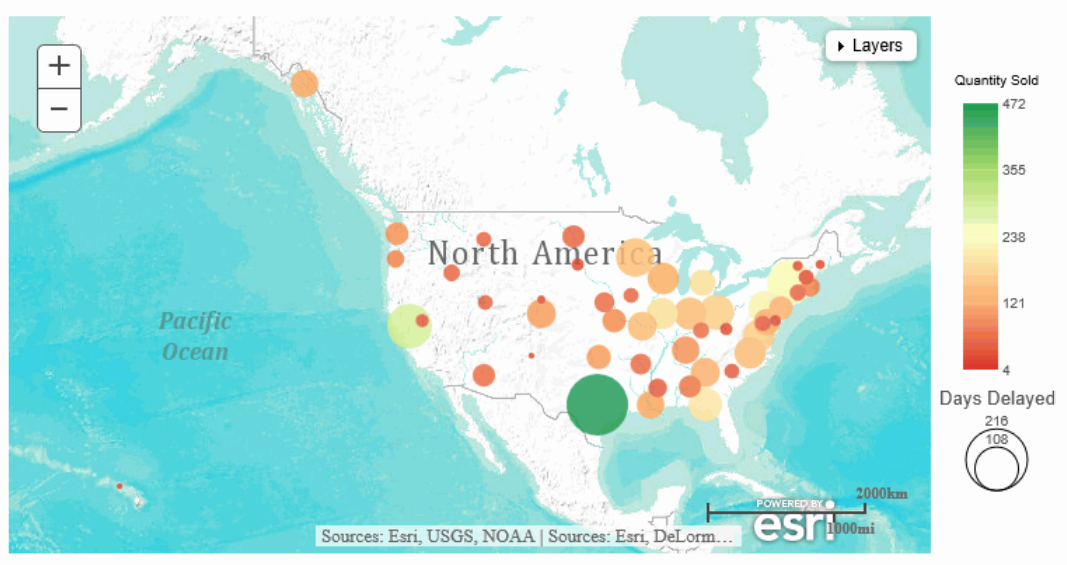
ベースマップの "terrain" は、geo_services.xml ファイルで定義されています。

```
 <basemap custom="false" default="false" name="terrain" title="Terrain with
Labels" icon="qb/terrain_map_108x72.png" />
```

次のリクエストでは、DEFINE FILE コマンドで作成された GEOPOINT が BY フィールドとして使用されます。また、シンボルのサイズは DAYSDELAYED フィールド、シンボルの色は QUANTITY_SOLD フィールドで定義されます。

```
DEFINE FILE WF_RETAIL_LITE
GEOPOINT/A200 = GIS_POINT('4326', STATE_PROV_CAPITAL_LONGITUDE,
STATE_PROV_CAPITAL_LATITUDE);
END
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY GEOPOINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *

INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=DAYSDELAYED, BUCKET=size, $
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                       "map_by_field": "GEOPOINT"
                                            }
                                     }
         }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "terrain"
                }
            ]
          }
                        }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

構文

TIBCO WebFOCUS Reporting Server 構文による人口統計およびフィーチャレイヤの ESRI マップグラフへの追加

人口統計および参照フィーチャレイヤを ESRI マップグラフに追加するには、geo_services.xml ファイルで定義された人口統計レイヤまたは参照レイヤの名前を提供するレイヤをオーバーレイレイヤに追加する必要があります。

```
{
    "ibiAddLayer": "layer_name"
}
```

説明

`"ibiAddLayer": "layer_name"`

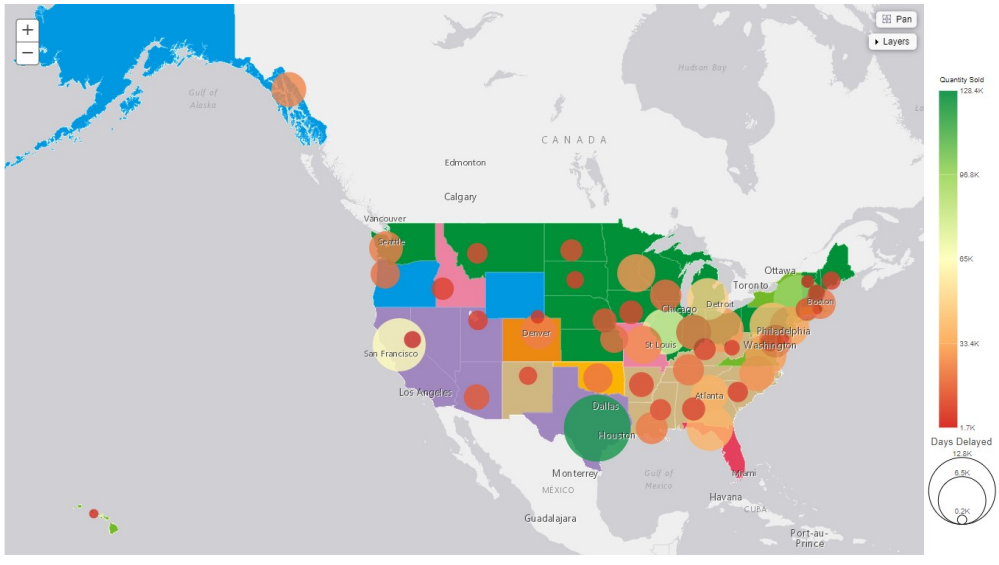
geo_services.xml ファイルの name プロパティの値を使用して、人口統計レイヤまたは参照レイヤの名前を指定します。

例　TIBCO WebFOCUS Reporting Server 構文による人口統計レイヤの ESRI マップへの追加

次のリクエストでは、BY フィールドとして、WF_RETAIL_GEOGRAPHY マスターファイルの DEFINE で作成された STATE_PROV_CAPITAL_POINT を使用します。また、このフィールドには GEOMETRY_POINT 地理的役割が割り当てられます。USA_Tapestry_Segmentation_2017 という名前の人口統計レイヤがグラフに追加されます。

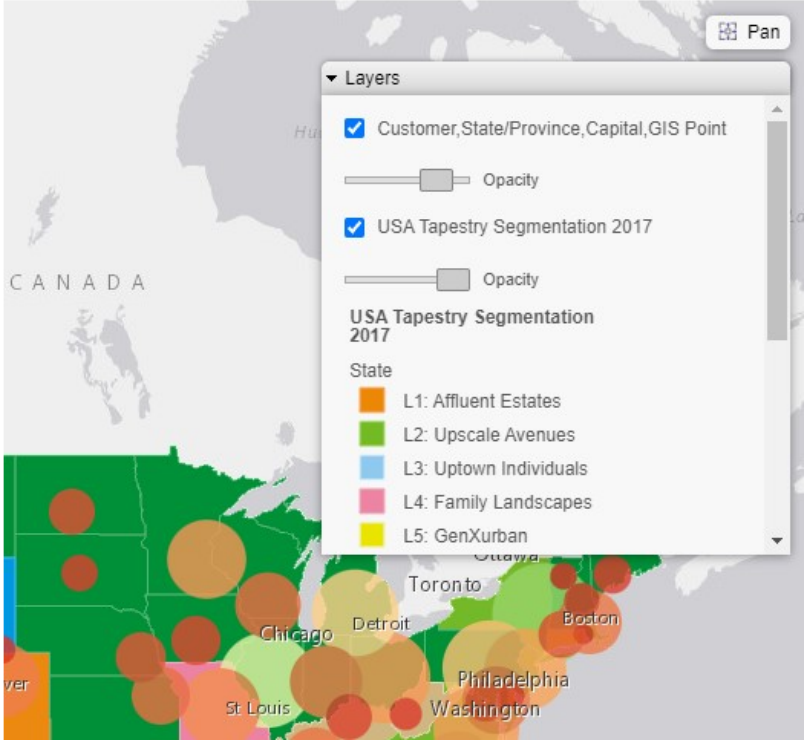
```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_CAPITAL_POINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
 ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=color, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                       "map_by_field":
"WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_CAPITAL_POINT"
                                                }
                                      }
         },
         {"ibiAddLayer": "USA_Tapestry_Segmentation_2017"}
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                        }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。プロポーショナルシンボルはデータレイヤを表し、多角形は人口統計レイヤを表します。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

下図のように、[目次] を開いて、人口統計レイヤの凡例を参照したり、レイヤを非表示にしたり、レイヤの不透明度を変更したりできます。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

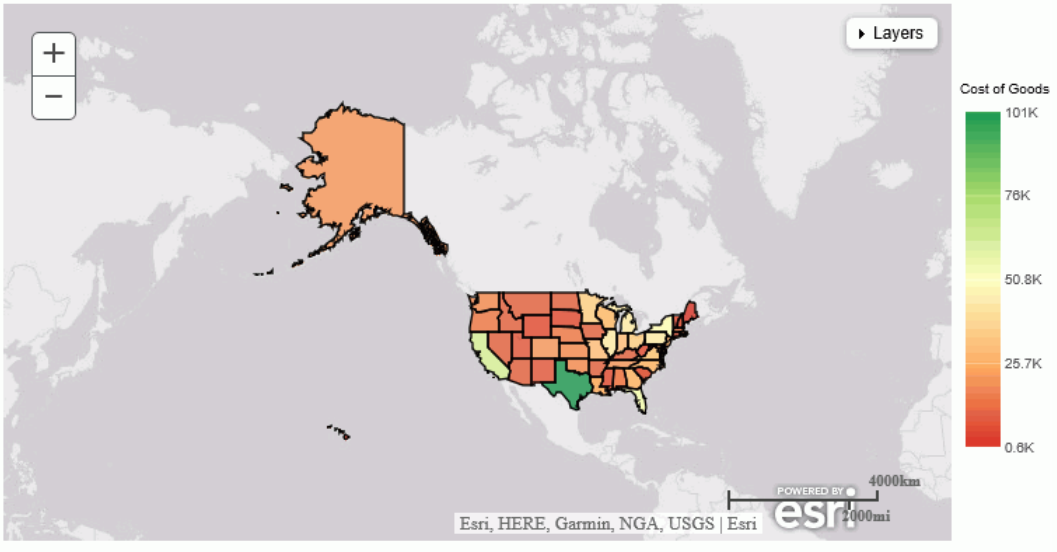
例

TIBCO WebFOCUS Reporting Server 構文による参照フィーチャレイヤの ESRI マップへの追加

次のリクエストは、BY フィールドの STATE_PROV_NAME を使用してコロプレス ESRI マップグラフを生成します。USA_States_Gen という名前の参照フィーチャレイヤがグラフに追加されます。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "STATE_PROV_NAME"
                                            }
                                     }
        },
       {"ibiAddLayer": "USA_States_Gen"}
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                          }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

構文　TIBCO WebFOCUS Reporting Server 構文による ESRI マップレイヤタイトルの変更

デフォルト設定では、[目次] (レイヤ) コントロールの各レイヤタイトルは、地理構成ファイルから取得されます。タイトルは、レイヤのプロパティに "title" プロパティを追加することで変更できます。

```
"overlayLayers":
    [
       {
        "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "fieldname"
                                       }
                                  },
                    "title": "data_layer_title"
        },
        {"ibiAddLayer": "layer name",
                    "title": "context_ref_layer title"
        }
      ],
```

説明

`"title": "data_layer_title"`

データレイヤの新しいタイトルを指定します。

`"title": "context_ref_layer title"`

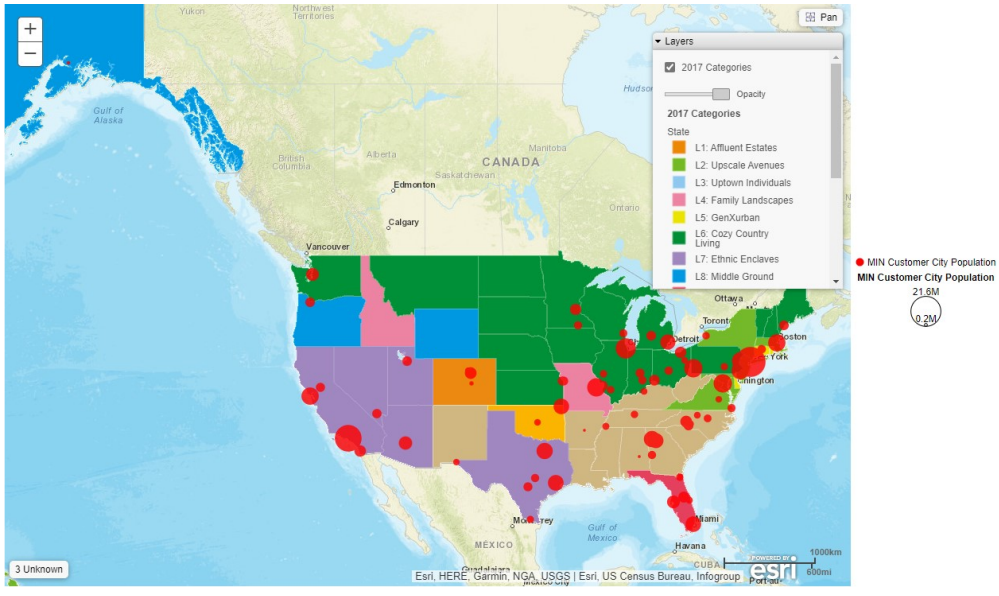
人口統計レイヤまたは参照レイヤの新しいタイトルを指定します。

例　TIBCO WebFOCUS Reporting Server 構文による ESRI マップレイヤタイトルの変更

次のリクエストは、データレイヤのタイトルを "City Population"、人口統計レイヤのタイトルを "2017 Categories" に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY CITY_NAME
WHERE CITY_POPULATION NE  ' ' OR MISSING
WHERE COUNTRY_NAME EQ 'United States'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS_FINAL
"bubbleMarker": {"maxSize":"5%", "color": "red"},

"extensions": {
  "com.esri.map": {
      "overlayLayers":
     [
        {
         "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "CITY_NAME"
                                       }
                                  },
                    "title": "City Population"
         },
         {"ibiAddLayer": "USA_Tapestry_Segmentation_2017",
           "title": "2017 Categories"
         }
       ],

      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "streets"
                }
            ]
          }
                          }
                   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

TIBCO WebFOCUS Reporting Server の地理構成ファイル

Reporting Server に同梱されている構成ファイル (geo_services.xml) には、地理的役割、地理階層、マップサービスの URL、ESRI マップビューアで選択可能なベースマップのすべてを定義する要素が記述されています。このファイルは、サーバの home ディレクトリ下の catalog ディレクトリに格納されています。

```
_edahome/catalog/
```

注意：このディレクトリをアプリケーションツリーに表示するには、Reporting Server コンソールの [アプリケーション] ページで、[フィルタ] メニューから [すべてのユーザの edahome、edaconf、edaprfu、scaroot、edatemp、edalog 、foccache を表示] を選択します。次に、Reporting Server コンソールで geo_services.xml ファイルを直接編集することができます。

リクエストで選択可能な地理的役割は、この構成ファイルを使用して動的に作成されます。ESRI マップリクエストで地理的役割が参照されると、構成ファイル内の各地理的役割の定義に基づいてメタデータと、ESRI に送信されるリクエストが生成されます。このリクエストが ESRI に送信され、適切なマップがダウンロードされた後、マップ上にマーカーまたは地理オブジェクトが配置されます。

地理的役割は、階層の一部にすることができます。たとえば、World 地理的役割は、大陸、国、州、都市で構成された階層の最上位にあります。これらの階層は、geo_services.xml ファイルにも記述されています。

geo_services.xml ファイルが格納されているアプリケーションディレクトリには、このファイルで構成された項目についての情報を表示する一連のレポートリクエストが格納されます。

カスタム地理的役割を追加するには、その地理的役割に必要なパラメータをこのファイルに追加する必要があります。

標準の xml 構文規則に準拠し、各要素を要素の開始タグおよび終了タグで囲み (<elementname>, </elementname>)、属性値を二重引用符 (") で囲みます。

参照　地理的役割の定義

地理的役割は、geo_services.xml ファイルの geo_roles オブジェクト下の geo_role 要素として定義されています。地理的役割は、次の項目で定義する必要があります。

- 構成ファイルで地理的役割を識別する ID
- 返されるデータのフォーマットおよび長さ
- 地理的役割の名前
- 地理的役割の表示タイトル (App Studio の [設定] パネルで選択する際のタイトル)
- オプションの地理的役割フォーマット (地理的役割に名前、短縮名などの複数のフォーマットを指定できる場合)
- フォーマットの表示タイトル
- 地理的役割のタイプ (国や州などの地理情報オブジェクトの場合は geography、点ジオメトリ、面ジオメトリ、線ジオメトリの場合は geometry、経度と緯度の場合は coordinate)
- オプションのボキャブラリルール要素 (地理的役割をメタデータのフィールドに関連付けるためのボキャブラリ要素を格納)

構文は次のとおりです。

```
<geo_role id="id" value_size="size" unified="false" role_name="rname"
        role_name_title="rname_title" role_format="rformat"
        role_format_title="rformat_title" geo_type="gtype"
        type="datatype" >
        <vocabulary_rules>
            <vocabulary>vrule</vocabulary>
         </vocabulary_rules>
      </geo_role>
```

説明

`id="id"`

地理的役割の識別に使用する文字値を最大 50 バイトの大文字で指定します。

`type="datatype"`

ID のデータタイプです。次のいずれかの値です。

- **"alpha"** A*n* または I*n* フォーマットの文字データに使用します。
- **"integer"** I*n* フォーマットの整数値データに使用します。
- **"numeric"** P*n.m*、D*n.m*、F*n.m* フォーマットの小数値データに使用します。
- **"text"** TX*n* フォーマットのテキストデータに使用します。

`value_size="size"`

USAGE フォーマット長のバイト数です (オプション)。設定されていない場合は任意の長さです。

`role_name="rname"`

地理的役割の名前です。

`role_name_title="rname_title"`

[設定] パネルで地理的役割を選択する際に表示されるタイトルです。

`role_format="rformat"`

地理的役割のフォーマットです (オプション)。このフォーマットは、名前、ISO コード、略名などの複数のフォーマットで地理的役割を参照できる場合に役立ちます。標準的な地理的役割のタイトルには次のものがあります。

- **NAME** 地理的役割が地理エンティティで定義されることを示します (例、Florida)。
- **ABBR** 地理的役割が短縮名で定義されることを示します (例、FL)。

- **ZIP3** 地理的役割が 3 桁の郵便番号で定義されることを示します。
- **ZIP5** 地理的役割が 5 桁の郵便番号で定義されることを示します。
- **FIPS** 地理的役割が FIPS (連邦情報処理標準) コードで定義されることを示します。
- **ISO2** 地理的役割が ISO (国際標準化機構) から発行された ISO 3166-2 コードで定義されることを示します。
- **ISO3** 地理的役割が ISO (国際標準化機構) から発行された ISO 3166-3 コードで定義されることを示します。
- **LINE** 地理的役割が 1 行で定義されることを示します (例、住所の番地)。
- **FULL** 地理的役割がフルネームで定義されることを示します (例、完全な住所)。

`role_format_title="rformat_title"`

地理的役割のフォーマットのタイトルです (オプション)。[設定] パネルには、地理的役割のタイトルとともに、このフォーマットタイトルが括弧内に表示されます (例、「State (Abbreviation)」)。

`geo_type="gtype"`

次の定義済み地理的役割タイプのいずれかです。

- **"geography"** 国名や州名などの地理情報オブジェクトに使用します。
- **"geometry"** 点ジオメトリ、線ジオメトリ、面ジオメトリなどのジオメトリオブジェクトに使用します。
- **"coordinate"** 緯度や経度などの座標に使用します。

`<vocabulary>vrule</vocabulary>`

一連のボキャブラリ要素で構成される要素です。これらのボキャブラリ要素で、地理的役割のフィールド名を明示的に定義します。これらのルールは、特定の地理的役割で最適な地理データを選択する際に使用されます。

ルール内の各要素は、ブール論理演算子 OR で接続されます (1 つの要素のみが条件を満たす必要あり)。各ボキャブラリ要素には、特殊文字で囲まれた単語が格納されます。ルール要素内の各単語は、ブール論理演算子 AND で接続されます (すべての単語が条件を満たす必要あり)。

任意の連続文字を表すプレースホルダとして、パーセント記号 (%) を単語の先頭または末尾に追加することができます。要素に複数の単語が含まれている場合は、各単語の先頭にプラス記号 (+) またはマイナス記号 (-) を追加する必要があります。プラス記号 (+) は、その単語がフィールド名に存在する必要があることを示します。マイナス記号 (-) は、その単語がフィールド名に存在できないことを示します。

geo_services.xml ファイルの geo_roles オブジェクトに、必要なパラメータを含む geo_role 要素を追加することで、カスタム地理的役割を追加することができます。地理的役割定義を追加した後、geo_srv_roles プロシジャを実行して、パラメータが正しく追加されたことを確認します。

例　サンプル地理的役割の定義

次の要素は、州名 (略名) に使用する地理的役割を定義します。地理的役割 ID は USSTATE_ABBR です。地理的役割名は USSTATE、地理的役割フォーマットは ABBR です。[設定] パネルに表示されるタイトルは「US state (Abbreviation)」です。フォーマットは A2 です。ボキャブラリルールで「state」という文字が存在すること、「iso」、「capital」、「population」という文字が存在しないことが条件に指定されています。地理的役割タイプは geography で、返されるデータが地域であることを示しています。

```
<geo_role
id="USSTATE_ABBR"
 value_size="2"
 type="alpha"
role_name="USSTATE"
role_name_title="US state"
role_format="ABBR"
role_format_title="Abbreviation"
 geo_type="geography">
<vocabulary_rules>
<vocabulary>+%state%-%iso%-%capital%-%population%</vocabulary>
</vocabulary_rules>
</geo_role>
```

次の要素は、緯度値に使用する地理的役割を定義します。地理的役割 ID は LATITUDE です。地理的役割の名前も LATITUDE です。フォーマットは numeric です。[設定] パネルに表示されるタイトルは「Latitude」です。地理的役割タイプは coordinate で、返されるデータが点であることを示しています。ボキャブラリルールでは、「latitude」という文字が存在することが条件として指定されています。

```
<geo_role
 id="LATITUDE"
 type="numeric"
 role_name="LATITUDE"
role_name_title="Latitude"
 geo_type="coordinate">
 <vocabulary_rules>
  <vocabulary>%latitude%</vocabulary>
 </vocabulary_rules>
</geo_role>
```

次の要素は、都市名に使用する地理的役割を定義します。地理的役割 ID は CITY です。地理的役割の名前も CITY です。フォーマットは NAME です。[設定] パネルに表示されるタイトルは「City (Name)」です。この要素には、一連のボキャブラリ要素が定義されています。リスト内の要素のいずれかが true である必要があります。そのため、「city」、「town」、「country と capital」、「state と capital」のいずれかの文字が存在する必要があります。

```
<geo_role
 id="CITY"
type="alpha"
role_name="CITY"
role_name_title="City"
role_format="NAME"
role_format_title="Name"
 geo_type="geography">
 <vocabulary_rules>
  <vocabulary>+%city%-%population%</vocabulary>
  <vocabulary>+%town%-%population%</vocabulary>
  <vocabulary>+%country%+%capital%-%population%</vocabulary>
  <vocabulary>+%state%+%capital%-%population%</vocabulary>
 </vocabulary_rules>
</geo_role>
```

参照　地理的役割の URI の定義

geo_services.xml ファイルにカスタム地理的役割を追加する場合、その URI も <URIS> オブジェクトに追加する必要があります。URI オブジェクトは、特定の地理的役割に必要なマップサービスを参照します。構文は次のとおりです。

```
<uri description="description">
   <returned_geometry>type </returned_geometry>
   <returned_georole>role</returned_georole>
   <url type="esri" authorization="auth" synonym="">
         "url_to_georole"
   <parameters>
    <parm order="number" parm_name="pname" parm_georole="parmrole"/>
   </parameters>
  </uri>
```

説明

`"description"`

URI の参照先となる地理的役割の説明です。

`type`

サポートされているジオメトリタイプのいずれかです (例、GEOMETRY AREA)。

`role`

返される地理的役割の名前です。

`"auth"`

この地理的役割へのアクセスに必要な認可のタイプです。有効な値には、次のものがあります。

- **silent** ArcGIS アプリケーションの認証情報は、ESRI アダプタの接続文字列に記述されています。
- **none** 認可は必要ありません。
- **named** ユーザの認証情報は、ESRI アダプタの接続文字列に記述されています。
- **on premise** ローカルホストの ArcGIS サーバのユーザ認証情報は、ESRI アダプタの接続文字列に記述されています。

`"url_to_georole"`

地理的役割の URL です。

`parm order="number"`

正しい地理的役割の取得に必要なパラメータの番号です。

`parm_name="pname"`

parm order に関連するパラメータの名前です。

リクエスト内の BY フィールドは、1 つ目のパラメータに対応する必要があります。このパラメータは、地理的役割と適切な URI 定義を一致させるために使用されます。

```
parm_georole="parmrole"
```

parm order に関連する地理的役割の名前です。

URI を追加した後、geo_srv_map_uris プロシジャを実行して、パラメータが正しく追加されたことを確認します。

参照 ベースマップの定義

次のサンプルは、geo_services.xml ファイルの標準ベースマップ定義を示しています。

```
<BASEMAPS>
...
 <basemap custom="false" default="false" name="streets"
    title="World Street Map" icon="qb/streets_map_108x72.png" />
 <basemap custom="false" default="false" name="satellite"
    title="World Imagery" icon="qb/imagery_map_108x72.png" />
 <basemap custom="false" default="false" name="terrain"
    title="Terrain with Labels" icon="qb/terrain_map_108x72.png" />
 <basemap custom="false" default="true" name="gray"
    title="Light Gray Canvas Map" icon="qb/gray_map_108x72.png" />
...
</BASEMAPS>
```

標準ベースマップに対して定義されているプロパティは、custom (標準ベースマップの場合は false)、default、name、title、icon です。

カスタムベースマップを定義するには、カスタムベースマップに対してこれらのプロパティを定義し、ベースマップイメージを提供するマップサービスの URL を追加する必要があります。

注意：追加するベースマップはすべて、タイルマップレイヤである必要があります。

また、<basemaps> オブジェクトからベースマップ定義を削除することで、使用可能なベースマップをカスタマイズすることもできます。ESRI On Premise を構成するには、参照先 URL がローカルホスト ESRI Server の外部を指しているベースマップ定義をすべて削除する必要があります。

ベースマップ定義の構文は次のとおりです。

```
<basemap custom="boolean" default="boolean" name="mapname"
  title="maptitle" icon="url_to_icon" url="url_to_basemap"
/>
```

説明

`custom="boolean"`

標準ベースマップまたはカスタムベースマップのいずれかを指定します。有効な値は、true または false です。カスタムベースマップを追加する場合は、custom="true" を指定します。

`default="boolean"`

ユーザがベースマップを選択しない場合に、このベースマップをデフォルトベースマップにするかどうかを指定します。有効な値は、true または false です。

`name="mapname"`

ベースマップの名前です。

`title="maptitle"`

ESRI ビューアの [プロパティ] パネルの [ベースマップ] ドロップダウンリストに表示されるタイトルです。

`icon="url_to_icon"`

ベースマップに使用するサムネールファイルのパスです。

`url="url_to_basemap"`

ベースマップを提供するマップサービスの URL です。

geo_services.xml ファイルの BASEMAPS オブジェクトに、必要なパラメータを含む basemap 要素を追加することで、カスタムベースマップを追加することができます。ベースマップ定義を追加した後、geo_srv_basemaps プロシジャを実行して、パラメータが正しく追加されたことを確認します。

参照

人口統計レイヤの定義

人口統計レイヤは、人口に関するデータを表示します。geo_services.xml ファイルでは、2 つのタイプの人口統計レイヤが構成されます。一方のタイプのレイヤでは、キャッシュされたデータが提供されます。このタイプの人口統計レイヤの layerType は "tile" です。もう一方のタイプの人口統計レイヤは、人口に関するデータを動的に返します。このタイプの人口統計レイヤの layerType は "featurelayer" です。各人口統計レイヤは、geo_services.xml ファイルの CONTEXTLAYERGROUPS オブジェクトのコンテキストレイヤグループ内のコンテキストレイヤとして定義されます。次の構文例は、人口統計コンテキストレイヤを示しています。

```
<contextlayer name="USA_Population_Density_2017" authorization="silent"
    layerType="featurelayer" r1txt_title="USA Population Density 2017" >
<addon_json>  "layerObjectType" : "esri/layers/
ArcGISDynamicMapServiceLayer",
        "smartMapping": {"webMapInfo": { "queryString" :
"id:b16d74fba48a4919aacfaac2cd77e5e5",
        "itemDataUrl" : "https://www.arcgis.com/sharing/rest/content/items/
b16d74fba48a4919aacfaac2cd77e5e5/data?f=json",
        "layerTypeEx" : "map-image" }}  </addon_json>
<uri>https://demographics7.arcgis.com/arcgis/rest/services/
USA_Demographics_and_Boundaries_2017/MapServer</uri>
</contextlayer>
```

構文は次のとおりです。

```
<contextlayer name="string" authorization="string"
    layerType="string" title="string" >
    <addon_json>addon_json</addon_json>
    <uri>uri_to_map_server</uri>
</contextlayer>
```

説明

`name="string"`

リクエストで参照されるレイヤの名前です。

`authorization="string"`

このレイヤへのアクセスに必要な認可のタイプを指定します。有効な値には、次のものがあります。

- **silent** ArcGIS アプリケーションの認証情報は、ESRI アダプタの接続文字列に記述されています。
- **none** 認可は必要ありません。
- **named** ユーザの認証情報は、ESRI アダプタの接続文字列に記述されています。

- **on premise** ローカルホストの ArcGIS サーバのユーザ認証情報は、ESRI アダプタの接続文字列に記述されています。

`layerType="string"`

キャッシュされた人口統計レイヤの場合、layerType は "tile" です。動的な人口統計レイヤの場合、layerType は "featurelayer" です。

`addon_json`

人口統計レイヤの表示に必要な追加の JSON プロパティです。たとえば、smartMapping プロパティは、レイヤ内の境界スタイルを定義します。

コンテキストレイヤのタイプによって、アドオン JSON プロパティは次のようになります。

```
"layerObjectType" : "esri/layers/ArcGISDynamicMapServiceLayer",
    "smartMapping": { "webMapInfo": {
        "queryString" : "id:id",
        "itemDataUrl" : "url",
        "layerTypeEx" : "layerType"
}}
```

説明

`esri/layers/ArcGISDynamicMapServiceLayer`

コンテキストレイヤをダイナミックマップサービスとして指定します。

`id`

ESRI ArcGIS のコンテキストレイヤの ID です。これは、ESRI ArcGIS サイトのこのレイヤの URL、またはこのレイヤのサービス URL からアクセス可能な JSON プロパティの "serviceItemId" プロパティの値から取得できます。

レイヤの ID が存在しない場合は、このプロパティは省略できます。

`url`

このレイヤのマップサービス URL です。これは、uri フィールドで指定する値と一致する必要があります。

`layerType`

コンテキストレイヤのタイプを指定します。次のいずれかの値です。

- **group** 複数のフィーチャレイヤで構成されたグループレイヤとして、コンテキストレイヤを識別します。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-GroupLayer.html を参照してください。

- **map-image** マップイメージレイヤとして、コンテキストレイヤを識別します。マップイメージレイヤには、フィーチャではなくイメージとして描画される複数のサブレイヤを含めることができます。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-MapImageLayer.html を参照してください。
- **feature** フィーチャレイヤとして、コンテキストレイヤを識別します。フィーチャレイヤは、マップサービスまたはフィーチャサービスから作成されたシングルレイヤです。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-FeatureLayer.html を参照してください。
- **imagery-tile** イメージタイルレイヤとして、コンテキストレイヤを識別します。イメージタイルレイヤは、タイルイメージから作成されます。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-ImageryTileLayer.html を参照してください。
- **imagery** イメージレイヤとして、コンテキストレイヤを識別します。イメージレイヤは、イメージサービスから作成されます。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-ImageryLayer.html を参照してください。
- **elevation** 3D シーンビューアーで使用される標高レイヤとして、コンテキストレイヤを識別します。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-ElevationLayer.html を参照してください。
- **tile** タイルレイヤとして、コンテキストレイヤを識別します。タイルレイヤは、キャッシュされたマップサービスから作成されます。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-TileLayer.html を参照してください。
- **web-map** 完全な 2D Web マップを識別します。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-WebMap.html を参照してください。
- **web-scene** 完全な 3D Web シーンを識別します。詳細は、https://developers.arcgis.com/javascript/latest/api-reference/esri-WebScene.html を参照してください。

たとえば、マップイメージレイヤのアドオン JSON は、次のようになります。

```
    "layerObjectType" : "esri/layers/ArcGISTiledMapServiceLayer",
    "smartMapping": { "webMapInfo": {
        "queryString" : "id:1234567890abcdefghij1234567890ab",
        "itemDataUrl" :
    "https://services5.arcgis.com/1234567890abcdef/arcgis/rest/services/
    Layer_Name/FeatureServer",

        }
    }
```

`uri_to_map_server`

人口統計レイヤのマップサーバの URI です。

参照　参照フィーチャレイヤの定義

参照フィーチャレイヤは、マップ上のジオグラフィックフィーチャに外枠を追加します。各参照レイヤは、geo_services.xml ファイルの CONTEXTLAYERGROUPS オブジェクトの Ref_layers という名前のコンテキストレイヤグループ内のコンテキストレイヤとして定義されます。次の構文例は、参照レイヤを示しています。参照レイヤの場合、layerType は "featurelayer" です。

```
<CONTEXTLAYERGROUPS>
    <contextlayergroup name="Ref_layers" title="Reference layers" >
      <contextlayer name="USA_ZIP" authorization="none"
layerType="featurelayer" title="USA ZIP" >
      <uri>https://services.arcgis.com/P3ePLMYs2RVChkJx/ArcGIS/rest/
services/USA_ZIP_Code_Points_analysis/FeatureServer/0</uri>
    </contextlayer>
```

構文は次のとおりです。

```
 <contextlayer name="string" authorization="string"
           layerType="featurelayer" title="string" >
           <uri>uri_to_map_server</uri>
      </contextlayer>
```

説明

`name="string"`

リクエストで参照されるレイヤの名前です。

`authorization="string"`

このレイヤへのアクセスに必要な認可のタイプを指定します。有効な値には、次のものがあります。

- **silent**　ArcGIS アプリケーションの認証情報は、ESRI アダプタの接続文字列に記述されています。

- **none**　認可は必要ありません。
- **named**　ユーザの認証情報は、ESRI アダプタの接続文字列に記述されています。
- **on premise**　ローカルホストの ArcGIS サーバのユーザ認証情報は、ESRI アダプタの接続文字列に記述されています。

`layerType="featurelayer"`

参照レイヤの場合、layerType の値は "featurelayer" です。

`uri_to_map_server`

フィーチャレイヤのマップサーバの URI です。

グラフエンジン構文による ESRI マップグラフの生成

ESRI マップグラフを生成するグラフエンジン構文には、地理情報、人口統計レイヤ、フィーチャレイヤ、ベースマップを ESRI から取得するのに必要なパラメータおよびマップサービスへの参照が含まれます。

各レイヤタイプのプロパティについては、次のセクションで説明します。

構文　グラフエンジン構文による ESRI マップ拡張機能の指定

グラフ内のすべてのレイヤおよびオブジェクトは、WebFOCUS スタイルシートの ESRI マップ拡張機能プロパティで定義されます。拡張機能は、外部データをリクエストに組み込むブロックです。

WebFOCUS では、ベースレイヤとオーバーレイレイヤの配列が 1 つずつサポートされます。オーバーレイレイヤの 1 つには、WebFOCUS リクエストの出力が含まれ、色分けされた多角形またはシンボルを配置するための位置情報または座標が含まれます。

位置フィールドに基づくオーバーレイレイヤのプロパティは、緯度と経度のフィールドに基づくオーバーレイレイヤのプロパティとは異なります。位置フィールドに基づくリクエストには、各ロケーション値の境界の緯度と経度の値を示すロケーションファイル (エンドポイント) が必要です。また、地理位置情報フィールドはエンドポイント内に指定する必要があります。緯度と経度の値を指定するリクエストではエンドポイントは必要ありません。WebFOCUS で座標を表示することができます。

人口統計レイヤは独自のエンドポイントを使用するため、地理位置情報フィールドの指定は必要ありません。

ベースレイヤとオーバーレイレイヤの定義はどちらが先でも構いません。次の構文では、オーバーレイレイヤが先に定義されています。

```
"extensions": {
    "com.esri.map": {
        "overlayLayers": [
           array_of_overlay_layers
            ],
        "baseLayer": {
            basemapproperties
                  }
   }
}
```

構文　グラフエンジン構文によるベースレイヤの定義

ベースレイヤの生成には、ベースマップ名のみが必要です。

```
 "baseLayer": {
      "title": "string",
      "basemap": "string"
              }
```

説明

`"title": "string"`

レイヤのタイトルです (オプション)。

`"basemap": "string"`

サポートされるベースマップの 1 つの名前です。有効な値には、次のものがあります。

- **dark-gray**　濃い灰色で背景を表示します。この背景を使用すると、他のレイヤを際立たせることができます。
- **dark-gray-vector**　濃い灰色の背景をベクタベースマップとして表示します。
- **gray**　薄い灰色 (中間色) で背景を表示します。
- **gray-vector**　薄い灰色 (中間色) の背景をベクタタイルベースマップとして表示します。
- **hybrid**　詳細な世界画像マップレイヤをラベル付きで表示します。
- **national-geographic**　National Geographic から提供される一般参照マップを表示します。
- **oceans**　海洋データ用の参照マップとして使用される背景を表示します。

- **osm** OpenStreetMap コミュニティマップレイヤを表示します。
- **satellite** 詳細な世界画像マップレイヤを表示します。
- **streets** 世界のマルチスケールストリートマップを表示します。
- **streets-vector** 世界のマルチスケールストリートマップをベクタベースマップとして表示します。
- **streets-navigation-vector** 簡素化されたストリートマップをベクタベースマップとして表示します。
- **streets-night-vector** 世界のマルチスケールストリートマップを暗色のベクタベースマップとして表示します。
- **streets-relief-vector** 世界のマルチスケールストリートマップを簡素化した地形情報のベクタベースマップとして表示します。
- **terrain** 地勢図をラベル付きで表示します。
- **topo** 地形図を表示します。
- **topo-vector.** 地形図をベクタベースマップとして表示します。

注意：初期中央位置、ズームレベルなどその他のベースレイヤプロパティについての詳細は、939 ページの 「 ESRI マップ出力のカスタマイズ 」 を参照してください。

構文 グラフエンジン構文による位置フィールドに基づくオーバーレイレイヤの定義

WebFOCUS リクエスト出力およびフィーチャレイヤは、次のプロパティを使用し、位置フィールドに基づいて定義されます。

```
{
   "title": "string",
   "layerType": "string",
   "quantizationThreshold": number,
   "geometrySourceType": "string",
   "geometryLocateField": ["string"],
   "geometryDataField": "name",
   "url": "url_to_endpoint"
   }
```

説明

`"title": "string"`

マップの [目次] コントロールに表示するタイトルです。

`"layerType": "string"`

レイヤのタイプです。有効な値には、次のものがあります。

- **choropleth** WebFOCUS コロプレスマップレイヤを表示します。
- **bubble** WebFOCUS バブルマップレイヤを表示します。
- **featurelayer** 地図上に外枠で囲まれた位置を表示します。参照フィーチャレイヤ上に位置の外枠を示すために使用するシンボルのプロパティ編集についての詳細は、939 ページの 「 参照フィーチャレイヤシンボルのフォーマット設定 」 を参照してください。参照フィーチャレイヤまたはダイナミックマップサービスレイヤのスタイル設定でレンダラを生成する smartMapping プロパティの追加についての詳細は、921 ページの「 グラフエンジン構文によるダイナミックマップサービスレイヤまたは参照フィーチャレイヤのスタイル設定 」 を参照してください。
- **tile** 事前集計およびキャッシュされた人口統計データを表示します。人口統計タイルレイヤについての詳細は、919 ページの 「 グラフエンジン構文による人口統計オーバーレイレイヤの定義 」 を参照してください。

`"quantizationThreshold": number`

範囲の狭い位置の描画に適用するしきい値をフィート単位で示します。このプロパティを設定しないと、長さや幅がしきい値より小さい郵便番号 (ZIP code) エリアは描画されません。返されたジオメトリが quantizationThreshold の値より小さい場合、quantizationParameters の設定なしでクエリが再送信されます。この方法を使用することで、狭い範囲の郵便番号 (ZIP code) エリアが適切に描画されます。デフォルト値は、10,560 フィート (2 マイル / 3.2km) です。

`"geometrySourceType": "string"`

使用するロケーションファイルのタイプを指定します。有効な値には、次のものがあります。

- ESRI
- geoJSON

`"geometryLocateField": ["string"]`

ロケーションエンドポイントの地理位置情報フィールド名です。エンドポイントに URL を指定する場合は、フィールド名は次のように記述されます。

```
"url": "url_to_endpoint"
```

ロケーションエンドポイントの URL です。

エンドポイントの例

次のエンドポイントは、WebFOCUS のインストール直後から WebFOCUS ツールで無料で使用することができます。その他のエンドポイントにアクセスする場合は、URL を指定してください。

World Countries:

https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/World_Countries_(Generalized)/FeatureServer/0

World Cities:

https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/World_Cities/FeatureServer/0

US Zipcodes - 3 digit:

https://services1.arcgis.com/szcgiQwor8m0ZTum/arcgis/rest/services/USA_3zip_codes/FeatureServer/0

US Zipcodes - 5 digit:

https://services.arcgis.com/P3ePLMYs2RVChkJx/ArcGIS/rest/services/USA_ZIP_Codes_2014/FeatureServer/0

US States:

https://services.arcgis.com/P3ePLMYs2RVChkJx/ArcGIS/rest/services/USA_States_Generalized/FeatureServer/0

US Cities:

https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/USA_Major_Cities/FeatureServer/0

US Counties:

https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/USA_Counties_Generalized/FeatureServer/0

Provinces and Territories of Canada

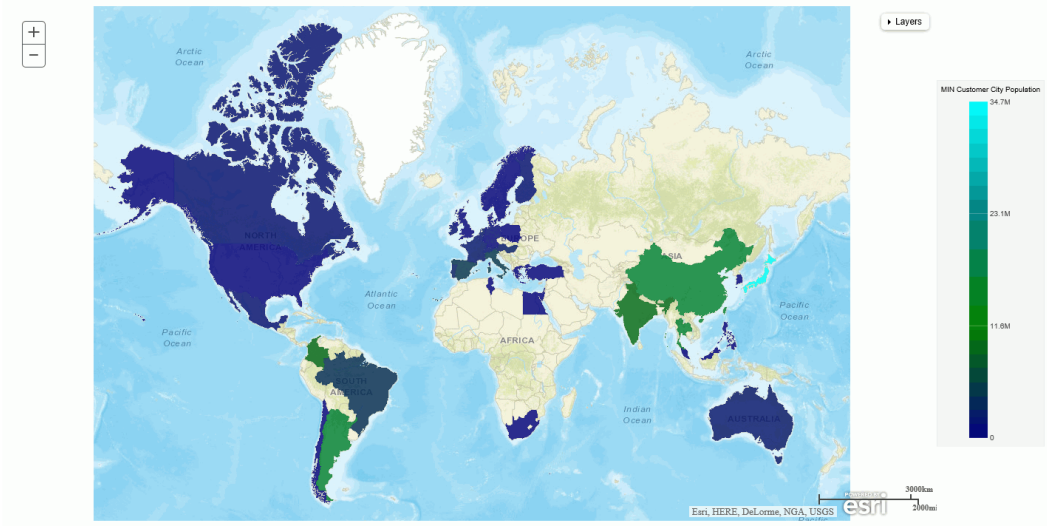
https://services.arcgis.com/zmLUiqh7X11gGV2d/ArcGIS/rest/services/Canada_Provinical_boundaries_generalized/FeatureServer/0

例　グラフエンジン構文による位置フィールドに基づく WebFOCUS オーバーレイレイヤの定義

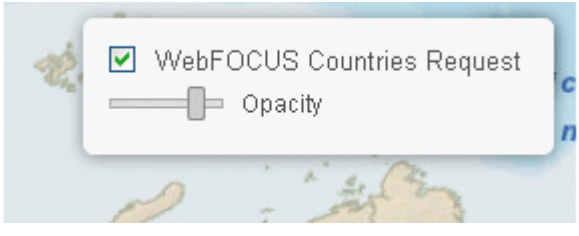
次のリクエストは、streets のベースレイヤと World Countries のエンドポイントを使用してコロプレスマップを生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY COUNTRY_NAME
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=color, $
*GRAPH_JS
"legend": {
    "visible": true
},
"colorScale": {"colors":["navy", "green", "teal", "cyan"]},
"extensions": {
"com.esri.map": {
"overlayLayers":
[
 {
 "title": "WebFOCUS Countries Request",
 "layerType": "choropleth",
 "geometrySourceType": "esri",
 "geometryLocateField": ["Country"],
 "geometryDataField": "name",
 "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
 }
 ],
 "baseLayer": {
 "basemap": "streets"
          }
}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフ右上の [目次] コントロールをクリックすると、オーバーレイレイヤのタイトル、これを表示するかどうかのチェックボックス、および不透明度を変更するためのスライダが表示されます。

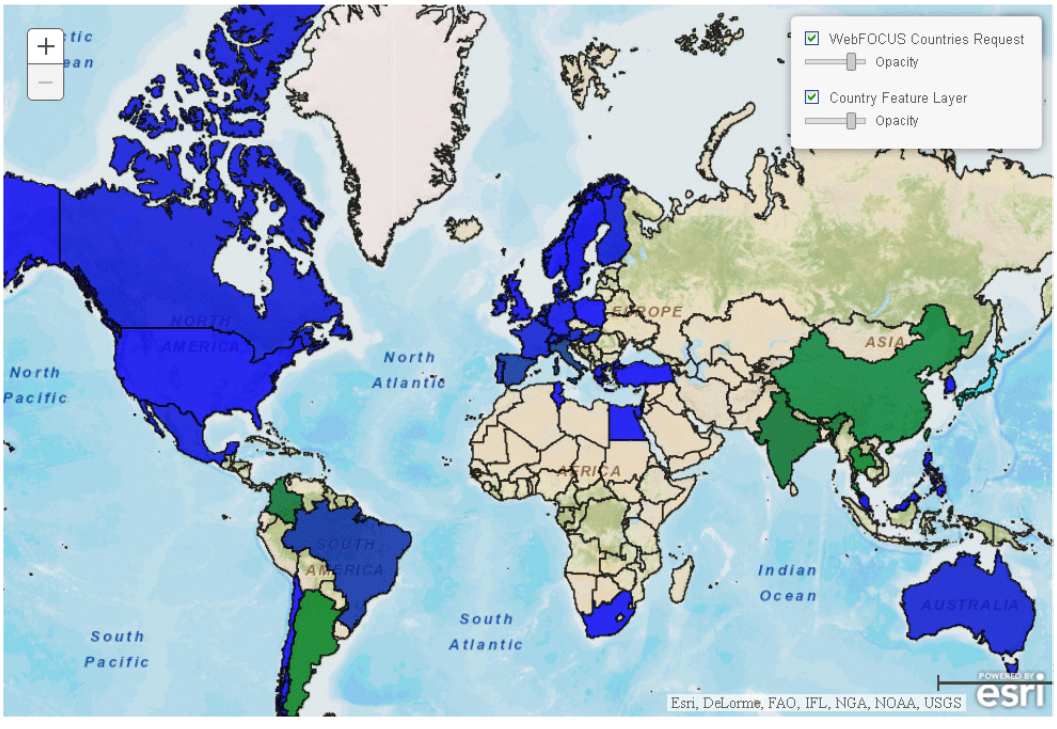


例　グラフエンジン構文による参照フィーチャレイヤのグラフへの追加

次のリクエストは、コロプレスマップに参照フィーチャレイヤを追加します。次のフィーチャレイヤは、World Countries エンドポイントを使用して国境を描画します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY  COUNTRY_NAME
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=color, $
*GRAPH_JS
"legend": {"visible": false},
"colorScale": {"colors":["blue", "green", "teal", "cyan"]},
"extensions": {
 "com.esri.map":
 {
  "baseLayer":{ "title": "Base Layer",
  "basemap": "streets"
 },
 "overlayLayers":
 [
  {
   "title": "WebFOCUS Countries Request",
 "layerType": "choropleth",
 "geometrySourceType": "esri",
 "geometryLocateField": ["Country"],
 "geometryDataField": "name",
 "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
 },


{
  "title": "Country Feature Layer",
  "layerType": "featurelayer",
  "geometrySourceType": "esri",
  "geometryLocateField": ["Country"],
  "geometryDataField": "name",
  "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
  }
 ],
 }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。[目次] コントロールには、各オーバーレイレイヤ表示のチェックボックスおよび不透明度スライダが表示されます。



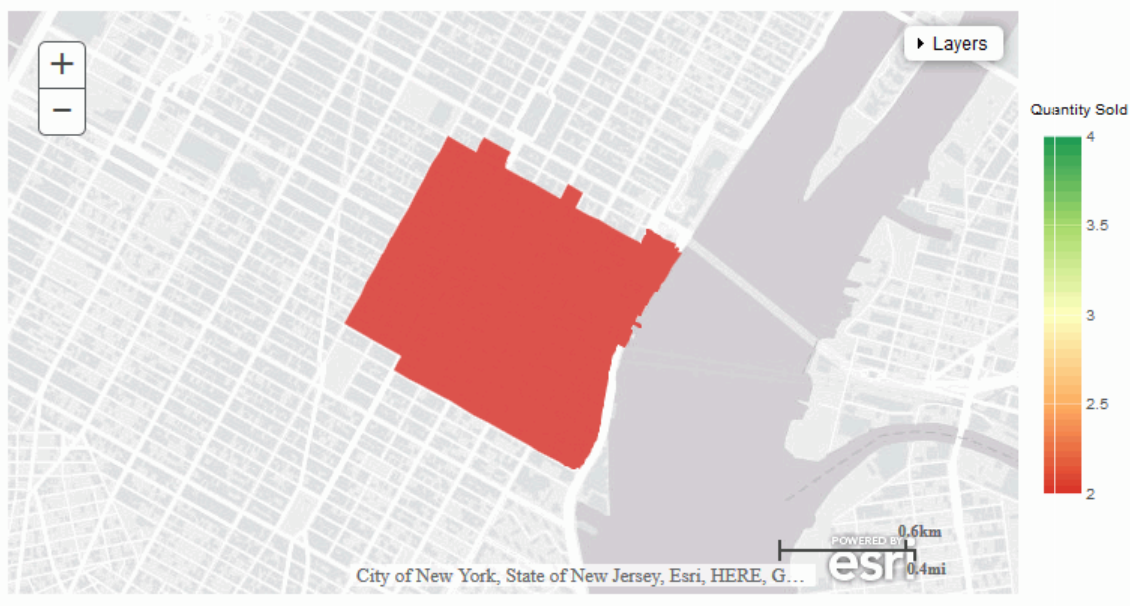
例　グラフエンジン構文による量子化しきい値を使用した狭いエリアの描画

次のリクエストでは、quantizationThreshold プロパティを 1000 に設定し、郵便番号 (ZIP code) 10116 のエリア描画を試みます。

```
GRAPH FILE WF_RETAIL_LITE
SUM QUANTITY_SOLD
 BY POSTAL_CODE
 WHERE COUNTRY_NAME EQ 'United States';
 WHERE POSTAL_CODE EQ '10016' ;
 ON GRAPH PCHOLD FORMAT JSCHART
 ON GRAPH SET LOOKGRAPH CHOROPLETH
 ON GRAPH SET STYLE *
 INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
 TYPE=REPORT, ORIENTATION=LANDSCAPE, CHART-LOOK=com.esri.map, $
 TYPE=DATA, COLUMN=N1, BUCKET=location, $
 TYPE=DATA, COLUMN=N2, BUCKET=color, $
 *GRAPH_JS_FINAL
 "extensions": {
 "com.esri.map": {
 "overlayLayers": [
 {
 "layerType": "choropleth",
 "quantizationThreshold": 1000,
 "geometryDataField": "name",
 "geometryLocateField": ["ZIP_CODE"],
 "title": "World Postal Codes (temporary = USA ZIP5)",
 "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/ArcGIS/rest/services/
USA_ZIP_Codes_2015/FeatureServer/0",
 "geometrySourceType": "esri"
 }
 ]
}
},
 "legend":
{ "visible": true }
*END
 ENDSTYLE
 END
```

出力結果は次のとおりです。地図上には何も描画されず、郵便番号は「不明 (unknown) 」と表示されます。

下図は、quantizationThreshold を 6000 に変更して出力した結果を示しています。郵便番号エリアが地図上に描画されています。



構文

グラフエンジン構文による緯度と経度のフィールドに基づくオーバーレイレイヤの定義

バブルマップは、緯度と経度のフィールドに基づいて描画されます。これらのフィールド値は数値です。数値でない場合は、可能な限り数値に変換されます。このタイプのリクエストにはエンドポイントは適用されません。必要な値はデータ ("geometrySourceType": "seriesdata") フィールドに入力します。

```
{
  "title": "string",
  "layerType": "bubble",
  "geometryXY": {
      "y": "lat",
      "x": "lng"
          },
  "geometrySourceType": "seriesdata"
  }
```

説明

`"title": "string"`

レイヤのタイトルです。

```
"y": "lat", "x": "lng"
```

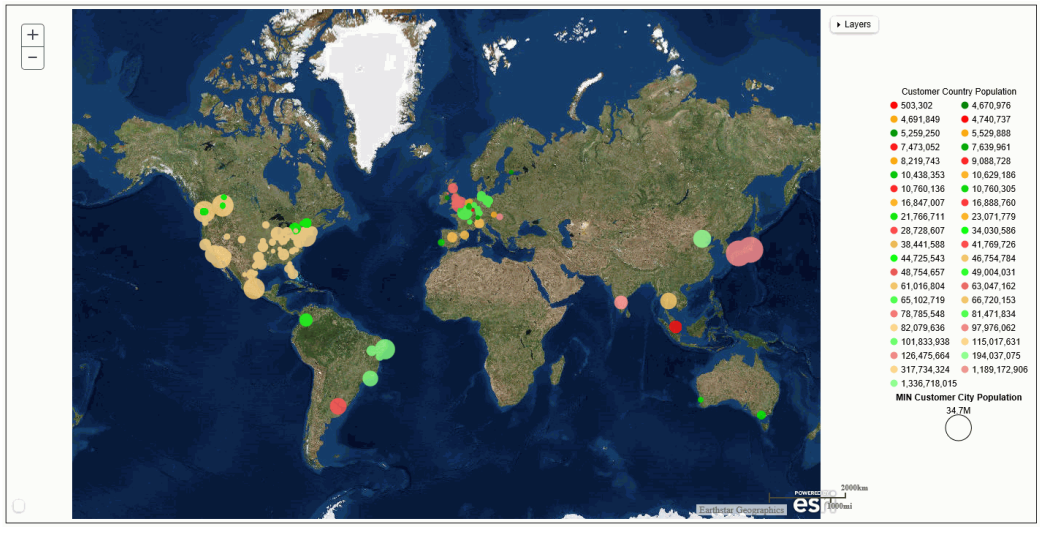
グラフ属性カテゴリへの接続に必要です。

例 グラフエンジン構文による緯度と経度のフィールドに基づく WebFOCUS オーバーレイレイヤの定義

次のリクエストは、satellite ベースマップおよび CITY_LONGITUDE と CITY_LATITUDE のソートフィールドを使用してバブルマップを生成します。シンボルは、COUNTRY_POPULATION ソートフィールドで色分けされ、MIN.CITY_POPULATION メジャーでサイズが決定されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY COUNTRY_POPULATION
BY CITY_LATITUDE
BY CITY_LONGITUDE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_POPULATION, BUCKET=color, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"bubbleMarker":{"maxSize":"5%"},
"extensions": {
    "com.esri.map": {
        "overlayLayers":             [
                {
                    "title": "Population",
                    "layerType": "bubble",
                    "geometryXY": {
                        "y": "lat",
                        "x": "lng"
                    },
                    "geometrySourceType": "seriesdata"
                }
            ],
        "baseLayer": {
            "basemap": "satellite"
        }    }
}

*END
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　　グラフエンジン構文による人口統計オーバーレイレイヤの定義

ESRI マップグラフに人口統計オーバーレイレイヤを追加することができます。レイヤタイプは、tile です。適切なエンドポイントにレイヤタイトルと URL を指定します。人口統計レイヤは、凡例の表示されないコロプレスマップとして表示されます。

```
{
 "title": "string",
 "layerType": "tile",
 "url": "url_to_endpoint"
 }
```

説明

`"title": "string"`

　レイヤのタイトルです。

`"url": "url_to_endpoint"`

　人口統計レイヤのエンドポイントの URL です。

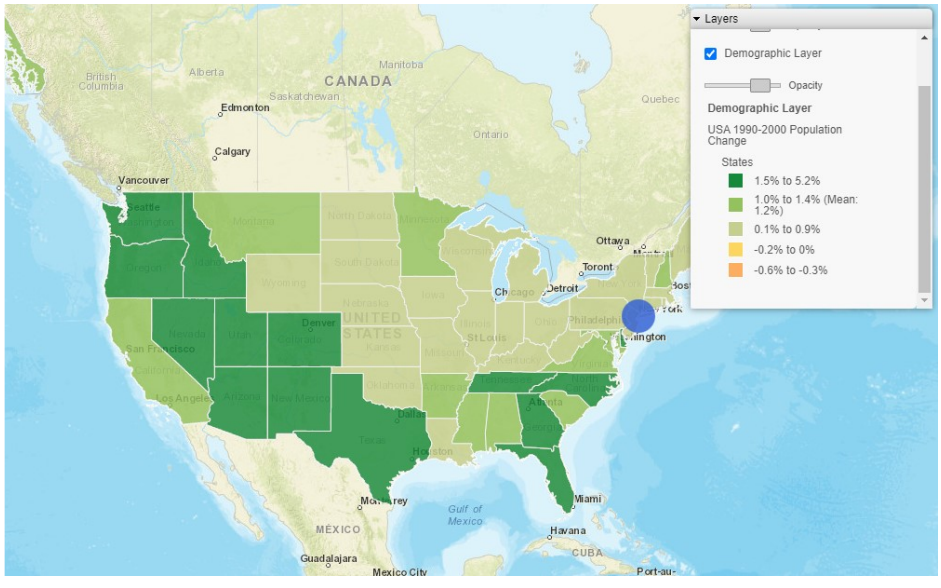
例　グラフエンジン構文による人口統計オーバーレイレイヤのグラフへの追加

次のリクエストは、バブルマップに USA Population Change 人口統計レイヤを追加します。この人口統計レイヤは米国に適用されるため、米国がベースマップの中央位置となり、拡大表示されます。詳細は、939 ページの 「 ESRI マップ出力のカスタマイズ 」 を参照してください。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY  CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker":{"maxSize":"5%"},
"extensions": {
 "com.esri.map": {
   "overlayLayers":             [
     {
     "title": "WebFOCUS Request Layer",
     "layerType": "bubble",
     "geometryXY": {
             "y": "lat",
             "x": "lng"
                 },
    "geometrySourceType": "seriesdata"
     },
  {
       "title": "Demographic Layer",
       "layerType": "tile",
       "url": "https://server.arcgisonline.com/ArcGIS/rest/services/Demographics/
USA_1990-2000_Population_Change/MapServer"
     },
   ],
```

```
 "baseLayer": {
   "basemap": "streets",
  "center": [-97, 40],
  "zoom": 4
        }
  }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。[目次] コントロールには、各オーバーレイレイヤ表示のチェックボックスおよび不透明度スライダが表示されます。



構文　グラフエンジン構文によるダイナミックマップサービスレイヤまたは参照フィーチャレイヤのスタイル設定

ダイナミックマップサービスレイヤまたは参照フィーチャレイヤのレンダラ作成に使用するスタイルおよび関数は、レイヤで使用可能なフィールドに smartMapping プロパティを追加することでカスタマイズすることができます。

これは、人口統計レイヤなどのタイルマップレイヤとは異なります。タイルマップレイヤは、そのままベースマップに保存 (キャッシュ) され、ベースマップに重ねて表示されます。ダイナミックマップサービスレイヤおよび参照フィーチャレイヤはどちらも、WebFOCUS リクエスト内で "layerType": "featurelayer" として分類されます。

ダイナミックマップサービスレイヤは、キャッシュされていないデータでかつ、ユーザがマップをナビゲートする際にマップサービスにより動的に生成されるデータを表示します。

次の構文は、スマートマッピングのパラメータを示します。

```
{
   "title": "string",
   "layerType": "featurelayer",
   "layerObjectType": "string",
   "smartMapping" :
   [
   array of fields or smartmapping objects,
    ],
  "geometrySourceType": "esri",
  "url": "url_to_endpoint"
   }
```

説明

`"title": "string"`

レイヤのタイトルです。

`"layerObjectType": "string"`

スタイル設定するレイヤのタイプを指定します。

- ダイナミックマップサービスレイヤの場合。

  `"layerObjectType": "esri/layers/ArcGISDynamicMapServiceLayer"`

- 参照フィーチャレイヤの場合。これがデフォルト設定の layerObjectType です。

  `"layerObjectType": "esri/layers/FeatureLayer"`

  参照フィーチャレイヤの URL には、全体のマップサービスではなく、機能マップサービスの特定のレイヤを指定する数値が末尾に追加されます。

`"smartMapping":`

次のいずれかの配列を指定することができます。

- マップサーバのレイヤのフィールド名文字列。
- スマートマッピングによって以下に定義された Promise を作成するために ESRI に渡されるオブジェクト。

  https://developers.arcgis.com/javascript/jsapi/esri.renderers.smartmapping-amd.html#createclassedcolorrenderer

マップサーバのレイヤは、同じ地理位置に対して異なる精度レベルを表します。たとえば、州、郡、郵便番号などのレイヤがあります。これらのレイヤそれぞれに対してフィールドまたはオブジェクトを指定することができます。レイヤフィールドまたはオブジェクトは、カンマ区切りで示されます。レイヤを省略するには、カンマのみ (,) でレイヤを示します。

smartMapping オブジェクトのパラメータの 1 つを、レンダラ作成関数の名前に指定することができます。指定した関数が smartMapping オブジェクトで呼び出されます。指定しなかった場合、デフォルト設定のレンダラ作成関数 (createClassedColorRenderer) が呼び出されます。レンダラ作成関数にはそれぞれ、スマートマッピング配列でカスタマイズが可能なプロパティが含まれています。これらのプロパティは、前述のリストのスマートマッピング URL で定義されます。smartMapping プロパティを指定した場合は、ESRI に渡されます。

各レイヤのレイヤ名とフィールド名は、マップサーバのエンドポイントで定義されます。

smartMapping プロパティを指定した場合、レイヤシンボルをカスタマイズすることはできません。

`"url": "url_to_endpoint"`

マップサーバのエンドポイントの URL です。

WebFOCUS により、レイヤタイプとベースマップがプロパティに追加されます。

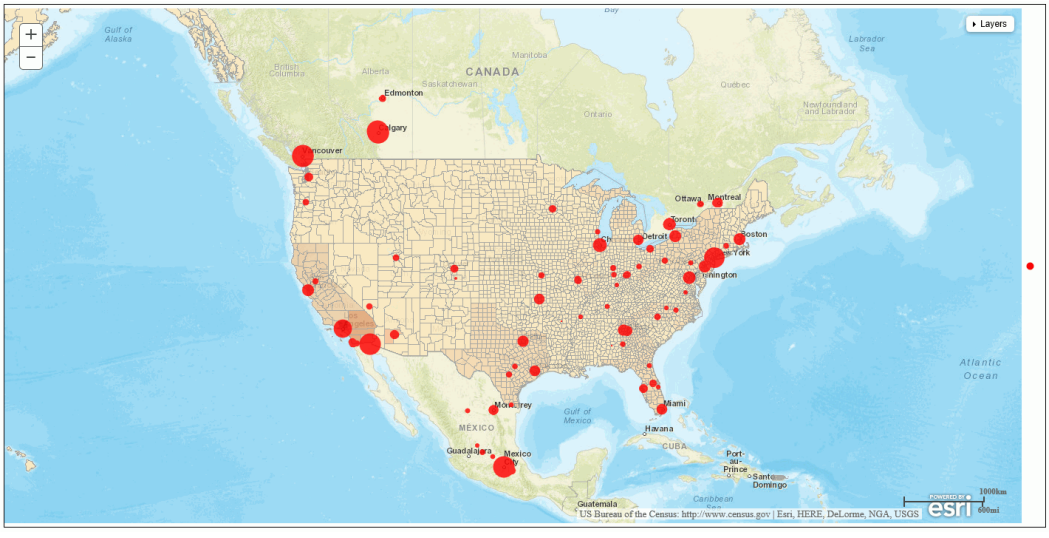
例　グラフエンジン構文によるダイナミックマップサービスレイヤのスタイル設定

次の例は、smartMapping 配列にエントリが含まれる各レイヤに対して、smartMapping の createClassedColorRenderer 関数 (デフォルト設定のレンダラ作成関数) を呼び出します。0 番目の配列位置 (国勢調査細分区ポイントを表し、最初のカンマを使用して省略される) にはエントリが含まれないため、このレイヤには何も表示されません。1 番目の位置 (国勢調査細分区グループ) および 2 番目の位置 (郡) については、フィールドエントリが "POP2007" に設定されています。1 番目の位置については、"smartMapping" の "classificationMethod" プロパティが "quantile" (WebFOCUS のデフォルト設定) に設定されています。2 番目の位置については、"classificationMethod" プロパティが "equal-interval" に設定されています。州を表す 3 番目の位置については、オブジェクトは指定されていません。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY  CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
```

```
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker": {"maxSize":"5%"},
"extensions": {
    "com.esri.map":
           {
             "baseLayer":{ "title" : "Base Layer",
             "basemap": "streets",
                       "center": [-98, 40],
                       "zoom": 4,
    },
             "overlayLayers":
[
               {
                   "title": "WebFOCUS Request Layer",
                   "layerType": "bubble",
                   "geometryXY": {
                       "y": "lat",
                       "x": "lng"},
                   "geometrySourceType": "seriesdata"
               },

              {
                    "title": "Feature Layer",
                    "layerType": "featurelayer",

                  "layerObjectType" : "esri/layers/ArcGISDynamicMapServiceLayer",
                   "smartMapping" :
                   [ ,
                    {"field":"POP2007","classificationMethod" : "quantile", },
                    {"field":"POP2007","classificationMethod" : "equal-
interval","numClasses":10}
                   ],

                    "geometrySourceType": "esri",
                    "url": "https://sampleserver6.arcgisonline.com/arcgis/rest/
services/Census/MapServer"
                 }
 ]
             }
                 }

*END
ENDSTYLE
END
```

出力結果は次のとおりです。



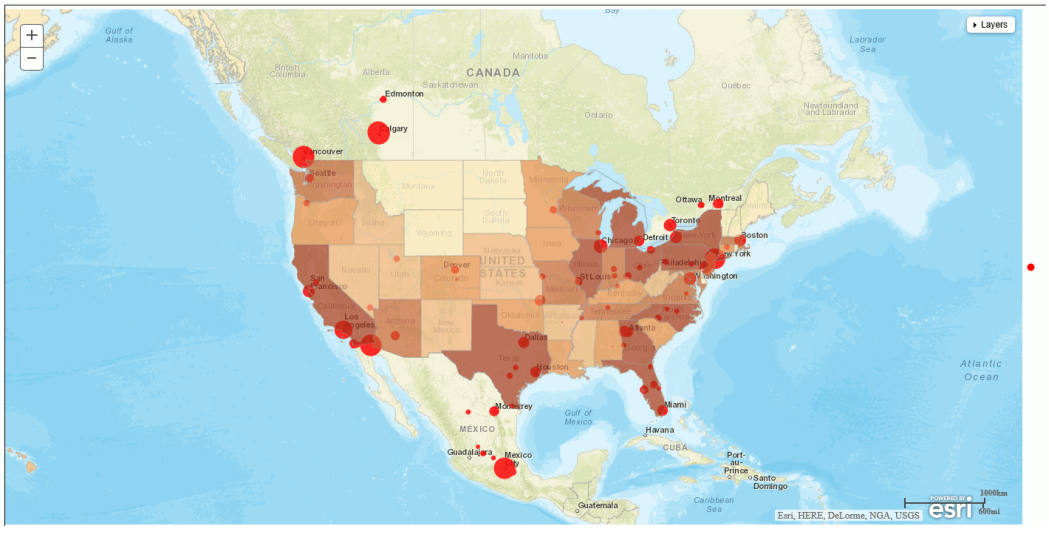
例　グラフエンジン構文による参照フィーチャレイヤのスタイル設定

次のリクエストでは、エンドポイントが州を表す参照フィーチャレイヤに設定されています。これはデフォルト設定のレイヤタイプであるため、layerObjectType プロパティは省略することができます。smartMapping 配列では、フィールド名が 1 つのみ指定され (POP2010)、デフォルト設定のスタイルが使用されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY  CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker": {"maxSize":"5%"},
"extensions": {
    "com.esri.map":
           {
      "baseLayer":{ "title" : "Base Layer",
              "basemap": "streets",
            "center": [-98, 40],
            "zoom": 4,
      },
   "overlayLayers":             [
    {
     "title": "WebFOCUS Request Layer",
     "layerType": "bubble",
     "geometryXY": {
      "y": "lat",
      "x": "lng"},
     "geometrySourceType": "seriesdata"
     },
```

```
      {
       "title": "Feature Layer",
       "layerType": "featurelayer",
"url":"https://services.arcgis.com/P3ePLMYs2RVChkJx/ArcGIS/rest/services/
USA_States_Generalized/FeatureServer/0",
      smartMapping : ["POP2010"],
      geometrySourceType : "esri"
     }
     ],
 }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフエンジン構文による区切りファイルのサポート

HTML5 フォーマットで ESRI マップを作成する際に、取得する位置の地理情報が記述されたロケーションファイルをリクエストで指定する必要があります。地図上に表示する各地域の名前と各地域の緯度と経度の値が格納されたカンマ区切り値 (.csv) ファイルまたは他の区切りファイルを指定することができます。

構文　区切り付きファイルのプロパティ定義

ジオメトリデータが格納された区切りファイルを使用する場合、その区切りファイルにアクセスしてジオメトリデータを取得するレイヤに csvOptions オブジェクトを追加することができます。ただし、すべてのオプションにはデフォルト値が設定されています。そのため、使用する区切りファイルがデフォルトオプションに適合している場合は、csvOptions オブジェクトを追加する必要はありません。

デフォルト設定の区切りファイルには、フィールド名 (タイトル) が記述された先頭行は含まれません。このファイルは、各行の末尾に行末文字「¥n」が追加されたカンマ区切りファイルです。このファイルには、各行に次の 3 つの値が含まれます。

1. 一重引用符 (') または二重引用符 (") で囲まれた領域の名前。このフィールドは、マップ拡張機能のプロパティでは「region」と表されます。
2. 領域に対応する経度。この値は、マップ拡張機能のプロパティでは「lng」と表されます。
3. 領域に対応する緯度。この値は、マップ拡張機能のプロパティでは「lat」と表されます。

使用するファイルがこれらの条件に適合しない場合は、レイヤのプロパティに csvOptions オブジェクトを使用することで、ファイルの構造を記述することができます。

```
"extensions": {
  "com.esri.map": {
    "overlayLayers":
    [
     {
        "csvOptions":
         {"fieldNames": ["fld1", "fld2", "fld3"],
          "fieldSep": "string",
          "lineSep": "string",
          "firstRowIsTitles": boolean,
          "removeSurroundingQuotes": boolean}
      }
    ],
                  }
                    }
```

説明

`"fieldNames": ["fld1", "fld2", "fld3"]`

ファイル内のフィールド名です。デフォルト値は ["region", "lng", "lat"] です。

`"fieldSep": "string"`

ファイル内のフィールド区切り文字です。デフォルト値はカンマ (,) です。この値は、任意の表示可能文字にすることができます。

`"lineSep": "string"`

ファイル内の行区切り文字列です。デフォルト値は、行末文字の「¥n」です。

`"firstRowIsTitles": boolean`

ファイルの先頭行にタイトルが含まれているかどうかを指定します。有効な値には、次のものがあります。

- true - ファイルの先頭行にタイトルが含まれていることを示します。先頭行にタイトルが含まれている場合、これらのタイトルは、fieldNames オブジェクトに列記されたフィールド名より優先されます。geometryLocateField 値を指定する際に、ファイル内のタイトルを使用します。
- false - ファイルの先頭行にタイトルが含まれていないことを示します。これがデフォルト値です。fieldNames オブジェクトを指定した場合は、対応するフィールド名を geometryLocateField 値として使用します。

`"removeSurroundingQuotes": boolean`

ファイル内で値を囲む引用符を削除するかどうかを指定します。有効な値には、次のものがあります。

- true - 値を囲む引用符を削除します。これがデフォルト値です。
- false - 値を囲む引用符がファイルに含まれていないことを示します。

以下は、csvOptions オブジェクトのデフォルト値を示しています。

```
"csvOptions":
     {
      "fieldNames": ["region", "lng", "lat"],
      "fieldSep": ",",
      "lineSep": "¥n",
      "firstRowIsTitles": false,
      "removeSurroundingQuotes": true
     }
```

次のレイヤプロパティを、csvOptions オブジェクトに対応する値に設定する必要があります。

```
"title": "string",
"layerType": "string",
"geometrySourceType": "csv",
"geometryLocateField": ["string"],
"url": "url_to_csv_file"
```

説明

`"title": "string"`

レイヤのタイトルです。

"layerType": "string"

レイヤのタイプです。有効な値には、次のものがあります。

- **choropleth** WebFOCUS コロプレスマップレイヤを表示します。
- **bubble** WebFOCUS バブルマップレイヤを表示します。

"geometryLocateField": ["string"]

地理位置情報フィールドのフィールド名です。使用するファイルにタイトルが含まれている場合、そのタイトルをフィールド名として使用する必要があります。ファイルにタイトルが含まれていない場合、csvOptions オブジェクトの fieldNames プロパティで指定されたフィールド名を使用する必要があります。いずれも指定されていない場合は、デフォルト値の "region" を使用します。

"url": "url_to_csv_file"

ファイルのパスを指定する URL です。

例 .csv ファイルによるバブルマップの生成

country.csv ファイルには、各国の経度と緯度の値が格納されています。

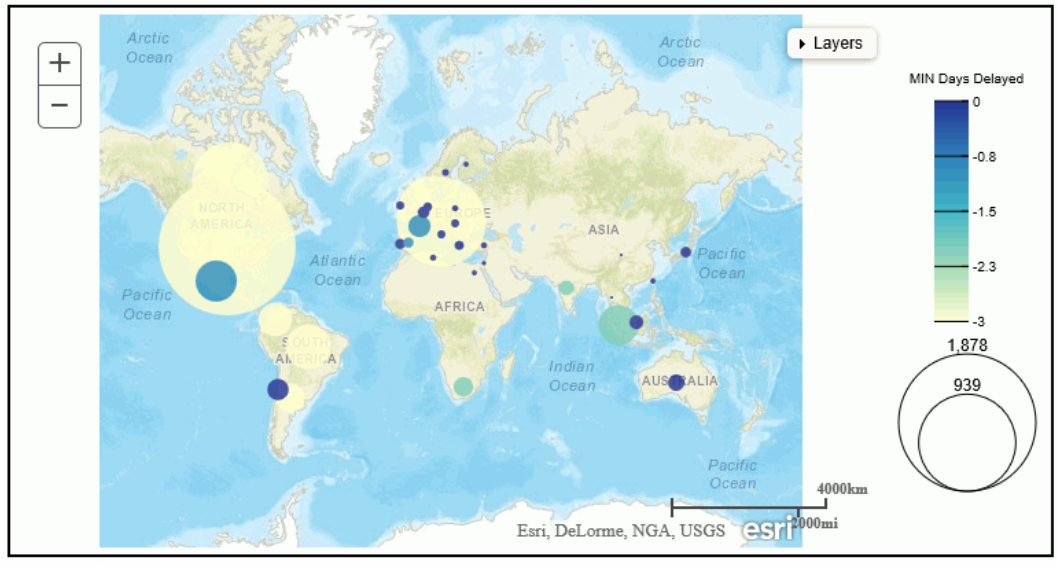
以下は、country.csv ファイルの最初の数行を示しています。

```
"Argentina",-64.00000000,-34.00000000
"Australia",133.00000000,-27.00000000
"Austria",13.33330000,47.33330000
"Belgium",4.00000000,50.83330000
"Brazil",-55.00000000,-10.00000000
"Canada",-95.00000000,60.00000000
"Chile",-71.00000000,-30.00000000
"China",105.00000000,35.00000000
"Colombia",-72.00000000,4.00000000
```

次のリクエストは、country.csv ファイルをジオメトリデータファイルとして使用し、国別の遅延日数を表示するバブルマップを生成します。このファイルはデフォルト構造に適合しているため、レイヤのプロパティに csvOptions オブジェクトは含まれていません。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
MIN.DAYSDELAYED
BY COUNTRY_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=country_name, BUCKET=location, $
TYPE=DATA, COLUMN=daysdelayed, BUCKET=size, $
TYPE=DATA, COLUMN=min.daysdelayed, BUCKET=color, $
*GRAPH_JS
"extensions": {
  "com.esri.map": {
    "baseLayer": {
      "title": "base title",
      "basemap": "streets"
                },
    "overlayLayers": [
      {
         "title": "csv default",
         "layerType": "bubble",
        "geometrySourceType": "csv",
        "geometryLocateField": ["region"],
         "url": "http://ecl.informationbuilders.com/jschart/country.csv"
      }
                        ],
                    }
                  }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



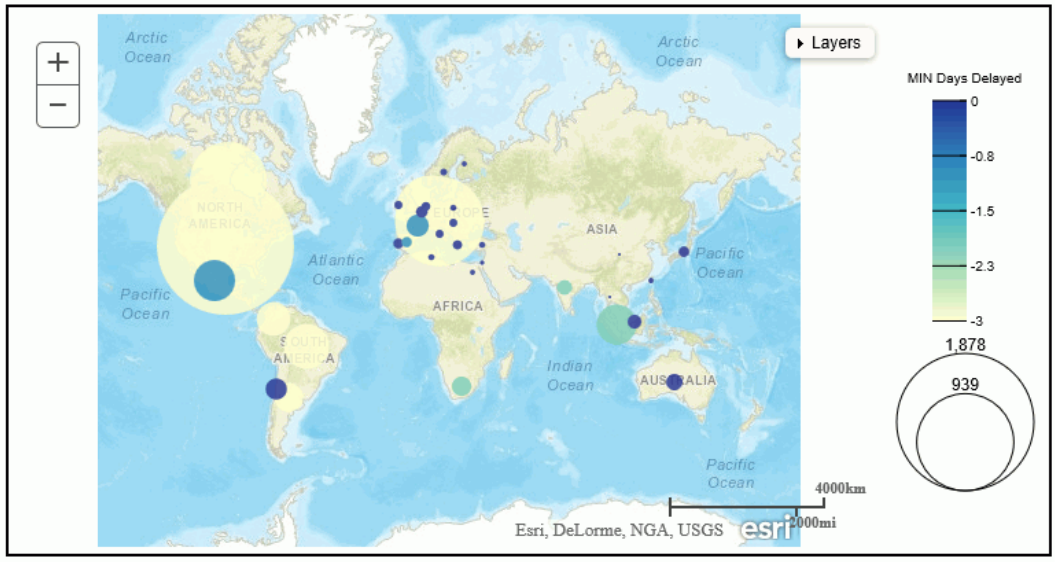
次の countryp.txt ファイルには、先頭行にタイトルが配置されています。また、区切り文字としてパイプ文字 (|) が使用され、値は二重引用符(") で囲まれています。以下は、このファイルの最初の数行を示しています。

```
"Country"|"lng"|"lat"
"Argentina"|"-64.0000000"|"-34.0000000"
"Australia"|"133.0000000"|"-27.0000000"
"Austria"|"13.3333000"|"47.3333000"
"Belgium"|"4.0000000"|"50.8333000"
"Brazil"|"-55.0000000"|"-10.0000000"
"Canada"|"-95.0000000"|"60.0000000"
```

次のリクエストでは、このファイルを使用します。このリクエストでは、csvOptions オブジェクトを使用して、デフォルト以外のオプションを定義します。このファイルにはタイトル行が配置されているため、fieldNames オブジェクトは含まれていません。このファイルのタイトル行で、地理位置情報フィールドとして "Country" が指定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
MIN.DAYSDELAYED
BY COUNTRY_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=country_name, BUCKET=location, $
TYPE=DATA, COLUMN=daysdelayed, BUCKET=size, $
TYPE=DATA, COLUMN=min.daysdelayed, BUCKET=color, $
*GRAPH_JS
"extensions": {
  "com.esri.map": {
    "baseLayer": {
      "title": "base title",
      "basemap": "streets"
                        },
    "overlayLayers": [
      {
        "title": "csv options",

        "csvOptions":
         {
            "fieldSep": "|",
            "firstRowIsTitles": true,
            "removeSurroundingQuotes": true
         } ,
        "layerType": "bubble",
        "geometrySourceType": "csv",
        "geometryLocateField": ["Country"],
         "url": "http://ecl.informationbuilders.com/jschart/countryp.txt"
       }
                                ],
                                   }
                            }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



グラフエンジン構文による Geojson ファイルのサポート

HTML5 フォーマットで ESRI マップを作成する際に、取得する位置ディメンションの地理情報が記述されたロケーションファイルをリクエストで指定する必要があります。マップ上に表示する各領域の名前と、各領域の経度と緯度の値が格納された geojson ファイルを指定することができます。

構文　Geojson ファイルのプロパティ定義

geojson ファイルを使用するオーバーレイレイヤに、次のプロパティを追加します。

```
"overlayLayers": [
 {
 "title": "string",
 "layerType": "choropleth",
 "geometrySourceType": "geojson",
 "geometryLocateField": ["string"],
 "geometryDataField": "name",
 "url": "http://url_to_geojson_file"
 }
 ],
```

説明

`"title": "string"`

レイヤ目次に表示するタイトルです。

```
"geometryLocateField": ["string"]
```

geojson ファイル内の位置フィールドの名前です。

```
"url": "http://url_to_geojson_file"
```

geojson ファイルのパスを指定する URL です。

例　HTML5 マップグラフでの Geojson ファイルの使用

sample_wfretail_regions.geojson ファイルには、アメリカ合衆国の各地域の境界を表す経度と緯度の値が格納されています。以下は、このファイルの先頭部分を示しています。

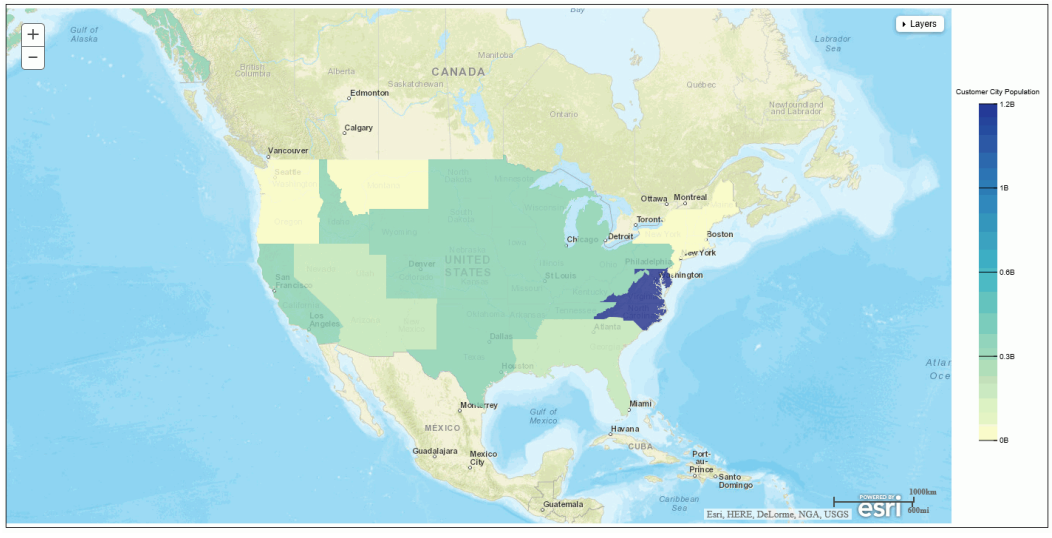
```
{
"type": "FeatureCollection",
"features":[
{ "type": "Feature","properties": { "region": "Central" },
"geometry": { "type": "Polygon","coordinates":
[[[-84.7570,45.7847],[-84.7288,45.7880],[-84.7212,45.7666],
[-84.5570,45.7084],[-84.4780,45.6566],[-84.4295,45.6448],
[-84.4150,45.6530],[-84.4285,45.6697],[-84.3768,45.6558],
[-84.3293,45.6643],[-84.2165,45.6349],[-84.1326,45.5666],
[-84.1123,45.5077],[-84.0749,45.4888],[-83.9362,45.4925],
[-83.8103,45.4204],[-83.7414,45.4037],[-83.7157,45.4144],
[-83.5988,45.3523],[-83.5374,45.3578],[-83.4953,45.3396],
[-83.4912,45.3590],[-83.4780,45.3453],[-83.4860,45.3308],
[-83.4636,45.3307],[-83.3830,45.2718],[-83.3913,45.2527],
[-83.4164,45.2566],[-83.4125,45.2402],[-83.3624,45.1652],
[-83.3165,45.1418],[-83.3199,45.1178],[-83.3008,45.0922],
[-83.3263,45.0850],[-83.3130,45.0849],[-83.3140,45.0546],
[-83.2616,45.0253],[-83.3390,45.0412],[-83.3779,45.0776],
[-83.4364,45.0558],[-83.4564,45.0230],[-83.4332,45.0045],
[-83.4647,45.0028],[-83.4280,44.9282],[-83.3153,44.8802],
[-83.3218,44.8553],[-83.2938,44.8083],[-83.2959,44.7413],
[-83.2717,44.7132],[-83.3145,44.6087],[-83.3072,44.5435],
[-83.3334,44.3372],[-83.3749,44.3267],[-83.4498,44.2507],
[-83.4431,44.2715],[-83.4788,44.2800],[-83.5357,44.2507],
[-83.5659,44.1634],
 ...
}
```

地域名が格納されているフィールドの名前は region です。次のリクエストは、geojson ファイルで定義された地域名に州名を割り当てた上で、このファイルをマップレイヤのジオメトリファイルとして使用します。

```
DEFINE FILE WF_RETAIL_LITE
STATE/A20=WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_NAME;
REGION/A20 WITH WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_NAME=
IF STATE EQ 'Illinois' OR 'Indiana' OR 'Iowa'
   OR 'Kentucky' OR 'Michigan' OR 'Minnesota' OR 'Missouri'
   OR 'Ohio' OR 'Wisconsin' OR 'Kansas' OR 'Nebraska' THEN 'Central'
ELSE IF STATE EQ 'Alabama' OR 'Florida' OR  'Georgia' OR 'Louisiana'
   OR 'Mississippi' OR 'North Carolina' OR 'South Carolina'
   OR 'Virginia' THEN 'SouthEast'
ELSE IF STATE EQ 'Arkansas' OR 'Tennessee' OR 'West Virginia'
   OR 'Texas' OR 'Oklahoma'
   OR 'New Mexico' OR 'Hawaii' THEN 'SouthWest'
ELSE IF STATE EQ 'Connecticut' OR 'Maine' OR 'Massachusetts'
   OR 'New Hampshire' OR 'Rhode Island' OR 'Vermont' THEN 'NorthEast'
ELSE IF STATE EQ 'New Jersey' OR 'New York' OR 'Pennsylvania'
   OR 'Delaware' OR 'Maryland' THEN 'MidEast'
ELSE IF STATE EQ 'Colorado' OR 'Idaho' OR 'Montana' OR 'North Dakota'
   OR 'South Dakota' OR 'Wyoming' OR 'Alaska' THEN 'NorthWest'
ELSE IF STATE EQ 'Arizona' OR 'California' OR 'Nevada' OR 'Oregon'
   OR 'Utah' OR 'Washington' THEN 'West'
ELSE 'Unknown';
END
```

```
GRAPH FILE WF_RETAIL_LITE
SUM CITY_POPULATION
BY REGION
WHERE COUNTRY_NAME EQ 'United States'
WHERE REGION NE 'Unknown'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=region, BUCKET=location, $
TYPE=DATA, COLUMN=city_population, BUCKET=color, $
*GRAPH_JS
"extensions": {
 "com.esri.map": {
 "overlayLayers": [
 {
 "title": "Geojson",
 "layerType": "choropleth",
 "geometrySourceType": "geojson",
 "geometryLocateField": ["region"],
 "geometryDataField": "name",
 "url": "http://ecl.informationbuilders.com/jschart/sample_wfretail_regions.geojson"
 },
 ],
 "baseLayer": {
 "basemap": "streets",
"center": [-98, 40],
"zoom": 4                  }
 }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



ESRI
マップ出力のカスタマイズ

グラフリクエストで ESRI プロパティを指定し、マップの初期表示およびマップ出力で使用可能なコントロールをカスタマイズすることができます。

構文　　参照フィーチャレイヤシンボルのフォーマット設定

参照フィーチャレイヤは、マップグラフ上にシンボルを配置します。WebFOCUS は、位置の外枠にこれらのレイヤを使用します。WebFOCUS で使用されるシンボルは、単純な塗りつぶしのシンボルと単純な線のシンボルで構成されます。ポイントレイヤを使用する場合、単純なマーカーシンボルを使用することができます。フィーチャレイヤにシンボルオブジェクトを追加してシンボルのプロパティを変更することもできます。

たとえば、WebFOCUS が位置の外枠に使用するデフォルト設定のシンボルは、黒色、不透明の単純な外枠線、および塗りつぶしなし、スタイル指定なしで定義された単純な塗りつぶしのシンボルで構成されています。シンボルの定義を編集して、塗りつぶしのスタイルおよび色を追加したり、外枠の色、スタイル、幅を変更したりできます。

単純な塗りつぶしシンボルの定義は以下を参照してください。

https://developers.arcgis.com/javascript/jsapi/simplefillsymbol-amd.html

単純な線シンボルの定義は以下を参照してください。

https://developers.arcgis.com/javascript/jsapi/simplelinesymbol-amd.html

単純なマーカーシンボルの定義は以下を参照してください。

https://developers.arcgis.com/javascript/jsapi/simplemarkersymbol-amd.html

デフォルト設定では、WebFOCUS のシンボルは以下のシンボルプロパティで構成されています。

```
"symbol": {
          "type": "esriSFS",
            "style": "esriSFSstring",
            "color": [r, g, b, t],
          "outline": {
            "type": "esriSLS",
            "style": "esriSLSstring",
            "color": [r, g, b, t],
            "width": number                        }
        },
```

説明

`"type": "esriSFS"`

単純な塗りつぶしを使用するシンボルを指定します。単純な塗りつぶしシンボルの定義およびオプションについての詳細は、

https://developers.arcgis.com/javascript/jsapi/simplefillsymbol-amd.html を参照してください。

`"style" : "esriSFSstring"`

塗りつぶしスタイルを定義します。WebFOCUS のデフォルト設定は、塗りつぶしなし ("esriSFSNull") です。

`"color": [r, g, b, t]`

塗りつぶしの色を RGBA フォーマットで定義します。塗りつぶしの色は、赤、緑、青の 3 原色の強度と透明度の値を指定して定義します。3 原色の値にはそれぞれ 0 から 255 までの数値を指定します。透明度の値については、ゼロ (0) が完全な透明を表し、255 が完全な不透明を表します。WebFOCUS のデフォルト値は [0,0,0,0] です。

`"type": "esriSLS"`

単純な線シンボルを使用する外枠を指定します。単純な線シンボルの定義およびオプションについての詳細は、

https://developers.arcgis.com/javascript/jsapi/simplelinesymbol-amd.html を参照してください。

`"style": "esriSLSstring"`

線のスタイルを定義します。WebFOCUS のデフォルト設定は、実線 ("esriSLSSolid") です。

`"color": [r, g, b, t]`

線の色を RGBA フォーマットで定義します。塗りつぶしの色は、赤、緑、青の 3 原色の強度と透明度の値を指定して定義します。3 原色の値にはそれぞれ 0 から 255 までの数値を指定します。透明度の値については、ゼロ (0) が完全な透明を表し、255 が完全な不透明を表します。WebFOCUS のデフォルト値は [0,0,0,255] です。

`"width": number`

ピクセル単位で指定する線の幅です。WebFOCUS のデフォルト値は 1 です。

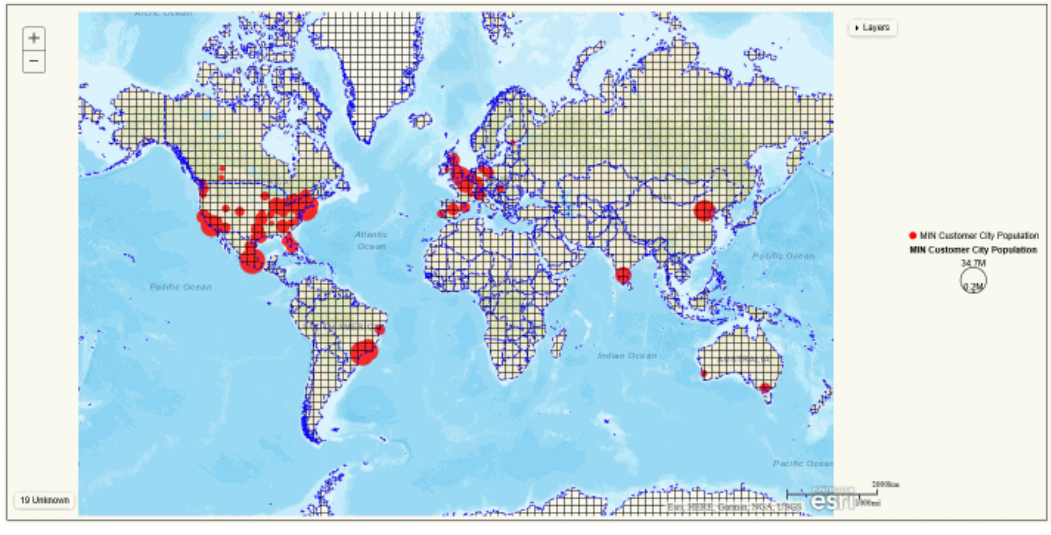
例　Reporting Server 構文による参照フィーチャレイヤシンボルのフォーマット設定

次のリクエストは、シンボルの塗りつぶしスタイルをクロスハッチに、外枠を青色の破線に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY CITY_NAME
WHERE CITY_POPULATION NE  ' ' OR MISSING
WHERE COUNTRY_NAME NE  'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS_FINAL
"bubbleMarker": {"maxSize":"5%", "color": "red"},

"extensions": {
  "com.esri.map": {
      "overlayLayers":
     [
        {
         "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "CITY_NAME"
                                            }
                                     }
          },
       {"ibiAddLayer": "World_Countries",
           "symbol": {
           "type": "esriSFS",
             "style": "esriSFSCross",
              "color": [0, 0, 0, 255],
                  "outline": {
                   "type": "esriSLS",
                  "style": "esriSLSDash",
                         "color": [0,0,255, 255],
                  "width": 1
                            }
                       }
       }
      ],
```

```
    "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "streets"
                }
            ]
          }
                        }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



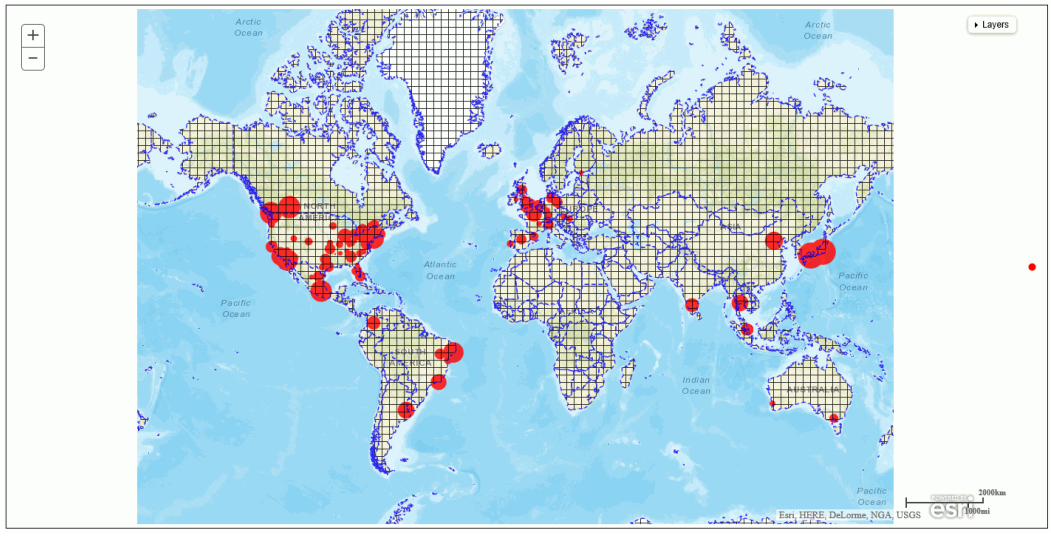
例　グラフエンジン構文による参照フィーチャレイヤシンボルのフォーマット設定

次のリクエストは、シンボルの塗りつぶしスタイルをクロスハッチに、外枠を青色の破線に変更します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY  CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker": {"maxSize":"5%"},
"extensions": {
    "com.esri.map":
           {
             "baseLayer":{ "title": "Base Layer",
         "basemap": "streets"
    },
            "overlayLayers":             [
               {
                   "title": "WebFOCUS Request Layer",
                   "layerType": "bubble",
                   "geometryXY": {
                       "y": "lat",
                       "x": "lng"},
                   "geometrySourceType": "seriesdata"
               },
```

```
                {
                      "title": "Country Feature Layer",
                      "layerType": "featurelayer",
                          "symbol": {
                               "type": "esriSFS",
                               "style": "esriSFSCross",
                               "color": [0, 0, 0, 255],
                               "outline": {
                                  "type": "esriSLS",
                                  "style": "esriSLSDash",
                                  "color": [0,0,255, 255],
                                  "width": 1
                                          }
                                    },
                      "geometrySourceType": "esri",
                      "geometryLocateField": ["Country"],
                      "geometryDataField": "name",
                      "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/
services/World_Countries_(Generalized)/FeatureServer/0"
                   }

              ],

               }
                    }

*END
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　レイヤオプションの追加

ESRI レイヤオプションを追加するには、レイヤ定義に options オブジェクトを追加します。WebFOCUS では追加したオプションを確認せずに ESRI に直接渡します。

```
"options": {"layer_option":value, ...}
```

説明

`"layer_option"`

ESRI API で使用可能なレイヤオプションです。これらは ESRI に渡されて処理されます。

`value`

レイヤオプションの有効な値です。

例　Reporting Server 構文によるレイヤ不透明度の初期設定

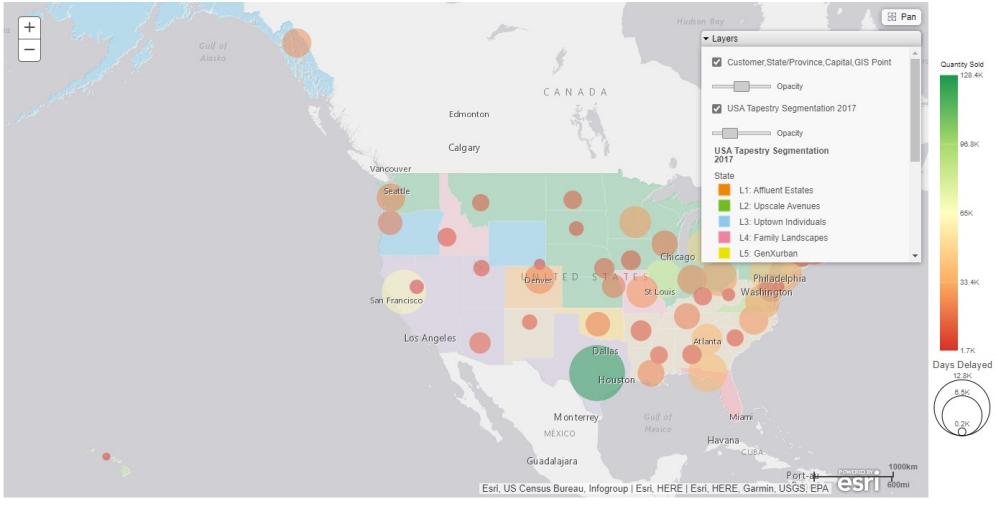
次のリクエストは、各オーバーレイレイヤに不透明度の初期値を追加します。不透明度は、0.0 から 1 までの数字で、レイヤ不透明度のパーセントを示します。0 は 100% 透明を表し、1 は 100% 不透明を表します。

注意：1 より小さい値の場合、リーディングゼロと小数点 (0.) が必要です。

次のリクエストでは、プロポーショナルシンボルが 50% 不透明、人口統計レイヤが 20% 不透明に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_CAPITAL_POINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=color, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                     "map_by_field":
"WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_CAPITAL_POINT"
                                                  }
                                        },
                  "options": {"opacity": 0.5}
         },
         {"ibiAddLayer": "USA_Tapestry_Segmentation_2017",
           "options": {"opacity": 0.2}
       }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                          }
                   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。[目次] コントロール内の不透明度スライダは、リクエスト内の不透明度の値を反映しています。



例　グラフエンジン構文によるレイヤ不透明度の初期設定

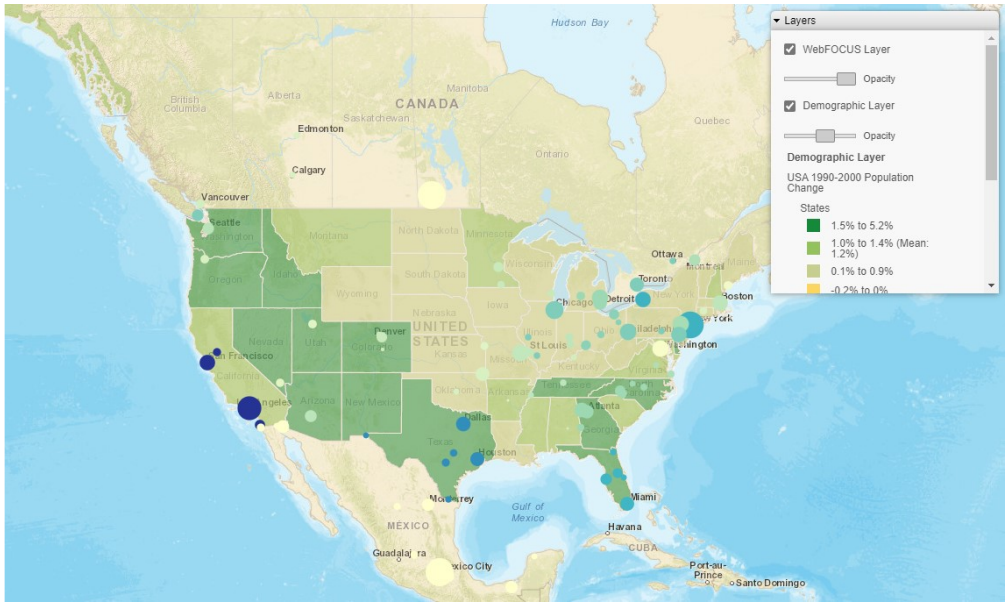
次のリクエストは、各オーバーレイレイヤに不透明度の初期値を追加します。不透明度は、0 から 1 までの数字で、レイヤ不透明度のパーセントを示します。0 は 100% 透明を表し、1 は 100% 不透明を表します。

次のリクエストでは、プロポーショナルシンボルが 100% 不透明、人口統計レイヤが 60% 不透明、フィーチャレイヤが 20% 不透明に設定されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
AVE.STATE_PROV_POPULATION
BY  CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=AVE.STATE_PROV_POPULATION, BUCKET=color, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker": {"maxSize":"5%"},
extensions:
{
 "com.esri.map":
  {
   "baseLayer":
  {"title" : "Base Layer",
   "basemap": "streets",
   "center": [-98, 40],
   "zoom": 4
   },
```

```
   "overlayLayers":
   [
   {
    "title": "WebFOCUS Layer",
    "layerType": "bubble",
    "geometryXY": {
         "y": "lat",
         "x": "lng"},
    "geometrySourceType": "seriesdata",
    "options": {"opacity":1}
    },
    {
    "title": "Demographic Layer",
    "layerType": "tile",
    "url": "https://server.arcgisonline.com/ArcGIS/rest/services/Demographics/
USA_1990-2000_Population_Change/MapServer",
    "options": {"opacity":.6}
    },
    {
    "title": "Country Feature Layer",
    "layerType": "featurelayer",
    "geometrySourceType": "esri",
    "geometryLocateField": ["Country"],
    "geometryDataField": "name",
    "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0",
    "options": {"opacity":.2}
    }
    ]
  }
 }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。[目次] コントロール内の不透明度スライダは、リクエスト内の不透明度の値を反映しています。



構文　ESRI マップグラフの初期中央位置およびズームのプロパティ設定

ベースレイヤの定義に追加できる便利なプロパティとして、初期中央位置、最大ズームレベル、最小ズームレベル、初期ズームレベルがあります。これらを指定しない場合は、リクエスト出力のジオメトリに基づいて WebFOCUS がこれらの値を計算します。

```
"baseLayer": {title: "layertitle",
          "basemap": "string",
          "center": [longitude,latitude],
          "minZoom": number,
          "maxZoom": number,
          "zoom": number        }
```

説明

`"title": "layertitle"`

レイヤのタイトルです (オプション)。

`"basemap": "string"`

サポートされるベースマップの 1 つの名前です。

注意：Reporting Server 構文では、ベースマップの名前は、baseMapInfo オブジェクトから取得されます。baseLayer オブジェクトで指定されたベースマップの名前は無視されます。

`"center": [longitude,latitude]`

最初に表示されたときのベースマップの中央位置を表す緯度と経度です。

`"minZoom": number`

グラフの最小ズームレベルです。

`"maxZoom": number`

グラフの最大ズームレベルです。

`"zoom": number`

最初に表示されたときのグラフの 19 段階のズームレベルを指定する 0 (ゼロ) から 19 までの整数です。

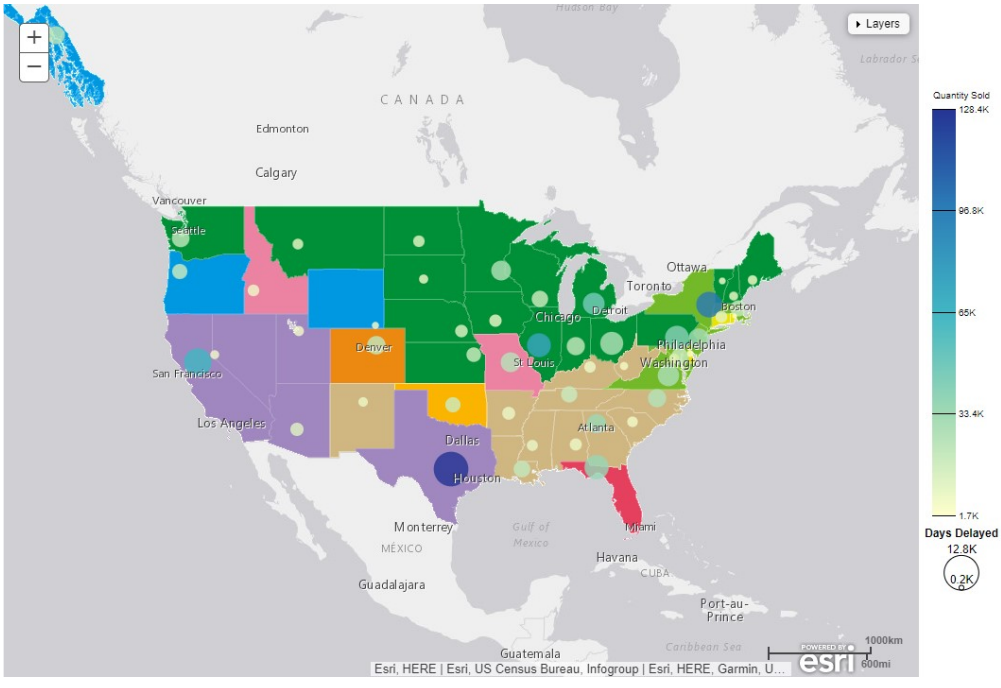
例

Reporting Server 構文による中央位置および初期ズームレベルの追加

次のリクエストは、ベースマップの中央位置を米国に設定し、初期ズームレベルを 4 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_CAPITAL_POINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=color, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"bubbleMarker": {"maxSize": "5%"},
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                     "map_by_field":
"WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.STATE_PROV_CAPITAL_POINT"
                                                  }
                                      }
          },
         {"ibiAddLayer": "USA_Tapestry_Segmentation_2017"
           }
      ],
   "baseLayer": {
       "basemap": "gray",
       "center": [-97, 40],
       "zoom": 4
              },
       "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
           ]
         }
                        }
                   }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

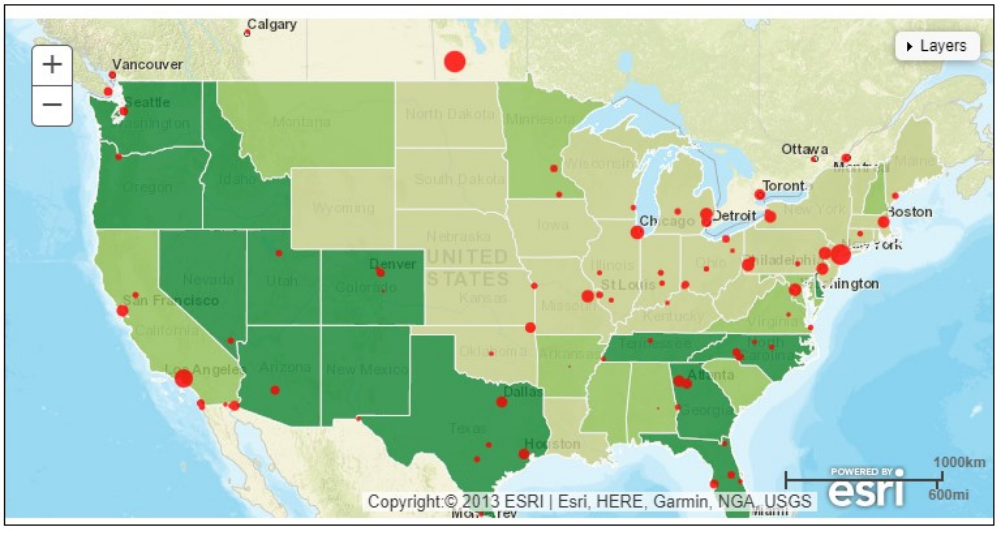


例　グラフエンジン構文による中央位置および初期ズームレベルの追加

次のリクエストは、ベースマップの中央位置を米国に設定し、初期ズームレベルを 4 に設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker":{"maxSize":"5%"},
"extensions": {
"com.esri.map": {
"overlayLayers": [
{
"title": "WebFOCUS Request Layer",
"layerType": "bubble",
"geometryXY": {
"y": "lat",
"x": "lng"
              },
"geometrySourceType": "seriesdata"
},
{
"title": "Demographic Layer",
"layerType": "tile",
"url": "https://server.arcgisonline.com/ArcGIS/rest/services/Demographics/
USA_1990-2000_Population_Change/MapServer"
}
],
"baseLayer": {
  "basemap": "streets",
  "center": [-97, 40],
  "zoom": 4
            }
}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



構文　ESRI マップグラフへのベースマップ情報の追加

グラフエンジン構文を使用して ESRI グラフリクエストに baseMapInfo オブジェクトを追加することで、グラフの背景として使用するカスタムマップイメージの配列を指定することができます。Reporting Server 構文を使用する場合は、地理構成ファイルにカスタムベースマップを追加することができます。詳細は、893 ページの 「 TIBCO WebFOCUS Reporting Server の地理構成ファイル 」 を参照してください。

グラフエンジン構文および Reporting Server 構文のどちらを使用する場合でも、使用可能なすべてのベースマップのサムネールを表示するベースマップコントロールをグラフ出力に配置し、生成されたグラフ出力上でベースマップを切り替えることができます。また、ベースマップコントロールに表示されるベースマップサムネールのタイトルを変更することもできます。

```
"baseMapInfo": {
 "drawBasemapControl": boolean,
 "showArcGISBasemaps": boolean,
 "customBaseMaps": [
 {
 "name": "string",
 "title": "string",
 "url": "url_to_basemap",
 "thumbnailUrl": "url_to_thumbnail"
 }
 ]
}
```

説明

`"drawBasemapControl" : boolean`

グラフ出力にベースマップコントロールを描画するかどうかを指定します。有効な値には、次のものがあります。

- true　ベースマップコントロールを描画します。
- false　ベースマップコントロールを描画しません。これがデフォルト値です。

`"showArcGISBasemaps" : boolean`

ベースマップコントロールに ArcGIS Online ベースマップを表示するかどうかを指定します。有効な値には、次のものがあります。

- true　ベースマップコントロールに ArcGIS Online ベースマップを表示します。これがデフォルト値です。
- false　ベースマップコントロールに ArcGIS Online ベースマップを表示しません。

`"name" : "string"`

Reporting Server 構文では、このプロパティは無視されます。グラフエンジン構文で "baselayer": "basemap" プロパティとして使用可能なベースマップの名前です。グラフ出力では、これがベースマップとして選択されます。

`"title": "string"`

ベースマップコントロールでベースマップサムネールの下側に表示するタイトルです。

`"url": "url_to_basemap"`

グラフエンジン構文にのみ適用されます。ベースマップのエンドポイントの URL です。

`"thumbnailUrl": "url_to_thumbnail"`

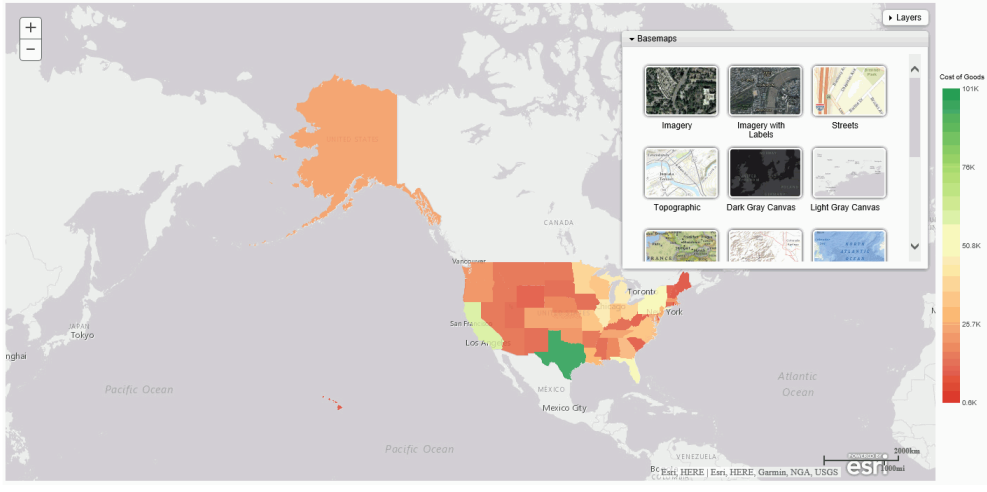
グラフエンジン構文にのみ適用されます。ベースマップのサムネールの URL です。

例 Reporting Server 構文による ESRI マップグラフへのベースマップコントロールの追加

次のリクエストは、グラフにベースマップコントロールを追加します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-metadata": {"map_by_field": "STATE_PROV_NAME"}
                                  }
        }
        ],
      "baseMapInfo":  {
          "drawBasemapControl" : true,
          "showArcGISBasemaps" : true,
          "customBaseMaps":
          [
              {"ibiBaseLayer": "gray" }
           ]
                                 }
                            }
                       }
*END
ENDSTYLE
END
```

下図は、出力結果を示しています。この図では、ベースマップコントロールが展開されています。



グラフの背景を変更するには、ベースマップコントロールでサムネールのいずれかをクリックします。

ベースマップコントロールを閉じるには、ヘッダの任意の位置をクリックします。

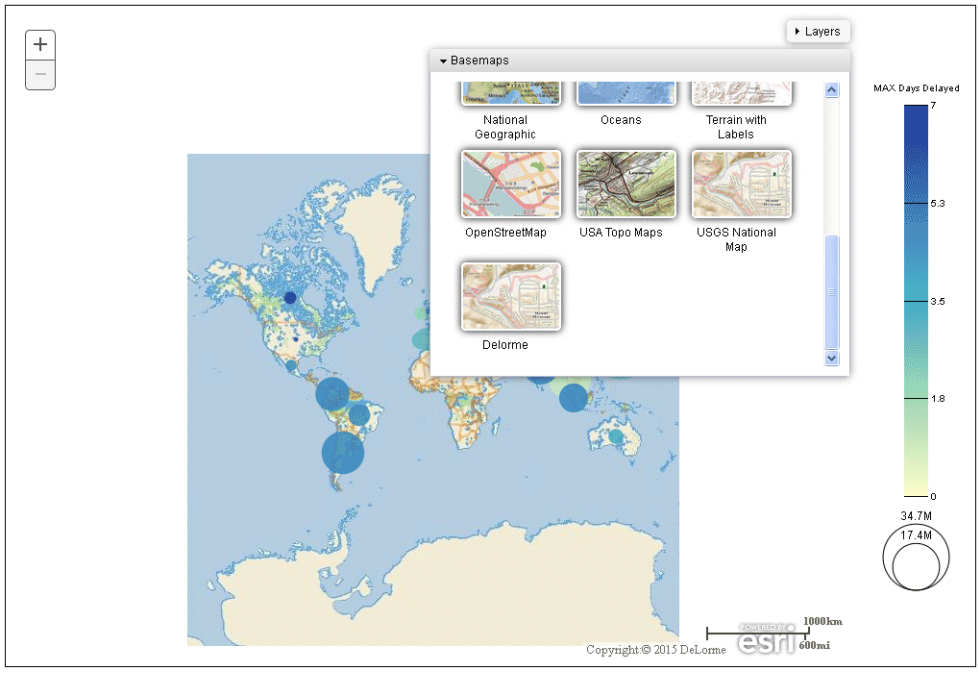
例　グラフエンジン構文による ESRI マップグラフへのベースマッププロパティの追加

次のリクエストは、DeLorme ベースマップおよびベースマップコントロールをグラフに追加します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION MAX.DAYSDELAYED
BY COUNTRY_NAME
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
TYPE=DATA, COLUMN=MAX.DAYSDELAYED, BUCKET=color, $
*GRAPH_JS
"bubbleMarker": {"maxSize": "10%" },
"extensions": {
"com.esri.map": {
"overlayLayers": [
{
"title": "Population",
"layerType": "bubble",
"geometrySourceType": "esri",
"geometryLocateField": ["Country"],
"geometryDataField": "name",
"url":
"https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
}
],
```

```
"baseMapInfo" : {
"drawBasemapControl" : true,
"showArcGISBasemaps" : true,
"customBaseMaps" : [
{
"name":"Delorme",
"title":"Delorme",
"url":"https://services.arcgisonline.com/ArcGIS/rest/services/Specialty/
DeLorme_World_Base_Map/MapServer",
"thumbnailUrl":"https://www.arcgis.com/sharing/rest/content/items/
6d9fa6d159ae4a1f80b9e296ed300767/info/thumbnail/national_map.jpg"
}
                    ]
                     },

"baseLayer": {
"basemap": "Delorme"
}
}
}
*END
ENDSTYLE
END
```

下図は、出力結果を示しています。この図では、ベースマップコントロールが展開されています。



グラフの背景を変更するには、ベースマップコントロールでサムネールのいずれかをクリックします。

ベースマップコントロールを閉じるには、ヘッダの任意の位置をクリックします。

構文　ESRI マップグラフ上のスケールバーのカスタマイズ

デフォルト設定では、マップグラフの右下に 2 つの単位でスケールバーを配置します。プロパティを追加して、別の位置にスケールバーを配置したり、メートル単位または英語圏の単位のいずれかを表示したりできます。

次のコードは、グラフ上にスケールバーを配置する方法および表示する単位のタイプを定義する方法を示します。

```
"extensions": {
    "com.esri.map":{
      "scalebar": {"scalebarUnit": "string",
                  "attachTo": "string" },
                    }
          }
```

説明

`"scalebarUnit": "string"`

表示する単位を指定します。有効な値には、次のものがあります。

- **english**　マイルなど英語圏の単位を表示します。
- **metric**　キロメートルなどメートル単位を表示します。
- **dual**　両方の単位タイプを表示します。

`"attachTo" : "string"`

グラフ上のどこにスケールバーを配置するかを指定します。有効な値には、次のものがあります。

- **top-right**
- **bottom-right**
- **top-center**
- **bottom-center**
- **bottom-left**
- **top-left**

その他のスケールバープロパティについては、以下のスケールバーの API 定義を参照してください。

https://developers.arcgis.com/javascript/jsapi/scalebar-amd.html

例

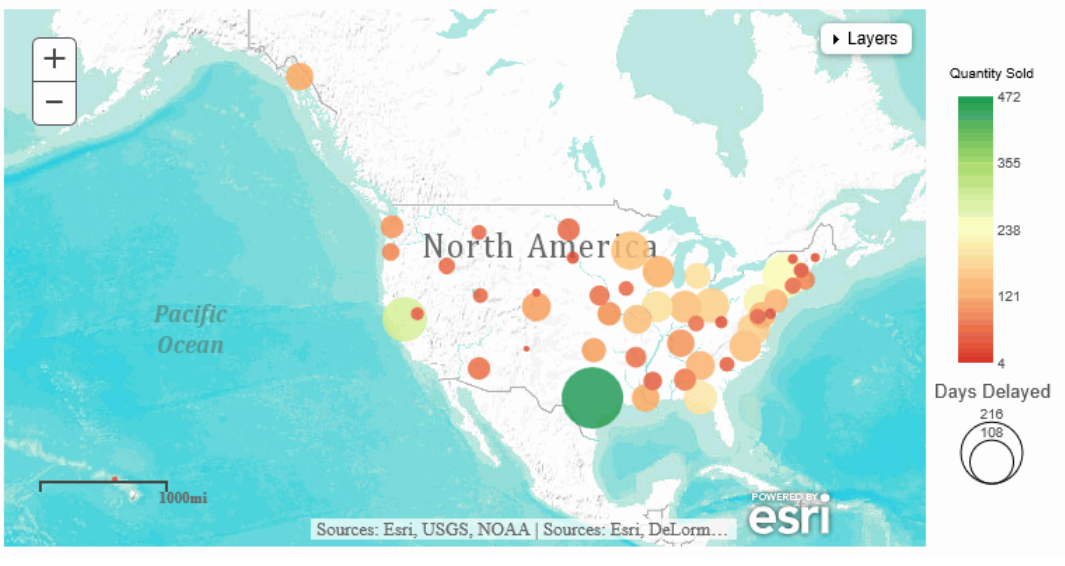
Reporting Server 構文による ESRI マップグラフ上のスケールバーのカスタマイズ

次のリクエストは、スケールバーをマップの左下に移動し、英語圏の単位のみを表示します。

```
DEFINE FILE WF_RETAIL_LITE
GEOPOINT/A200 = GIS_POINT('4326', STATE_PROV_CAPITAL_LONGITUDE,
STATE_PROV_CAPITAL_LATITUDE);
END
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY GEOPOINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=color, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
   "scalebar" : {"scalebarUnit": "english",
    "attachTo" : "bottom-left"
                },
```

```
        "overlayLayers":
       [
         {
          "ibiDataLayer": {
                   "map-geometry": {
                        "map_by_field": "GEOPOINT"
                                              }
                                    }
         }
       ],
       "baseMapInfo":
          {
           "customBaseMaps":
            [
                {
                 "ibiBaseLayer": "terrain"
                 }
              ]
           }
                          }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



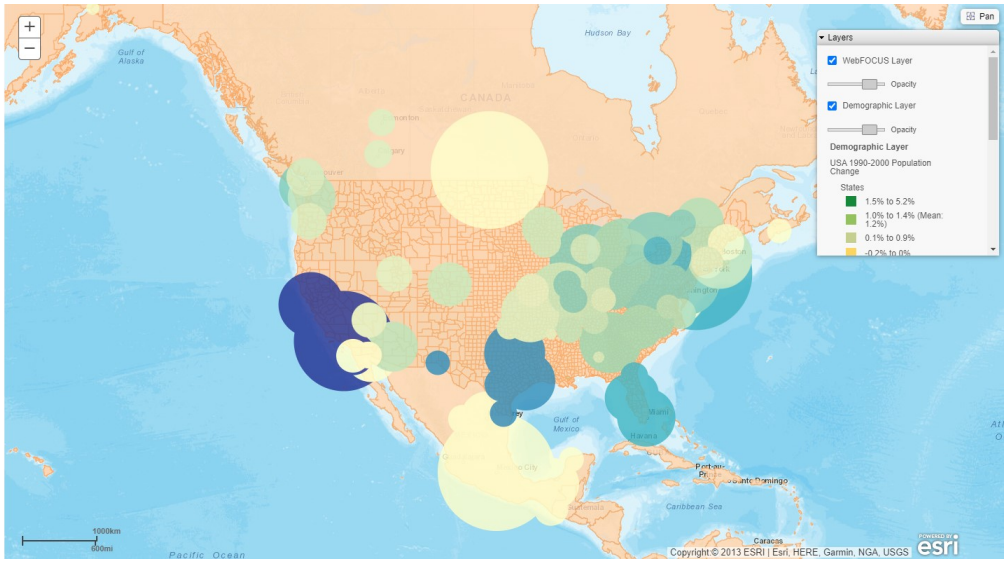
例　グラフエンジン構文による ESRI マップグラフ上のスケールバーのカスタマイズ

次のリクエストは、グラフの左下に 2 つの (dual) 単位を示すスケールバーを配置します。ここでは、1 つの人口統計レイヤと 2 つのフィーチャレイヤも定義されています。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
AVE.STATE_PROV_POPULATION
BY  CITY_LATITUDE
BY CITY_LONGITUDE
WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=AVE.STATE_PROV_POPULATION, BUCKET=color, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"extensions": {
  "com.esri.map":
  {
   "scalebar" : {"scalebarUnit": "dual",
                 "attachTo" : "bottom-left"
                             },
  "baseLayer":{ "title" : "Base Layer",
   "basemap": "streets",
   "center": [-98, 40],
   "zoom": 4
         },
  "overlayLayers":
  [
    {
     "title": "WebFOCUS Layer",
     "layerType": "bubble",
     "geometryXY": {
           "y": "lat",
           "x": "lng"},
    "geometrySourceType": "seriesdata"
    },
```

```
    {
     "title": "Demographic Layer",
     "layerType": "tile",
     "url": "https://server.arcgisonline.com/ArcGIS/rest/services/Demographics/
USA_1990-2000_Population_Change/MapServer"
   },
   {
    "title": "Country Feature Layer",
    "layerType": "featurelayer",
    "geometrySourceType": "esri",
    "geometryLocateField": ["Country"],
    "geometryDataField": "name",
    "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
    },
    {
     "title": "City Feature Layer",
     "layerType": "featurelayer",
     "geometrySourceType": "esri",
     "geometryLocateField": ["state_name"],
     "geometryDataField": "name",
     "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
USA_Counties_Generalized/FeatureServer/0"
    },
   ],
  }
 }
 *END
ENDSTYLE
END
```

出力結果は次のとおりです。



参照　目次表示の制御

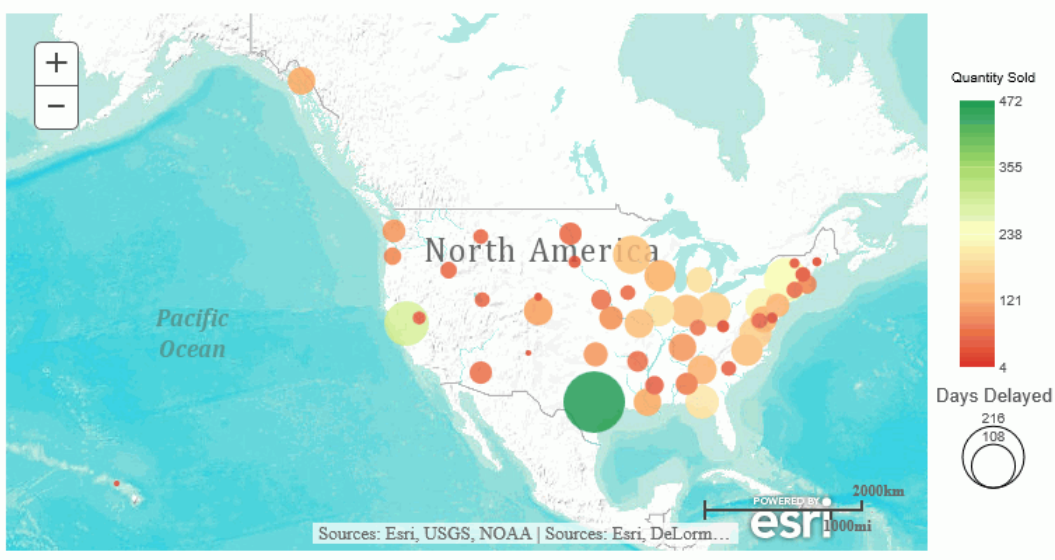
[目次] (レイヤ) コントロールは、com.esri.map オブジェクトの "drawToc": true プロパティ (デフォルト設定) により描画されます。このプロパティを false に設定することで [目次] コントロールを非表示にすることができます。

例 Reporting Server 構文による ESRI マップグラフからの目次の削除

次のリクエストは、マップグラフから目次を削除します。

```
DEFINE FILE WF_RETAIL_LITE
GEOPOINT/A200 = GIS_POINT('4326', STATE_PROV_CAPITAL_LONGITUDE,
STATE_PROV_CAPITAL_LATITUDE);
END
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED QUANTITY_SOLD
BY GEOPOINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=color, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"extensions": {
  "com.esri.map": {
      "drawToc": false,
       "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                       "map_by_field": "GEOPOINT"
                                          }
                                    }
         }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "terrain"
                }
            ]
          }
                        }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

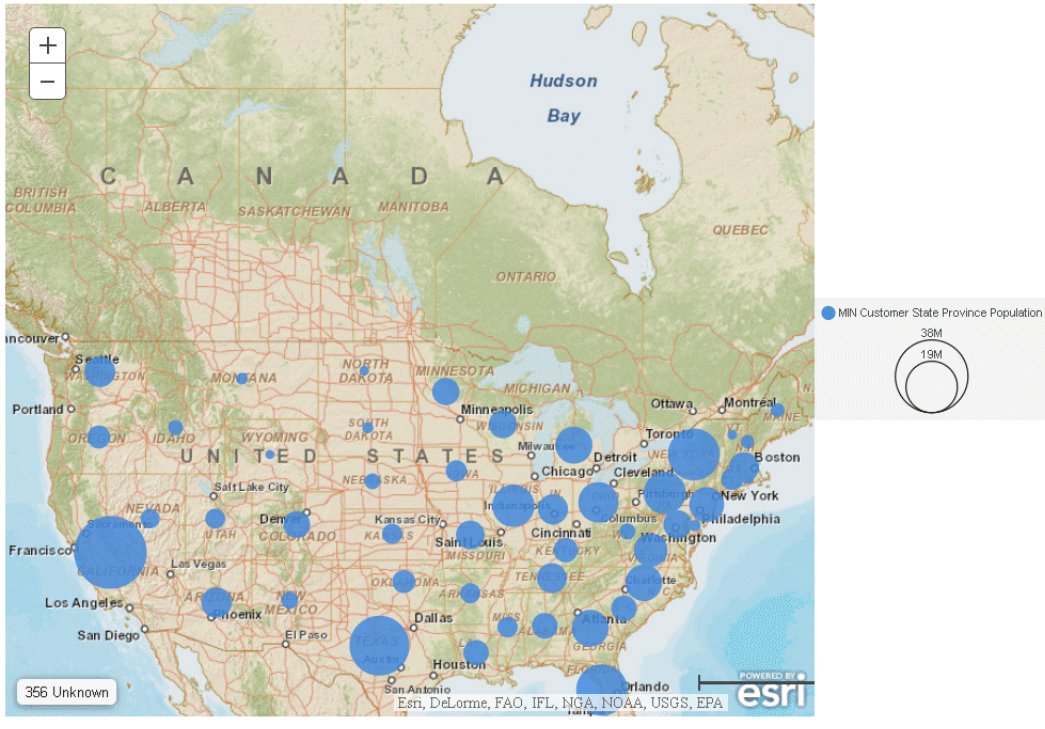


例　　グラフエンジン構文による ESRI マップグラフからの目次の削除

次のリクエストは、マップグラフから目次を削除します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.STATE_PROV_POPULATION
BY STATE_PROV_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=STATE_PROV_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.STATE_PROV_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": true},
"bubbleMarker": {"maxSize": "10%" },
"extensions": {
 "com.esri.map": {
  "drawToc":false,
  "overlayLayers":
  [
   {
   "title": "WebFOCUS Request Output",
   "layerType": "bubble",
   "geometrySourceType": "esri",
   "geometryLocateField": ["STATE_NAME"],
   "geometryDataField": "name",
   "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/ArcGIS/rest/services/
USA_States_Generalized/FeatureServer/0"
   }
  ],
   "baseLayer": {
   "basemap": "streets",
   "center":[-97,48],
   "zoom":4
   }
  }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

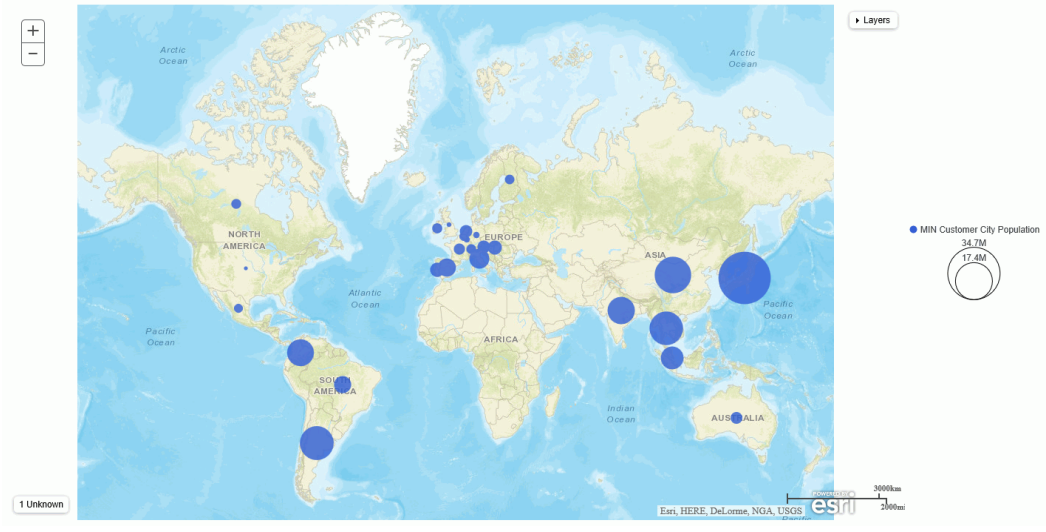


参照　　欠落した地理情報の処理

リクエスト内に地理情報が不明の位置がある場合は、グラフ上にそのことが表示されます。たとえば、次のリクエストについて考察します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
BY COUNTRY_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COUNTRY_NAME, BUCKET=location, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {
    "visible": true
},
"bubbleMarker": {"maxSize": "10%" },
"extensions": {
 "com.esri.map": {
  "overlayLayers":             [
  {
  "title": "Population",
  "layerType": "bubble",
  "geometrySourceType": "esri",
  "geometryLocateField": ["Country"],
  "geometryDataField": "name",
  "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
   }
  ],
 "baseLayer": {
   "basemap": "streets"
  }
 }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。[2 不明] のラベルが付いたボックス上にマウスポインタを置くと、欠落した地理情報の値が表示されます。



この情報を使用して、欠落したロケーションの値を選別し、欠落した地理情報の値を含むエンドポイントを検索することができます。

参照　マップグラフリクエストへのページ属性カテゴリの追加

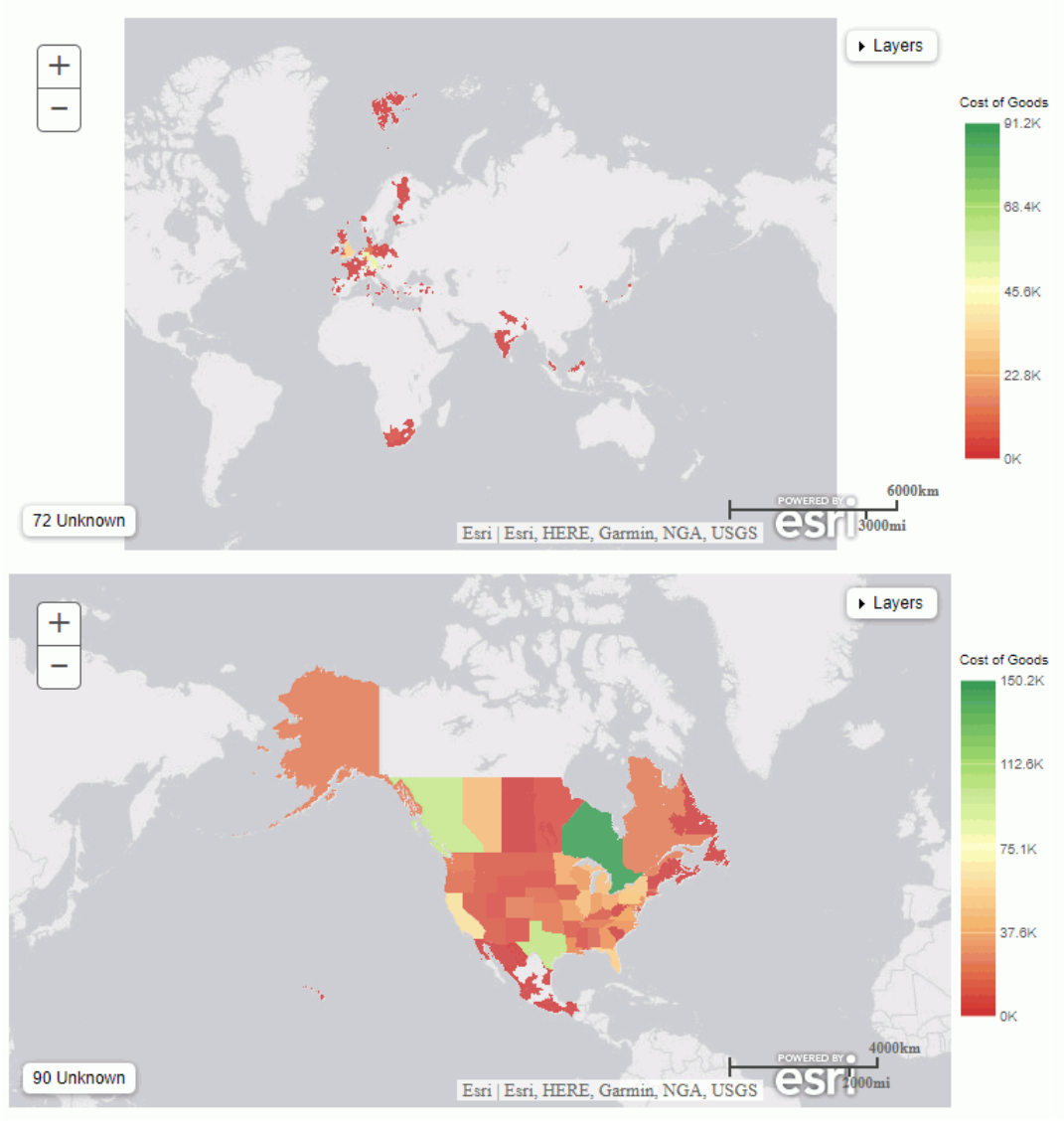
page 属性は、上位ソートフィールドの値に基づいて別個のグラフを生成します。

例 Reporting Server 構文による ESRI マップグラフへのページ属性の追加

次のリクエストでは、BUSINESS_REGION が上位ソートフィールドとしてリクエストに追加され、page カテゴリに割り当てられています。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY BUSINESS_REGION
BY STATE_PROV_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, TITLETEXT='WebFOCUS Chart', CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=N1, BUCKET=page, $
TYPE=DATA, COLUMN=N3, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
    "com.esri.map": {
        "overlayLayers":             [
                {
                    "ibiDataLayer": {
                        "map-metadata": {
                            "map_by_field": "STATE_PROV_NAME"
                        }
                    }
                }
            ],
        "baseMapInfo": {
            "customBaseMaps":                 [
                    {
                        "ibiBaseLayer": "gray"
                    }
                ]
        }
    }
}

*END
ENDSTYLE
END
```

下図は、生成されたグラフページ 4 ページのうちの最初の 2 ページを示しています。



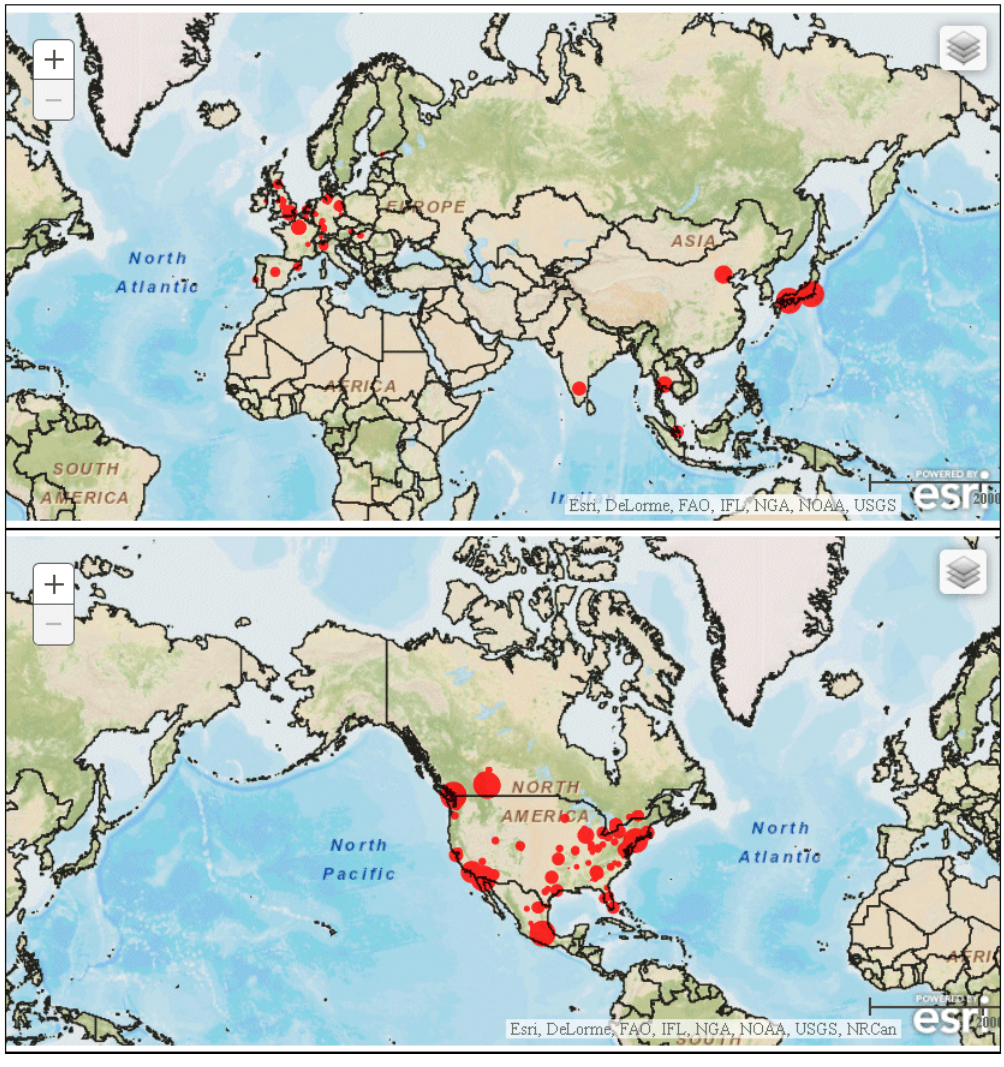
例　グラフエンジン構文による ESRI マップグラフへのページ属性の追加

次のリクエストでは、BUSINESS_REGION が上位ソートフィールドとしてリクエストに追加され、page カテゴリに割り当てられています。WebFOCUS は、地理情報に基づいてグラフごとに別々の最小ズームレベルやその他プロパティの初期値を計算します。そのため、このリクエストが、初期画面に各グラフの境界ボックスが完全表示されるように最小ズームレベルを調整します。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.CITY_POPULATION
  BY BUSINESS_REGION
BY  CITY_LATITUDE
BY CITY_LONGITUDE

WHERE COUNTRY_NAME NE 'Taiwan'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=page, $
TYPE=DATA, COLUMN=CITY_LATITUDE, BUCKET=latitude, $
TYPE=DATA, COLUMN=CITY_LONGITUDE, BUCKET=longitude, $
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=size, $
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker": {"maxSize":"5%"},
"extensions": {
    "com.esri.map":
          {
     "baseLayer":{ "title" : "Base Layer",
     "basemap": "streets",
     "minZoom" : 2
 },
    "overlayLayers":             [
       {
    "title": "WebFOCUS Request Layer",
    "layerType": "bubble",
    "geometryXY": {
    "y": "lat",
    "x": "lng"},
    "geometrySourceType": "seriesdata"
    },
```

```
    {
    "title": "Country Feature Layer",
    "layerType": "featurelayer",
    "geometrySourceType": "esri",
    "geometryLocateField": ["Country"],
    "geometryDataField": "name",
    "url": "https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/
World_Countries_(Generalized)/FeatureServer/0"
    }
    ],
  }
}
*END
ENDSTYLE
END
```

下図は、生成されたグラフページ 4 ページのうちの最初の 2 ページを示しています。



Leaflet マップグラフの生成

Leaflet マップグラフでは、WebFOCUS スタイルシートの *GRAPH_JS ブロックに mapProperties オブジェクトを追加する必要があります。これには、マップの不透明度、ズームレベル、縮尺コントロール、および初期位置を設定するプロパティが含まれます。また、ベースレイヤのマップタイル指定やオーバーレイレイヤの定義も必要です。

構文　Leaflet マップグラフのマッププロパティ指定

スタイルシートの GRAPH_JS ブロックの mapProperties オブジェクトで、マップビューアエンジン (leaflet) およびマッププロパティを指定します。

```
*GRAPH_JS
"mapProperties": {
"engine": "leaflet",
  "leafLet":{
   map_properties
}
}
  *END
```

説明

`"map_properties"`

グラフに表示するマップタイル、マップオーバーレイ、マップコントロールを制御するプロパティです。

別のマップタイルエンジンを使用することもできます。

構文　Leaflet マップグラフのベースレイヤプロパティの定義

```
"mapProperties":{
"baselayers": [{
"title": "string",
"layerInfo": {
  "maxZoom": number,
  "minZoom": number,
  "attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
  "url": function(){
return "https://services.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/
MapServer/tile/{z}/{y}/{x}";
}
}]
}
```

説明

`"title": "string"`

レイヤのタイトルです。このタイトルは、地図の右上に表示された [目次] コントロールをクリックした際に表示されます。

`"maxZoom": number`

-1 (完全自動) を指定するか、最大許容ズームレベルを定義する 1 から 19 までの数値を指定します。

"minZoom": number

-1 (完全自動) を指定するか、最小許容ズームレベルを定義する 1 から 19 までの数値を指定します。

"url": function()

返されたマップサービスの初期ズームレベルおよび中央位置を定義します。

z

"initPos": "level" プロパティで指定された初期ズームレベルです。

y

"initPos": "center" プロパティで指定された初期表示の中央位置の緯度です。

x

"initPos": "center" プロパティで指定された初期表示の中央位置の経度です。

注意

- minZoom と maxZoom に同じ値を指定してズーム (+/-) コントロールを無効にすることはできますが、値をすべて削除することはできません。
- attribution プロパティで定義した著作権ステートメントは、地図の右下に表示されます。上記のコードに含まれている attribution プロパティを使用すると、WebFOCUS に同梱されているマッププロバイダの著作権情報が表示されます。別のマッププロバイダの地図を使用する場合は、そのプロバイダの著作権情報が含まれるよう attribution プロパティを編集します。
- 上記のコードに含まれている url プロパティを使用すると、ESRI から提供されたマップタイプの 1 つである World_Street_Map が参照されます。別のマッププロバイダの地図を使用する場合は、そのマップタイルサーバが参照されるよう、このプロパティを編集します。

以下は、ESRI から提供されるさまざまなタイプの地図とアクセスに必要な URL 参照を示しています。

```
https://services.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/MapServer/
tile/{z}/{y}/{x};
https://services.arcgisonline.com/ArcGIS/rest/services/World_Topo_Map/MapServer/
tile/{z}/{y}/{x};
https://services.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer/
tile/{z}/{y}/{x};
https://services.arcgisonline.com/ArcGIS/rest/services/World_Terrain_Base/MapServer/
tile/{z}/{y}/{x};
https://services.arcgisonline.com/ArcGIS/rest/services/Canvas/World_Light_Gray_Base/
MapServer/tile/{z}/{y}/{x};
https://services.arcgisonline.com/ArcGIS/rest/services/NatGeo_World_Map/MapServer/
tile/{z}/{y}/{x};
https://services.arcgisonline.com/ArcGIS/rest/services/Ocean_Basemap/MapServer/
tile/{z}/{y}/{x};
```

マップグラフに表示される地図は、これらのマップの提供元サービスが所有者になるため、マップグラフには著作権属性を含めることが要求されます。著作権属性は、プロパティの 1 つとして Leaflet マップのベースレイヤに追加する必要があります。

次のコードは、WebFOCUS とともにインストールされる OpenStreetMap マップを使用する場合の attribution および url プロパティを示しています。

```
"attribution": function(){ return '&|copy; <a href="http://www.ibi.com">IBI</a> | ' +
'Map data &|copy; <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a>
contributors, Open Database Licence';}
    },
"url": function(){
return tdgchart.getScriptPath().replace(/¥/tdg¥/.*/,'') +
 '/maptiles/osm/{z}/{x}/{y}.png'}
```

次のコードは、WebFOCUS とともにインストールされ、Mapbox から提供される別タイプの OpenStreetMap マップを使用する場合の attribution および url プロパティを示しています。

```
"attribution": "Map data &|copy; <a href='http://www.openstreetmap.org/
copyright'>OpenStreetMap</a> contributors, Open Database Licence | Map Tiles by <a
href='http://mapbox.com'>Mapbox</a>",

"url": "http://{s}.tiles.mapbox.com/v3/bclc-apec.map-rslgvy56/{z}/{x}/{y}.png",
```

例　Leaflet マップのベースレイヤプロパティの定義

次のリクエストは、初期の位置とズームレベルを設定し、マップのベースレイヤを生成します。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MDN.STATE_PROV_POPULATION
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN = STATE_PROV_NAME, BUCKET=LOCATION,$
TYPE=DATA, COLUMN= MDN.STATE_PROV_POPULATION, BUCKET = COLOR,$
*GRAPH_JS
"legend": {"visible": false},
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
  "initPos": {
                  "center": [37.8, -96],
                  "level": 4
            },
 "baselayers": [{
  "title": "ArcGIS_World_Street_Map",
  "layerInfo": {
   "maxZoom": 17,
   "minZoom": 0,
   "attribution": function(){return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
  "url": function(){return "https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}";
}
}]
}
},
*END
ENDSTYLE
END
```

リクエストの出力はオーバーレイレイヤに描画されるため、出力された地図には表示されません。

地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

参照　ロケーションファイルから WebFOCUS リクエストへのフィールドの関連付け

次のコードは、米国の州および市のリストが格納された US.JSON ロケーションファイルの先頭部分を示しています。primary_layer (地図の区分) の名前は、regions です。position プロパティでは、市の緯度と経度が定義されています。

```
"map_type":"country",
 "country_name":"United States of America",
 "iso_a2":"US",
 "primary_layer":"regions",
 "layers":[{
  "type":"cities",
  "geocode":["city_name"],
  "units":"decimal_degrees",
  "geometry_type":"point",
  "primary_name_field":"city_name",
  "features":[{
   "city_name":"New York",
   "rank":"0",
   "region":"New York",
   "position":[-73.98,40.7499],
   "size":24.2
  },

  {
   "city_name":"Washington, D.C.",
   "rank":"0",
   "region":"District of Columbia",
   "position":[-77.0094,38.8995],
   "size":22
  },
  {
   "city_name":"Los Angeles",
   "rank":"0",
   "region":"California",
   "position":[-118.1799,33.9899],
   "size":23.6
```

次のコードは、州の位置情報が定義されたセクションの先頭部分を示しています。ここでは、type が regions に設定され、フィールド名として state_name および state_abbr が含まれています。州を定義するセクションには、州の境界を示す緯度と経度も定義されています。

```
"type":"regions",
  "geocode":["state_abbr","state_name","state_fips","iso_3166_2"],
  "units":"decimal_degrees",
  "geometry_type":"polygon",
  "primary_name_field":"state_name",
  "primary_abbr_field":"state_abbr",
  "primary_id_field":"state_abbr",
  "features":[{
   "state_name":"Hawaii",
   "state_abbr":"HI",
   "state_fips":"15",
   "iso_3166_2":"US-HI",
   "borders":[[[-155.9633,19.0815],[-155.9432,19.3442],
[-156.1192,19.7364],[-155.8843,20.0183],[-155.959,20.2268],
[-155.8367,20.3212],[-155.1403,19.9894],[-154.7566,19.5069],
[-154.9396,19.3128],[-155.466,19.0953],[-155.649,18.8705]],
[[-156.5651,20.7389],[-156.7519,20.9136],[-156.6058,21.0925],
[-156.451,20.9687],[-156.2685,21.0037],[-155.9589,20.875],
[-155.9267,20.7817],[-156.0299,20.604],[-156.375,20.5289],
[-156.4845,20.579],[-156.5334,20.4629],[-156.75,20.4543],
[-156.7336,20.5818],[-156.4957,20.6915]],[[-158.3176,21.5391],
[-158.3242,21.6156],[-157.9417,21.7545],[-157.6505,21.485],
[-157.6009,21.2829],[-157.8102,21.1992],[-158.1347,21.2423]],
[[-159.7936,21.9397],[-159.8481,22.076],[-159.6129,22.2722],
[-159.3896,22.2875],[-159.25,22.1757],[-159.2843,21.9406],
[-159.4328,21.8207]],[[-157.015,21.2611],[-156.722,21.226],
[-156.6527,21.125],[-156.8779,20.9933],[-157.3637,21.0923],
[-157.2812,21.2725]],[[-175.9394,27.6882],[-176.0396,27.7582],
[-175.921,27.8214],[-175.9814,27.9132],[-175.7043,27.9827],
[-175.7341,27.7388]],[[-157,20.6968],[-157.1112,20.9312],
[-156.8684,20.9643],[-156.75,20.8076],[-156.875,20.6909]],
[[-160.2589,21.7428],[-160.2799,21.9206],[-160.0634,22.0726],
[-160.031,21.8693]],[[-166.1722,23.587],[-166.3844,23.8319],
[-166.307,23.9248],[-166.1823,23.9023],[-166.1067,23.7425]],
[[-178.3014,28.3283],[-178.4404,28.4051],[-178.375,28.51],
[-178.2384,28.4561]],[[-174.0064,25.9902],[-174.0547,26.1077],
[-173.9579,26.1356],[-173.9033,26.06]],[[-171.7374,25.7065],
[-171.8096,25.7813],[-171.75,25.8379],[-171.6679,25.7812]],
[[-161.9318,23.0053],[-161.9825,23.0912],[-161.8875,23.1103]],
[[-164.723,23.5217],[-164.7611,23.5934],[-164.6826,23.6228],
[-164.6453,23.5515]],[[-160.5707,21.6066],[-160.5918,21.6858],
[-160.5087,21.7032],[-160.4849,21.6324]],[[-168.0235,24.9544],
[-168,25.0522],[-167.9442,24.9898]]]
  },
```

マップデータをデータソース内のデータと関連付けるには、WebFOCUS リクエストのソートフィールドと同一のデータが格納されたロケーションファイルで、使用する区分の type と、フィールド名を識別する必要があります。たとえば、このロケーションファイルを使用し、データソース内に州名のフィールドが含まれている場合、このロケーションファイルから使用する区分の type として regions、フィールド名として state_name を識別する必要があります。

構文　Leaflet マップのオーバーレイレイヤプロパティの定義

```
"mapProperties":{
"overlayLayers": [{
    "title": "string",
    "dataLookup": "properties.field",
    "layerInfo": {
     "maxZoom": number,
     "minZoom": number,
     "type": "string"
    },
    "type": "string",
    "url": function(){ return tdgchart.getScriptPath() + "map/
location.json"}
   }]
}
```

説明

`"title": "string"`

レイヤのタイトルです。このタイトルは、地図の右上に表示された [目次] コントロールをクリックした際に表示されます。

`"dataLookup": "properties.field"`

リクエストで指定するソート値と同一のデータが格納されているロケーションファイル内のフィールド名です。緯度と経度に基づくマップレイヤには必要ありません。

`"layerInfo":`

最小ズームレベル、最大ズームレベル、および返されるデータのタイプを定義します。

`"maxZoom": number`

-1 (完全自動) を指定するか、最大許容ズームレベルを定義する 1 から 19 までの数値を指定します。

`"minZoom": number`

-1 (完全自動) を指定するか、最小許容ズームレベルを定義する 1 から 19 までの数値を指定します。

"type": "string"

ロケーションファイルに記述されている地理的区分のタイプです。

"type": "string"

使用するロケーションファイルのタイプを定義します。有効な値には、次のものがあります。

- "tdg" - WebFOCUS に同梱されている JSON ファイルと geoJSON ロケーションファイルの両方をサポートします。
- "geojson" - geoJSON ロケーションファイルのみをサポートします。
- "latlng" - WebFOCUS リクエストで提供される経度 (x) と緯度 (y) の値を指定します。

"url": function

返されるマップのロケーションファイルを定義します。

location

ロケーションファイルの名前です (例、US.JSON)。geoJSON ファイルを使用する場合や WebFOCUS から提供されていない別の地域を使用する場合は、拡張子を正しい値に変更し、そのファイルの格納先を指定する必要があります。

注意

- url 属性では、ロケーションファイルのパスを指定します。WebFOCUS に同梱されている tdgchart エンジンの場合、WebFOCUS のインストール時にこのパスが構成されます。geoJSON ファイルを使用する場合は、拡張子を正しい値に変更し、そのファイルの格納先を指定する必要があります。

  次のコードは、WebFOCUS インストール時にサンプルとしてインストールされた sample_ggsales_regions.geojson ロケーションファイルの使用方法を示しています。

```
"url": function() { return tdgchart.getScriptPath().replace(/¥/tdg
¥/.*/,"/web_resource/map/") +"sample_ggsales_regions.geojson"}
```

例　Leaflet コロプレスマップのオーバーレイレイヤの定義

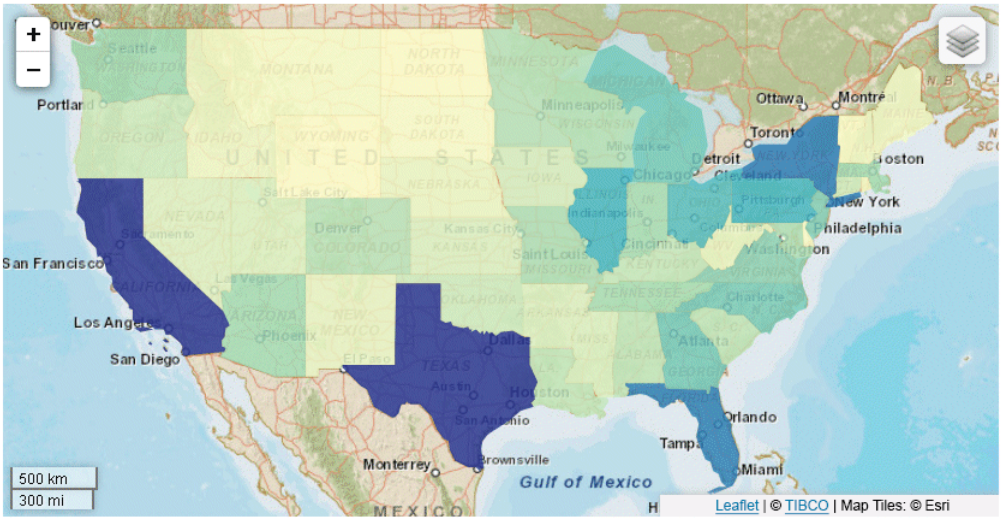
次のリクエストでは、オーバーレイレイヤ、レイヤコントロール、縮尺コントロールを定義します。"type" プロパティは "regions" に設定され、"dataLookup" プロパティは "properties.state_name" に設定されています。これは、これらの値がロケーションファイルで指定され、かつ WF_RETAIL_LITE データソース内のデータと一致するためです。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MDN.STATE_PROV_POPULATION
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN = STATE_PROV_NAME, BUCKET=LOCATION,$
TYPE=DATA, COLUMN= MDN.STATE_PROV_POPULATION, BUCKET = COLOR,$
*GRAPH_JS
"legend": {"visible": false},
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
           },
```

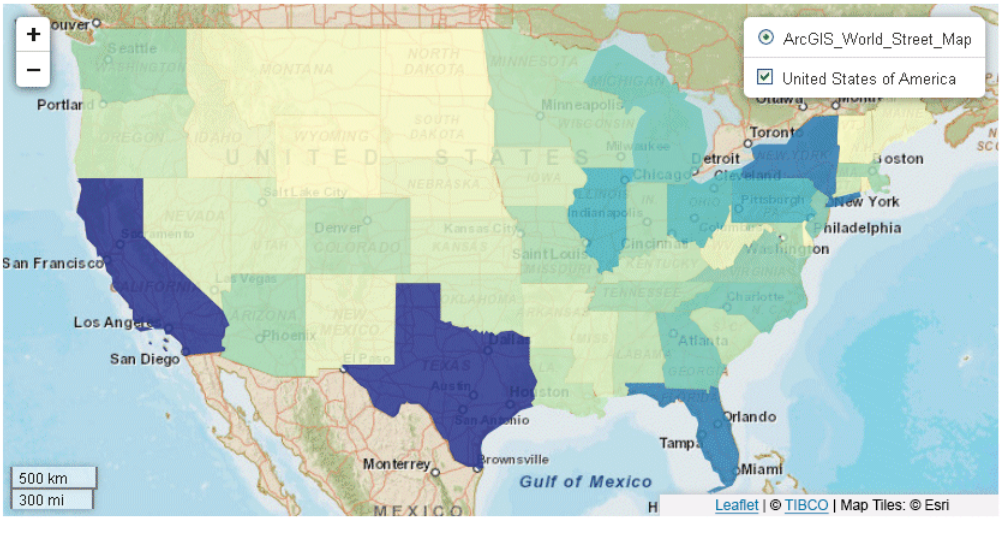
```
   "overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],
  "controls": [
    {"control": "L.Control.Layers"},
    {
     "control": "L.Control.Scale",
     "options": {
      "imperial": true,
      "metric": true }
    }
   ],

"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 17,
"minZoom": 0,
"attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
"url": function(){return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
},

*END
ENDSTYLE
END
```

リクエストの出力は、ベースレイヤの前面に描画されたコロプレスマップです。この地図では、さまざまな州の形状を表す多角形が人口の中央値に基づいて色分けされています。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

レイヤコントロールの上にマウスポインタを置くと、各レイヤのタイトルが表示されます。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

このコロプレスマップでは、州ごとに色分けされた人口値および実際値を示す指標はありません。これらのオブジェクトは、legend および tooltip プロパティを使用して追加することができます。

参照　コロプレスマップのポイント位置の指定

グラフリクエストでコロプレスマップ (ON GRAPH SET LOOKGRAPH CHOROPLETH) を指定するが、位置フィールドが多角形ではなくポイントを表す場合は、このポイント位置を表すバブルマーカー (丸) が描画されます。マーカーのサイズはすべて同一になります。また、color カテゴリに割り当てられたフィールドがマーカーの色指定に使用されます。

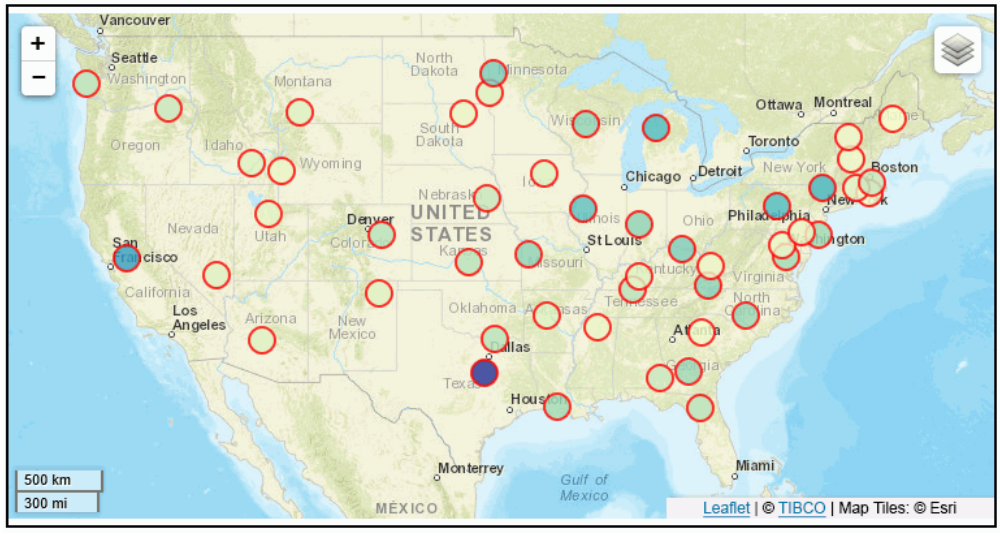
例　Leaflet コロプレスマップのバブルマーカーからのデフォルト境界線の削除

デフォルト設定では、コロプレスの LOOKGRAPH パラメータにポイント位置を使用する Leaflet マップグラフのマーカーには赤色の境界線が使用されます。この境界線は、シリーズの境界線幅を 0 (ゼロ) に設定することで削除することができます。

次のリクエストは、コロプレスマップを指定しますが、ポイント位置 (緯度と経度) を使用します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
WHERE COUNTRY_NAME EQ 'United States'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
*GRAPH_JS
"bubbleMarker":{"maxSize":"10%"},
"mapProperties": {
"leaflet": {
"initPos": {
"center": [37.8, -96],
"level": 4
},
"overlayLayers": [{
"title": "US",
"layerInfo": {
"type": "regions"
},
"type": "latlng",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],
"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 16,
"attribution": function() { return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy;Esri"; }
},
"url": function() { return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}'; }
}]
}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

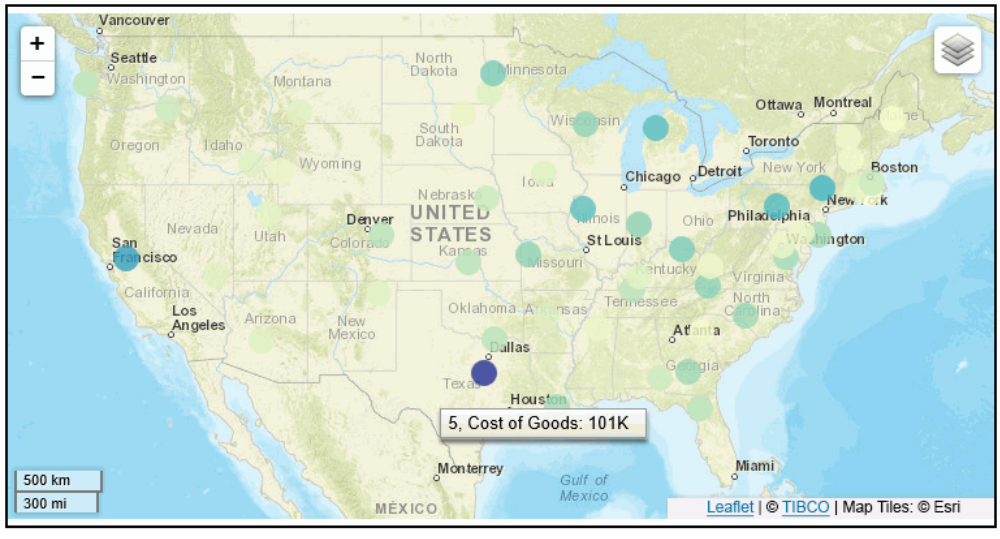


地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

次のリクエストでは、シリーズのプロパティを使用して境界線の幅を 0 (ゼロ) に設定します。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
WHERE COUNTRY_NAME EQ 'United States'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
*GRAPH_JS
"bubbleMarker":{"maxSize":"10%"},
"mapProperties": {
"leaflet": {
"initPos": {
"center": [37.8, -96],
"level": 4
},
"overlayLayers": [{
"title": "US",
"layerInfo": {
"type": "regions"
},
"type": "latlng",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],
"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 16,
"attribution": function() { return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy;Esri"; }
},
"url": function() { return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}'; }
}]
}
},
"series": [
         {
         "series": "all",
          "border": {
                    "width": 0
                    }
          }
         ]
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

初期位置の定義

マップグラフの自動位置設定は、次のように機能します。

- デフォルト設定では、マップビューアがデータソース内のマーカーすべてを包括する境界ボックスを設定し、データの表示範囲が適切になるようズームレベルを選択します。たとえば、アメリカ合衆国本土 48 州のデータの場合、合衆国本土の全体およびメキシコとカナダの一部のみが表示されるようズームレベルが設定されます。
- デフォルト設定では、マップビューアが自動境界ボックスの正確な中央位置を中心点として設定します。アメリカ合衆国本土の場合、画面中央に中西部の州が表示されます。

デフォルト設定の代わりに、リクエスト内のマッププロパティで、ユーザ独自の初期位置および初期ズームレベルを指定することもできます。

構文　初期位置の定義

```
"initPos":{
  "center":[latitude, longitude],
  "level": number}
```

説明

`"center":[latitude, longitude]`

地図の中央位置の緯度と経度です。

`"level": number`

グラフの初期ズームレベルです。有効な値は、1 (地球全体) から 19 までの数値です。

例　緯度と経度を使用した Leaflet マップの初期位置およびズームの定義

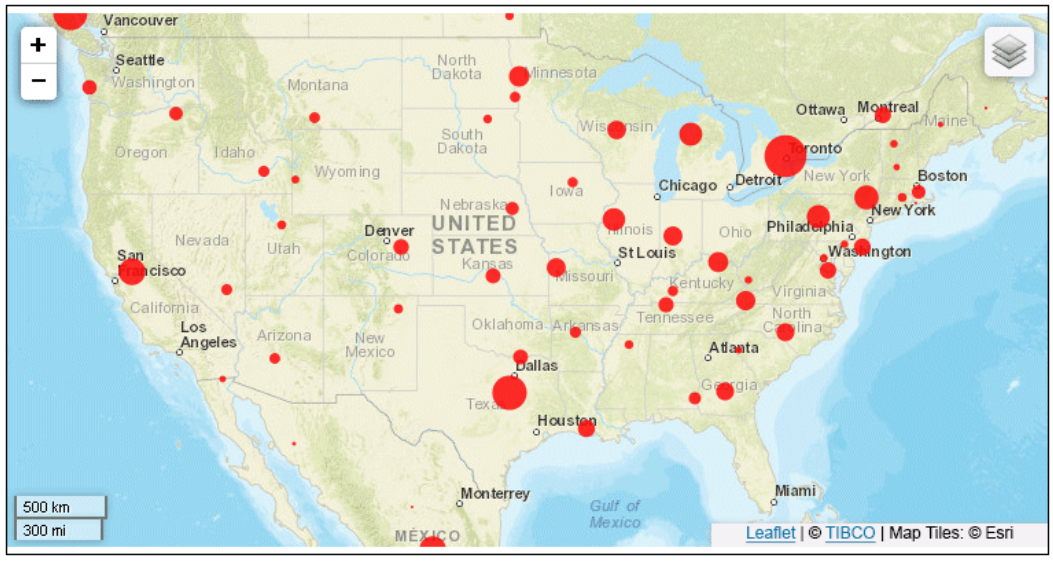
次のリクエストは、緯度と経度に基づいたバブルマップを生成します。初期位置とズームは、マップの初期表示が米国になるように設定されています。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
type=data, column=n1, bucket=latitude,$
type=data, column=n2, bucket=longitude,$
type=data, column=n3, bucket=size,$
*GRAPH_JS
"legend": {"visible": false},
"bubbleMarker":{"maxSize":"10%"},
"mapProperties": {
  "leaflet": {
   "initPos": {
    "center": [37.8, -96],
     "level": 4
              },
```

```
 "overlayLayers": [{
  "title": "US",
  "layerInfo": {
    "type": "regions"
           },
  "type": "latlng",
  "url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
          }],
  "baselayers": [{
   "title": "ArcGIS_World_Street_Map",
   "layerInfo": {
    "maxZoom": 16,
    "attribution": function() {  return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri"; }
                },
 "url": function() { return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}'; }
          }]
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

マップの不透明度の調整

mapOpacity プロパティを使用して、コロプレスマップまたはバブルマップの不透明度 (透明の度合い) を指定することができます。

構文　コロプレスマップまたはバブルマップの不透明度の設定

スタイルシートの mapProperties ブロックで、次のプロパティを指定します。

```
"mapOpacity": number
```

説明

```
"mapOpacity": number
```

0 (透明) から 1 (不透明) までの数値です。デフォルト値は 0.82 です。

マップビューア内外の要素の調整

グラフの凡例、マップの縮尺、マップの許容ズームレベルなどの要素を調整することができます。

凡例のフォーマット設定や位置の変更を行えるほか、凡例全体を非表示にすることもできます。詳細は、269 ページの 「 凡例のプロパティ 」 を参照してください。

マップの縮尺コントロールおよびレイヤコントロールの調整

縮尺コントロールは、マイル、キロメートル、またはその両方に設定することも、省略することもできます。レイヤコントロールは、マップに含めることも、省略することもできます。スタイルシートの leaflet ブロックで controls オブジェクトを省略した場合、デフォルト設定で両方のコントロールが出力結果に表示されます。

構文　マップの縮尺コントロールおよびレイヤコントロールの調整

スタイルシートの leaflet ブロックに controls オブジェクトを追加するには、次の構文を使用します。

```
 "leaflet":{
 "controls": [
    {"control": "L.Control.Layers"},
    {"control": "L.Control.Scale",
     "options": {
      "imperial": boolean,
      "metric": boolean}
    }
   ],
   ...
}
```

説明

`"imperial": boolean`
マイルの表示を制御します。有効な値には、次のものがあります。

- true - マイルを表示します。これがデフォルト値です。
- false - マイルを表示しません。

`"metric": boolean`
キロメートルの表示を制御します。有効な値には、次のものがあります。

- true - キロメートルを表示します。これがデフォルト値です。
- false - キロメートルを表示しません。

たとえば、次のようにプロパティを指定すると、縮尺コントロールにマイルとキロメートルの両方が表示されます。

```
"leaflet": {
    "controls": [
    {"control": "L.Control.Layers"},
    {
     "control": "L.Control.Scale",
     "options": {
      "imperial": true,
      "metric": true }
    }
        ],
  ...
    }
```

次のようにプロパティを指定すると、縮尺コントロールが完全に非表示になります。

```
"leaflet": {
    "controls": [
    {"control": "L.Control.Layers"},
    {
        }
   ],
   ...
   }
```

次のようにプロパティを指定すると、レイヤコントロールと縮尺コントロールの両方が非表示になります。

```
"leaflet": {
       "controls": [
                  ],
   ...
        }
```

マップグラフでのグラフプロパティの統合

一部のグラフ全体のプロパティやヒートマップおよびバブルグラフに固有のプロパティを使用して、グラフ出力を調整することができます。

コロプレスマップでの色スケールの調整

colorScale プロパティを使用して、コロプレスマップの色スケールを定義することができます。色名、RGB 値、16 進数値を使用して、任意の数の色を指定することができます。たとえば、次のコードは、赤色、白色、青色の色スケールを定義します。

```
"colorScale": ("colors":["#C4161C", "white", "rgb(0,0,255)"]}
```

また、"colorScale": "colorMode" プロパティを使用して、マップおよび凡例に表示する不連続な色バンドを定義することもできます。詳細は、819 ページの 「色スケールの色モードの定義」 を参照してください。

Reporting Server 構文では、デフォルト設定で凡例が表示されます。凡例を削除するには、"legend": {"visible": false} を設定します。

例　Leaflet コロプレスマップへの色スケール凡例の追加

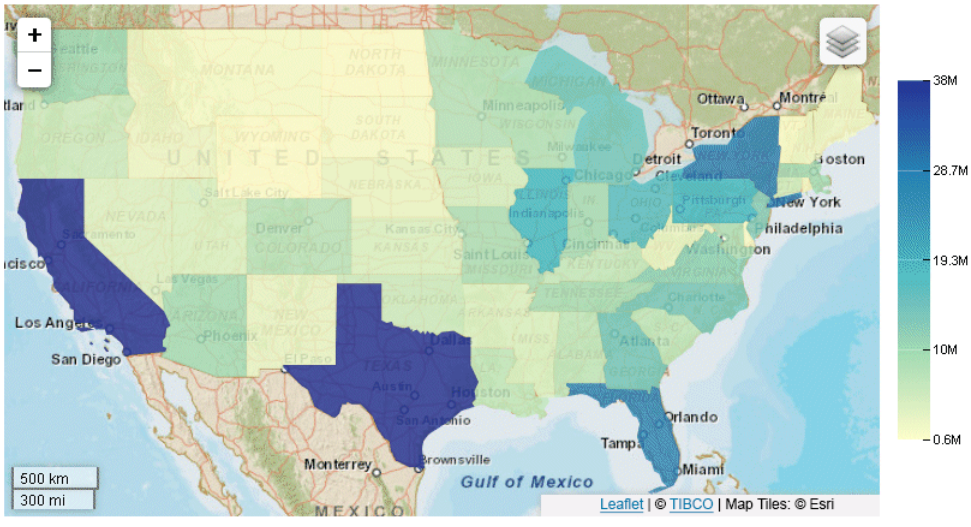
次のリクエストは、Leaflet グラフの凡例を表示します。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MDN.STATE_PROV_POPULATION
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
*GRAPH_JS
"legend": {"visible":true},
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
            },
```

```
"overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){ return tdgchart.getScriptPath() + 'map/US.json'}
}],
"controls": [
    {"control": "L.Control.Layers"},
    {
     "control": "L.Control.Scale",
     "options": {
      "imperial": true,
      "metric": true }
    }
   ],

"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 17,
"minZoom": 0,
"attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | "  + "Map Tiles: &|copy; Esri";}
},
"url": function(){return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
},
*END
ENDSTYLE
END
```

出力結果では、人口の範囲別に色分けされた凡例が表示されています。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

Leaflet マップグラフでの Hover 色の調整

Leaflet マップグラフでは、他のグラフタイプと同様に、色または不透明度のプロパティを定義して、マーカー上にマウスポインタを置いたときのマーカー表示を調整することができます。この調整を行うには、MouseOverIndicator プロパティを使用します。

たとえば、次のようにプロパティを指定すると、選択したグラフ領域またはマーカー上にマウスポインタを置いたときに 70% の不透明度になります。

```
"mouseOverIndicator": {
  "color": "70%"
}
```

次のようにプロパティを指定すると、マーカー上にマウスポインタを置いたときにマーカーが黄色になります。

```
"mouseOverIndicator": {
  "color": "yellow"
}
```

詳細は、859 ページの 「マウスオーバーインジケータのフォーマット設定 (mouseOverIndicator)」 を参照してください。

バブルマップでのマーカーの色とサイズの調整

シリーズ固有のプロパティを使用して、バブルグラフ (バブルマップを含む) のマーカーの色および境界線を制御することができます。

たとえば、次のコードは、マーカー色を赤色、境界線を透明に設定します。

```
"series": [
  {"series":0, "color": "red", "marker":{border: {color: "transparent"}}}
]
```

マーカーのプロパティについての詳細は、467 ページの 「シリーズマーカーのサイズ、境界線、色、形状、回転の定義」 を参照してください。

また、"bubbleMarker": "maxSize" プロパティを使用して、バブルマップまたはバブルグラフのバブルマーカーのサイズを制御することもできます。このプロパティは、バブルグラフのプロパティの 1 つで、mapProperties ブロックの一部ではありません。

構文　バブルマップの最大バブルサイズの設定

最大バブルの半径を指定するには、次のプロパティを使用します。

```
"bubbleMarker": {"maxSize": size}
```

説明

`"maxSize": size`

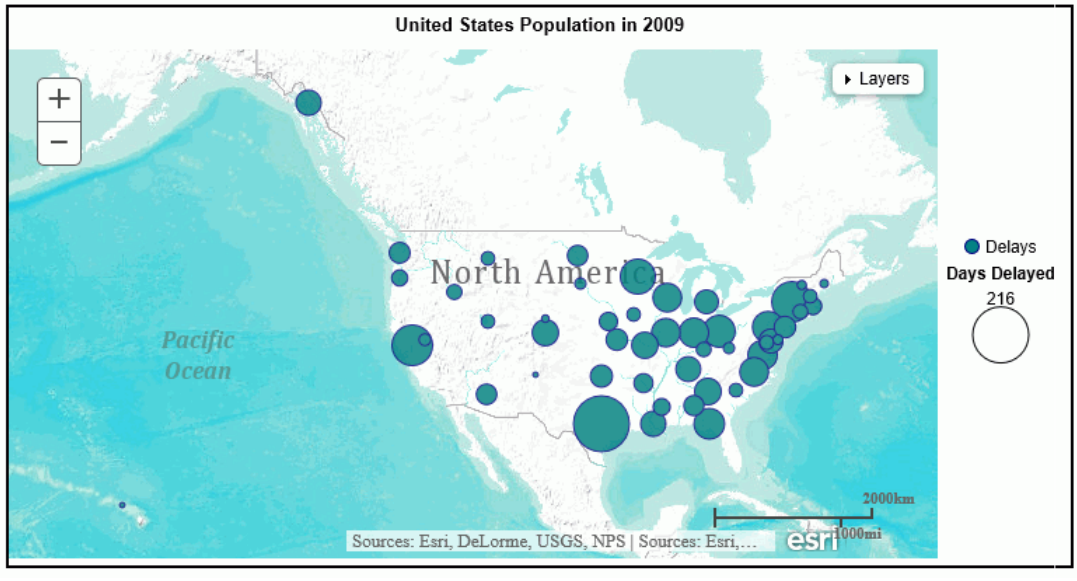
最大バブルの半径を指定します。数値をピクセル単位で指定するか、パーセント文字列として指定し、パーセント記号 (%) を含めて二重引用符 (") で囲みます ("1%" から "100%")。パーセント文字列を使用する場合は、グラフコンテナの幅と高さの最小値に基づいてパーセント計算されます。

例　ESRI バブルマップのバブルのフォーマット設定

次のリクエストは、Reporting Server 構文を使用してバブルマップを生成します。"bubbleMarker" オブジェクトで、バブルの最大サイズを設定します。シリーズの marker プロパティで、バブルの色を青緑色、境界線を紺色に設定し、凡例に表示するラベルを定義します。title プロパティで、グラフにタイトルを割り当てます。

```
DEFINE FILE WF_RETAIL_LITE
GEOPOINT/A200 = GIS_POINT('4326', STATE_PROV_CAPITAL_LONGITUDE,
STATE_PROV_CAPITAL_LATITUDE);
END
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY GEOPOINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"bubbleMarker": {"maxSize": "10%"},
"series":[
{"series": 0, "color": "teal", "marker": {"border":{"color": "navy",
"width": 1}},"label": "Delays"}],
"title": {"visible": true, "text": "United States Population in 2009"},
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                       "map_by_field": "GEOPOINT"
                                            }
                                     }
        }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "terrain"
                }
            ]
          }
                          }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

注意：グラフエンジン構文を使用して、バブルマーカーのフォーマットを設定することもできます。

例　Leaflet バブルマップのバブルのフォーマット設定

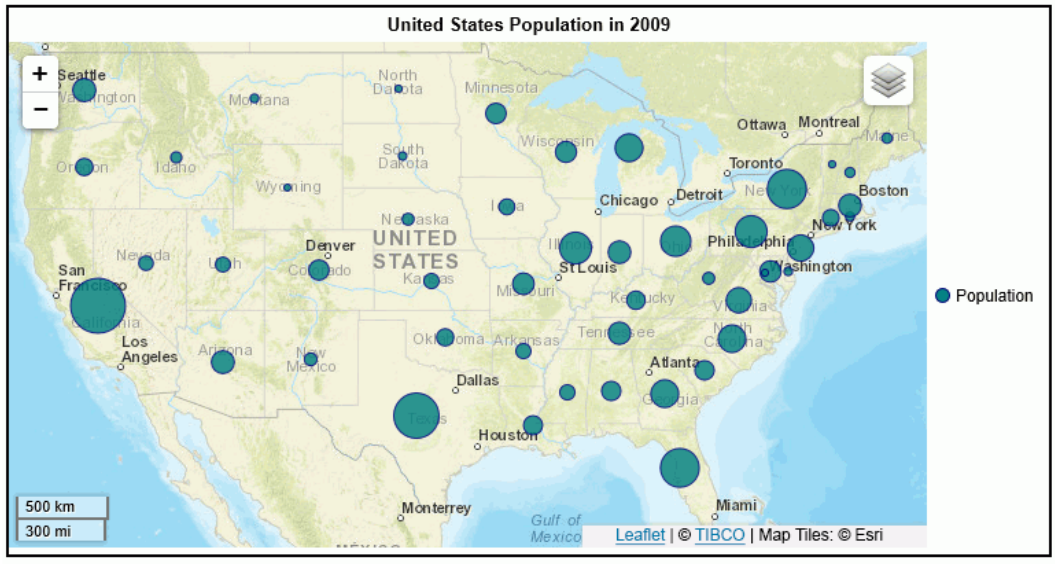
次のリクエストは、バブルマップを生成します。"bubbleMarker" オブジェクトで、バブルの最大サイズを設定します。シリーズの marker プロパティで、バブルの色を青緑色、境界線を紺色に設定し、凡例に表示するラベルを定義します。title プロパティで、グラフにタイトルを割り当てます。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MDN.STATE_PROV_POPULATION
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
WHERE TIME_YEAR EQ 2009
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
type=data, column=n1, bucket=latitude,$
type=data, column=n2, bucket=longitude,$
type=data, column=n3, bucket=size,$
*GRAPH_JS
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
           },

"overlayLayers": [{
"title": "US",
"layerInfo": {
"type": "regions"
},
"type": "latlng",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],

"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 17,
"minZoom": 0,
"attribution": function(){return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
"url": function(){return  'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
},
"legend": {"visible":true},
```

```
"bubbleMarker": {maxSize: "10%" },
"series":[{"series":0, "color": "teal", "marker": {"border":{"color": "navy", "width":
1}},"label": "Population"}],
"title": {"visible": true, "text": "United States Population in 2009"} ,
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

コロプレスマップの色およびツールヒントの定義

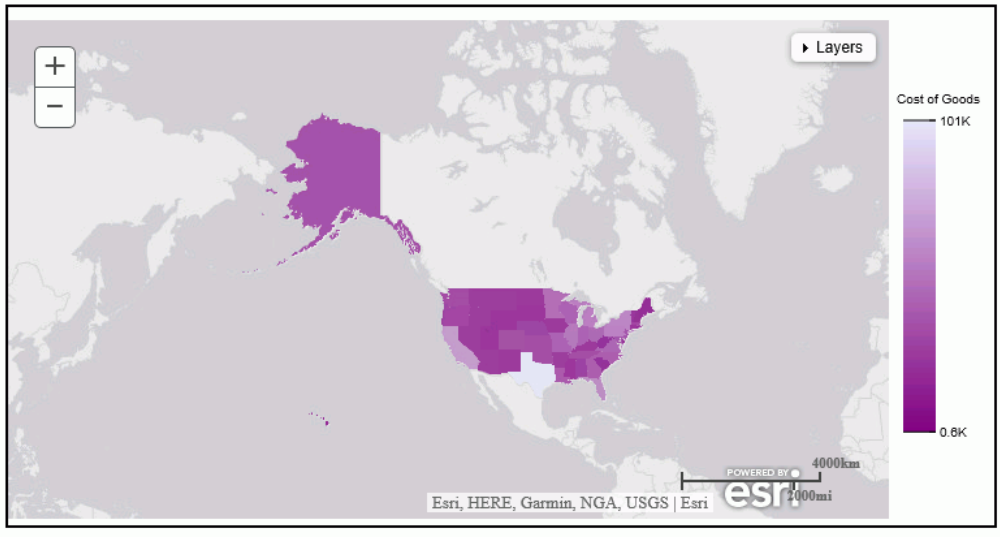
コロプレスマップの色スケールおよびツールヒントを定義することができます。不連続な色バンドとして色を表示することもできます。詳細は、819 ページの 「色スケールの色モードの定義」を参照してください。

例　ESRI コロプレスマップの色およびツールヒントの定義

次のリクエストは、Reporting Server 構文を使用してコロプレスマップを生成し、紫色と薄紫色で構成された色スケールを表示します。また、HTML スタイルのツールヒントを定義します。

```
DEFINE FILE wf_retail_lite
FIELD1/A30 (GEOGRAPHIC_ROLE=STATE) = STATE_PROV_NAME;
END
GRAPH FILE wf_retail_lite
SUM COGS_US
BY STATE_PROV_NAME
BY FIELD1
WHERE COUNTRY_NAME EQ 'United States'
AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=COGS_US, BUCKET=color, $
*GRAPH_JS_FINAL
"colorScale": {"colors": ["purple", "lavender"] },
"htmlToolTip": {"enabled": true, "snap": true},


"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "FIELD1"
                                            }
                                      }
        }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                        }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

注意：グラフエンジン構文を使用して、色およびツールヒントを定義することもできます。

例　Leaflet コロプレスマップの色およびツールヒントの定義

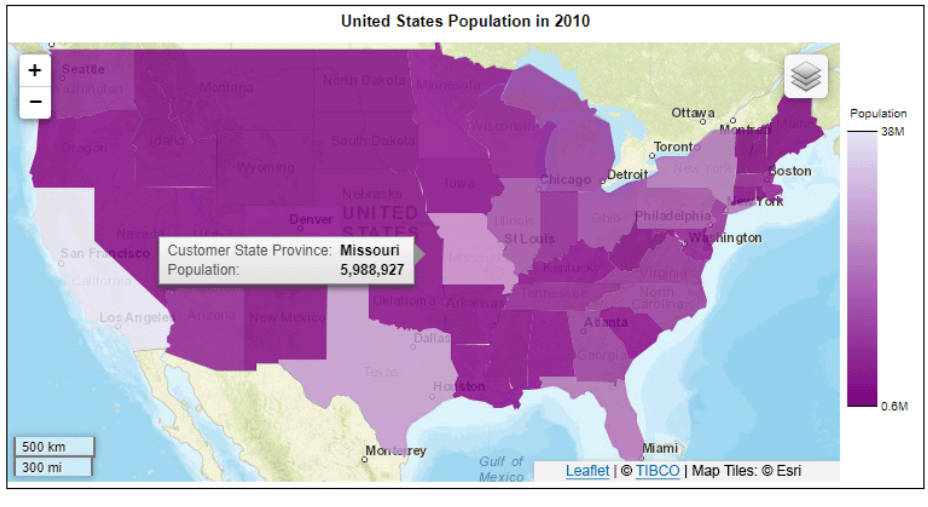
次のリクエストは、コロプレスマップを生成し、紫色と薄紫色で構成された色スケールを表示します。また、HTML スタイルの自動ツールヒントを定義します。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MDN.STATE_PROV_POPULATION AS Population
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN = STATE_PROV_NAME, BUCKET=LOCATION,$
TYPE=DATA, COLUMN= MDN.STATE_PROV_POPULATION, BUCKET = COLOR,$
*GRAPH_JS
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
            },

   "overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],
"controls": [
{"control": "L.Control.Layers"},
{
"control": "L.Control.Scale",
"options": {
"imperial": true,
"metric": true
}
}
],
```

```
"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 17,
"minZoom": 0,
"attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
"url": function(){return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
},
"legend":{"visible":true},
"title": {"visible": true, "text": "United States Population in 2010"},

"colorScale": {"colors": ["purple", "lavender"] },
"htmlToolTip": {"enabled": true, "snap": true},
"series": [{"series": "reset", "tooltip":"auto"}]  ,
*END
 ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

バブルマップでのデータテキストラベルの表示

バブルマップのデータテキストラベルは、"dataLabels": "visible" プロパティを true に設定した場合に表示されます。

```
"series": [{"series":0,
               "dataLabels": {"visible": true}
                         }
                        ],
```

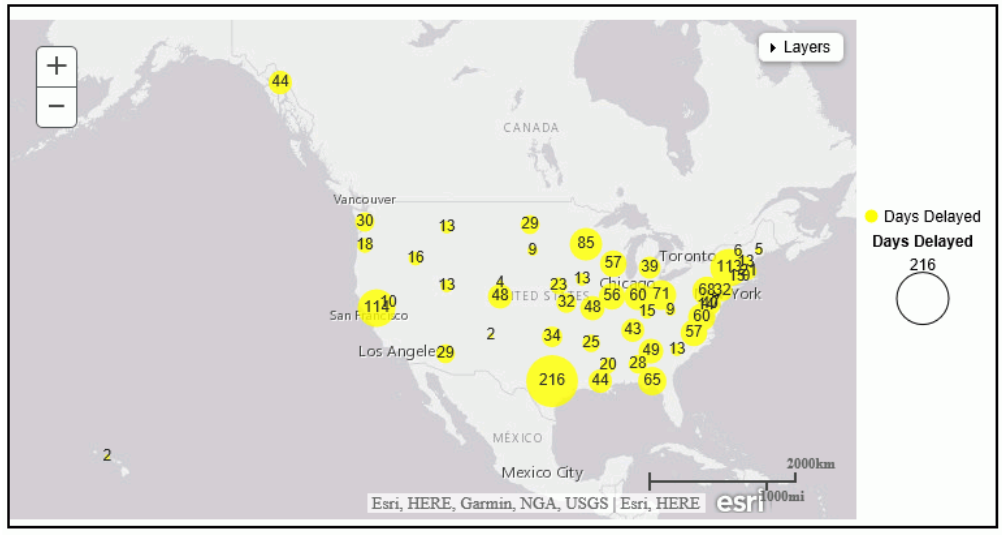
例　ESRI バブルマップでのデータテキストラベルの表示

次のリクエストは、Reporting Server 構文を使用してバブルマップを生成し、データテキストラベルを表示します。

```
DEFINE FILE WF_RETAIL_LITE
GEOPOINT/A200 = GIS_POINT('4326', STATE_PROV_CAPITAL_LONGITUDE,
STATE_PROV_CAPITAL_LATITUDE);
END
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY GEOPOINT
WHERE COUNTRY_NAME EQ 'United States' AND STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=DAYSDELAYED , BUCKET=size, $
*GRAPH_JS_FINAL
"bubbleMarker": {"maxSize": "10%"},
"series": [{"series":0, "color": "yellow", "dataLabels":{"visible":true}}],
```

```
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-geometry": {
                       "map_by_field": "GEOPOINT"
                                            }
                                      }
        }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                          }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

注意：グラフエンジン構文を使用して、データテキストラベルを表示することもできます。

例　Leaflet バブルマップでのデータテキストラベルの表示

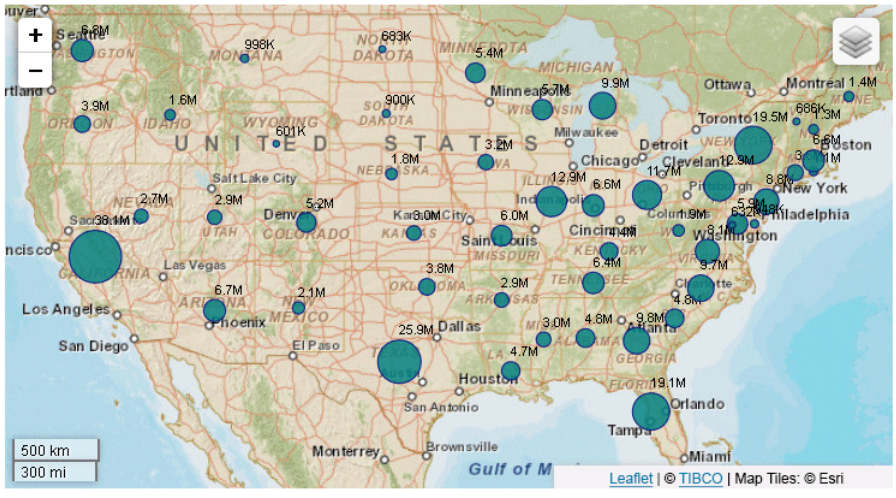
次のリクエストは、バブルマップを生成し、データテキストラベルを表示します。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MDN.STATE_PROV_POPULATION
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
WHERE TIME_YEAR EQ 2009
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
type=data, column=n1, bucket=latitude,$
type=data, column=n2, bucket=longitude,$
type=data, column=n3, bucket=size,$
*GRAPH_JS
 "series": [{"series":0, "color": "teal", "border":{"color": "navy", "width": 1}}],
"dataLabels": {
"visible": true
},
```

```
"legend": {"visible":false},
"bubbleMarker": {"maxSize": "10%" },
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
            },"overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){ return tdgchart.getScriptPath() + 'map/US.json'}
}],
  "controls": [
    {"control": "L.Control.Layers"},
    {
     "control": "L.Control.Scale",
     "options": {
      "imperial": true,
      "metric": true }
    }
   ],
 "baselayers": [{
   "title": "ArcGIS_World_Street_Map",
   "layerInfo": {
     "maxZoom": 17,
     "minZoom": 0,
      "attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
  "url": function(){return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

コロプレスマップでのデータテキストラベルとしての地名表示

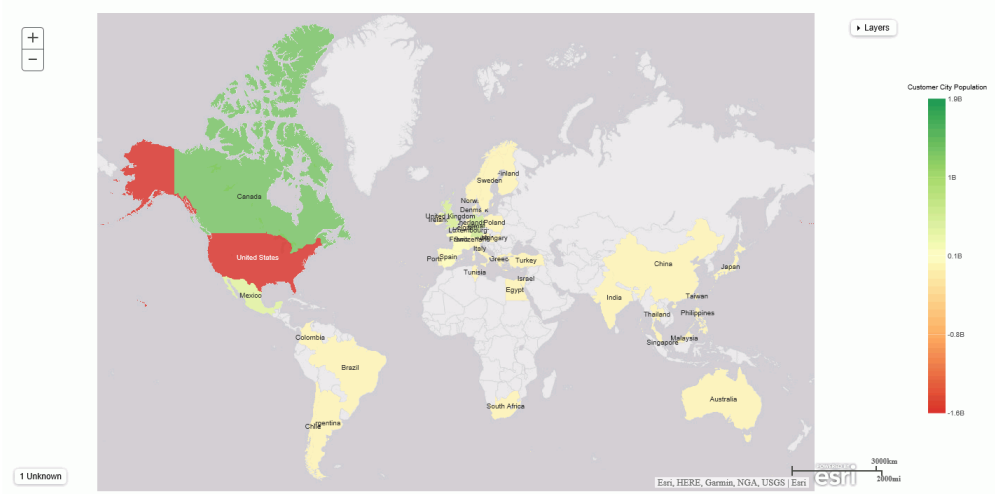
dataLabels オブジェクトは、"content": "name" プロパティを使用して地名を表示することができます。

例　ESRI マップグラフでのデータテキストラベルとしての地名表示

次のリクエストは、Reporting Server 構文を使用し、dataLabels オブジェクトの "content": "name" プロパティを適用して、国名をデータテキストラベルとして表示します。

```
DEFINE FILE wf_retail_lite CITY_POPULATION/D21 = CITY_POPULATION; END
GRAPH FILE wf_retail_lite
SUM CITY_POPULATION
BY COUNTRY_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=CITY_POPULATION, BUCKET=color, $
*GRAPH_JS_FINAL
"series": [{"series":0, "dataLabels":{"visible":true, "content": "name"}}],
"extensions": {
  "com.esri.map": {
      "overlayLayers":
      [
        {
         "ibiDataLayer": {
                  "map-metadata": {
                       "map_by_field": "COUNTRY_NAME"
                                            }
                                     }
         }
      ],
      "baseMapInfo":
         {
          "customBaseMaps":
           [
               {
                "ibiBaseLayer": "gray"
                }
            ]
          }
                        }
                    }
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

注意：グラフエンジン構文を使用して、地名をデータテキストラベルとして表示することもできます。

例　Leaflet マップグラフでのデータテキストラベルとしての地名表示

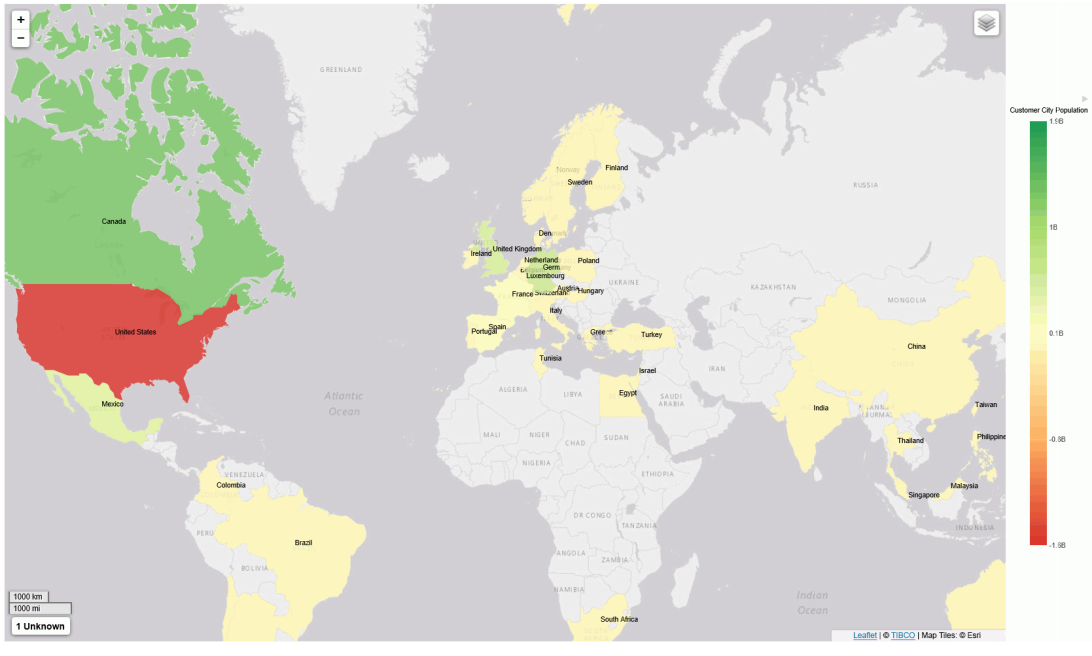
次のリクエストは、dataLabels オブジェクトの "content": "name" プロパティを使用して、国名をデータテキストラベルとして表示します。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
DEFINE FILE wf_retail_lite CITY_POPULATION/D21 = CITY_POPULATION; END
GRAPH FILE WF_RETAIL_LITE
SUM CITY_POPULATION
BY COUNTRY_NAME
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=N1, BUCKET=location, $
TYPE=DATA, COLUMN=N2, BUCKET=color, $
*GRAPH_JS_FINAL
"dataLabels": {"visible": true, "content": "name"},
"mapProperties": {
    "leaflet": {
      "baselayers":             [
                {
         "layerInfo": {
            "maxZoom": 16,
            "attribution": function() {  return "&|copy; <a target='_blank'
href='http://www.tibco.com'>TIBCO</a> | " +
               "Map Tiles: &|copy; Esri"; }
                  },
                "title": "ArcGIS_World_Street_Map",
                "url": function() { return 'https://services.arcgisonline.com/ArcGIS/
rest/services/Canvas/World_Light_Gray_Base/MapServer/tile/{z}/{y}/{x}'; }
                }
            ],
```

```
        "overlayLayers":            [
                {
                 "dataLookup": "properties.country_name",
                 "layerInfo": {
                     "type": "countries"
                    },
                "type": "tdg",
                "title": "World",
                "url": function() {  return tdgchart.getScriptPath() + 'map/
world.json' }
                }
            ]
    }
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

14

ユーザ独自グラフタイプのグラフライブラリへの追加

WebFOCUS は、組み込みグラフのリストに新しいカスタムグラフタイプを追加する機能をサポートします。これらのカスタムグラフタイプは、拡張機能 (エクステンション) またはプラグインと呼ばれます。拡張機能は、WebFOCUS 外部のリソースにアクセスするコードブロックです。この章では、拡張機能の構造およびユーザ独自のグラフタイプを作成し、グラフライブラリに追加するのに必要な手順について説明します。

トピックス

- グラフ拡張機能の概要
- グラフ拡張機能の作成
- グラフ拡張機能 API の使用
- グラフ拡張機能のインストール
- ibi WebFOCUS リクエストでの拡張機能の使用

グラフ拡張機能の概要

グラフ拡張機能は、JavaScript で記述されます。グラフの視覚化部分は、HTML、グラフキャンバス、SVG で描画することができます。拡張機能には、外部の CSS および JS ライブラリ (d3 など) を含めることができます。これらは、ほぼすべてのグラフ設計で使用可能です。WebFOCUS Extension API は、新しい、詳細グラフタイプのみに限定されています。既存のグラフタイプに機能を追加することはできません。また、WebFOCUS の構成要素を変更したり、ユーザ独自のグラフに割り当てられたグラフ領域外に拡張することもできません。

この章では、グラフ拡張機能の記述、構成、インストールの手順についての概要を説明します。

WebFOCUS の拡張機能は、WebFOCUS インストールの [web_resource] フォルダ下の [extensions] フォルダに格納します。デフォルト設定では、次のパスです。

```
c:¥ibi¥install_dir¥config¥web_resource¥extensions
```

説明

`install_dir`

WebFOCUS インストールのディレクトリです。たとえば、デフォルト設定では WebFOCUS バージョン 8.2 のインストールディレクトリは WebFOCUS82 です。

拡張機能フォルダには、グラフ拡張機能のサンプルがすでにいくつかインストールされています。これらのコード、構造、および WebFOCUS ツールを使用したアクセス方法についてはここで参照することができます。

注意：拡張機能をインストールするユーザは、グラフ拡張機能の生成に必要な JavaScript コードの記述についての知識が必要です。

グラフ拡張機能の作成

ここでは、拡張機能作成のための構築サイクルおよび拡張機能の構造と構成要素の概要について説明します。

参照 拡張機能記述の構築サイクル

拡張機能の作成には多くの場合、コードの記述、実行、デバッグの手順が含まれます。

ユーザ独自の拡張機能の properties.js ファイルに変更を加える場合は、WebFOCUS のキャッシュをクリアし、これらの変更を認識させる必要があります。WebFOCUS 管理コンソールの [キャッシュのクリア] リンクを使用してキャッシュをクリアします。

ユーザ独自の拡張機能の .js コードを変更する場合は (例、com.ibi.simple_bar.js)、WebFOCUS に変更を加える必要はありません。新しい JavaScript ファイルのダウンロードを確認するためのみ、ブラウザキャッシュのクリアが必要です。ユーザ独自の拡張機能に含まれた追加の .js ファイルを変更する場合も同様です。

参照 拡張機能の構造

Simple Bar 拡張機能の例は、拡張機能内の必須ファイルとオプションファイルを示し、これらのファイルがどのようにレイアウトされるかを示します。

テキストエディタで com.ibi.simple_bar および com.ibi.simple_bar.js を開き、拡張機能の正確な記述を参照することができます。

拡張機能 ID (ext_id) は com 形式の文字列です (your_company.extension_name)。ext_id はすべて小文字で表記し、文字、数字、アンダースコア (_)、ピリオド (.) のみを使用することができます。拡張機能のファイルはすべて ext_id の名前が付けられたフォルダに格納されます。拡張機能のコアは ext_id.js の名前が付けられたファイルに格納されます。このファイルには、WebFOCUS の新しいグラフタイプとして拡張機能を生成するコードが含まれます。

properties.json ファイルは、ユーザ独自の拡張機能を構成し、WebFOCUS でこれを実行します。このファイルには、ユーザ独自の拡張機能を WebFOCUS のユーザインターフェースに含めるために必要なメタデータのすべて、およびエンドユーザに公開する全プロパティのリストが含まれます。これにより、エンドユーザは拡張機能の動作をカスタマイズすることができます。

拡張機能フォルダには、オプションでそのほかに外部 CSS およびライブラリリソースのフォルダも含めることができます。ユーザ独自の拡張機能が追加の CSS または JavaScript ライブラリファイルを使用する場合、css または lib などの専用フォルダにこれらのリソースを整理、保存することができます。外部リソースは、ユーザ独自の拡張機能の ext_id.js ベースファイル内に構成され、ロードされます。

グラフ拡張機能 API の使用

グラフ拡張機能 API が提供する例はすべて、com.ibi.simple_bar.js で参照することができます。これらは主に、グラフの描画と拡張機能の構成の 2 つの部分に分けられます。

拡張機能 API バージョン 2 を使用する場合は、データバケットエントリに関連するフィールド名および数値フォーマットを取得することができます。API 2 を使用し、バケットエントリのフィールド名または数値フォーマットを取得するには、拡張機能で、properties.json の "info" ブロックに "implements_api_version" エントリを使用し、拡張機能 API バージョン 2 の実装を宣言する必要があります。

```
{
    "info": {
        "implements_api_version": "2.0"
            }
}
```

グラフの描画

拡張機能 API は、3 つのエントリポイントを提供します。ユーザは独自の JavaScript コールバック関数を定義することで、必要に応じてこれらを使用することができます。これらは、config オブジェクトの一連のプロパティに渡されます。プロパティには、描画処理全体で使用可能なプロパティと、レンダーコールバックでのみ使用可能なものがあります。

参照　グラフ描画のコールバック関数

次の 3 つの JavaScript コールバック関数を定義することができます。常に必要な関数は renderCallback のみです。

- **initCallback(successCallback, config)** - オプション。この関数は、ライブラリのロード時に 1 回だけエンジンに呼び出され、document.onload 初期化コードを実行する方法を提供します。この関数は、successCallback に渡されます。successCallback は、初期化コードが正常に実行された場合は true、正常に実行されなかった場合は false で呼び出します。successCallback(false) を呼び出した場合、拡張機能のインタラクティブな操作は終了し、拡張機能はブランクページとして表示されます。
- **preRenderCallback(config)** - オプション。この関数は拡張機能が生成されるたびに呼び出されます。これが生成処理全体の最初の手順となります。その後のグラフ描画に影響を与える内部グラフプロパティの検証、変更または上書きはここで行います。
- **renderCallback(config)** - 必須。この関数には、実際にカスタムグラフを描画するコードがすべて含まれます。config オブジェクトには、次のセクションで説明するプロパティが含まれます。

3 つのエントリポイントのコールバック関数はそれぞれ、config オブジェクトに渡されます。config オブジェクトには一連のよく使用されるプロパティが含まれます。

例 renderCallback 関数の例

次の renderCallback コードの例は、Simple Bar 拡張機能を生成します。

```
function renderCallback(renderConfig) {
  var chart = renderConfig.moonbeamInstance;
  var props = renderConfig.properties;
  var container = d3.select(renderConfig.container)
   .attr('class', 'com_ibi_chart');
  var data = renderConfig.data;
  if (renderConfig.dataBuckets.depth === 1) {
   data = [data];
  }

  var seriesCount = data[0].length;
  var seriesLabels = data[0].map(function(el){return el.labels;});
  data = d3.transpose(data).map(function(el, idx) {
   el = el[0];
   var v = Array.isArray(el.value) ? el.value : [el.value];
   var y0 = 0;
   return v.map(function(d, s) {
    return chart.mergeObjects(d, {y0: y0, y1: y0 += d, seriesID: s, value:
d, labels: seriesLabels[idx]});
   });
  });

  var w = renderConfig.width;
  var h = renderConfig.height;
  var x =
d3.scale.ordinal().domain(pv.range(seriesCount)).rangeRoundBands([0, w],
0.2);
  var ymax = d3.max([].concat.apply([], data), function(d){return d.y1;});
  var y = d3.scale.linear().domain([0, ymax]).range([25, h]);
  var svg = container.selectAll("g")
   .data(data)
   .enter().append('g')
   .attr('transform', function(d, i){return 'translate(' + x(i) + ', 0)';});
```

```
  svg.selectAll("rect")
   .data(function(d){return d;})
   .enter().append('rect')
   .attr("width", x.rangeBand())
   .attr("y", function(d) {return h - y(d.y1);})
   .attr("height", function(d){return y(d.y1) - y(d.y0);})
   .attr('tdgtitle', function(d, s, g) {
    // To support tooltips, each chart object that should draw a tooltip
must
    // set its 'tdgtitle' attribute to the tooltip's content string.

    // Retrieve the chart engine's user-defined tooltip content with
getToolTipContent():
    // 's' and 'g' are the series and group IDs for the riser in question.
    // 'd' is this riser's individual datum, and seriesData is the array of
data for this riser's series.
    var seriesData = chart.data[s];
    var tooltip = renderConfig.modules.tooltip.getToolTipContent(s, g, d,
seriesData);
    // getToolTipContent() return values:
    //  - undefined: do not add any content to this riser's tooltip
    //  - the string 'auto': you must define some 'nice' automatic tooltip
content for this riser
    //  - anything else: use this directly as the tooltip content
    if (tooltip === 'auto') {
     if (d.hasOwnProperty('color')) {
      return 'Bar Size: ' + d.value + '<br />Bar Color: ' + d.color;
     }
     return 'Bar Size: ' + d.value;
    }
    return tooltip;
   })
```

```
   .attr('class', function(d, s, g) {
    // To support data selection and tooltips, each riser must include a
class name with the appropriate seriesID and groupID
    // Use chart.buildClassName to create an appropriate class name.
    // 1st argument must be 'riser', 2nd is seriesID, 3rd is groupID, 4th
is an optional extra string which can be used to identify the risers in
your extension.
    return chart.buildClassName('riser', s, g, 'bar');
   })
   .attr('fill', function(d) {
    // getSeriesAndGroupProperty(seriesID, groupID, property) is a handy
function
    // to easily look up any series dependent property.  'property' can be
in
    // dot notation (eg: 'marker.border.width').
    return chart.getSeriesAndGroupProperty(d.seriesID, null, 'color');
   });

  svg.append('text')
   .attr('transform', function(d) {return 'translate(' + (x.rangeBand() /
2) + ',' + (h - 5) + ')';})
   .text(function(d, i){return seriesLabels[i];})

  renderConfig.modules.tooltip.updateToolTips();  // Tell the chart engine
your chart is ready for tooltips to be added
  renderConfig.modules.dataSelection.activateSelection();  // Tell the
chart engine your chart is ready for data selection to be enabled
 }
```

参照　常に使用可能なプロパティ

次のプロパティは常に使用可能です。

プロパティ名	説明
moonbeamInstance	生成されるグラフインスタンス
data	生成されるデータセット
properties	ユーザが設定する拡張機能のプロパティブロック
dataBuckets	オプションのカスタムデータバケット。詳細は、1034 ページの 「 dataBuckets ブロック 」 を参照してください。

参照　レンダーコールバックでのみ使用可能なプロパティ

次のプロパティは、レンダーコールバックでのみ使用可能で、カスタムグラフ生成コード (renderCallback) によって使用されます。

プロパティ名	説明
width	拡張機能が生成するコンテナの幅 (ピクセル単位)
height	拡張機能が生成するコンテナの高さ (ピクセル単位)
containerIDPrefix	拡張機能が生成する DOM コンテナの ID。拡張機能が生成するすべての ID の先頭にこれを追加し、拡張機能の複数コピーが 1 ページに表示されるようにします。
container	拡張機能が生成する DOM ノード。選択した containerType 拡張機能設定によって、HTML の div 要素か SVG の g 要素のいずれかになります。
rootContainer	生成される特定のグラフエンジンインスタンスを含む DOM ノードです。

グラフ拡張機能の構成

拡張機能は次の 2 つで構成されます。

- グラフエンジン構成。WebFOCUS のグラフエンジンおよびグラフキャンバスと連動するように拡張機能を構成します。拡張機能構成のこの部分は、グラフ描画機能に渡される config オブジェクトで定義されます。
- グラフインターフェース構成。ユーザインターフェースおよびグラフ属性カテゴリのグラフタイプの選択オプションと連動します。拡張機能構成のこの部分は、properties.json ファイルで定義されます。

グラフエンジン構成の config オブジェクト作成

拡張機能を構成するには、拡張機能に固有のすべての情報を含む config オブジェクトを作成し、作成した拡張機能を拡張機能 API に登録します。

参照

拡張機能の config オブジェクト作成

下表は、config オブジェクトの必須プロパティとオプションプロパティを示しています。

プロパティ名	説明
id	拡張機能 ID です。詳細は、1022 ページの「 拡張機能の構造 」 を参照してください。
name	ユーザインターフェースに表示されるグラフタイプの名前です。
description	ユーザインターフェースに表示されるグラフタイプの説明です。
containerType	'html' または 'svg' (デフォルト値) のいずれかです。
initCallback	オプション関数です。initCallback 関数についての詳細は、1024 ページの 「 グラフ描画のコールバック関数 」 を参照してください。
preRenderCallback	オプション関数です。preRenderCallback 関数についての詳細は、1024 ページの「 グラフ描画のコールバック関数 」を参照してください。
renderCallback	必須関数です。renderCallback 関数についての詳細は、1024 ページの 「 グラフ描画のコールバック関数 」 を参照してください。
resources	オプションです。この拡張機能に必要な追加の外部リソースです (CSS および JS)。

例　config オブジェクトの例

次のコードは、Simple Bar 拡張機能で使用される config オブジェクトの例です。

```
var config = {
    id: 'com.ibi.simple_bar',     // string that uniquely identifies this
extension
    containerType: 'svg',  // either 'html' or 'svg' (default)
    initCallback: initCallback,  // Refers to your init callback fn
(optional)
    preRenderCallback: preRenderCallback,  // Refers to your preRender
callback fn (optional)
    renderCallback: renderCallback,  // Refers to your render fn (required)
    resources:  {  // Additional external resources (CSS & JS) required by
this extension (optional)
        script: ['lib/d3.min.js'],
        css: ['css/extension.css']
    },
   }
```

参照　拡張機能の登録

拡張機能を WebFOCUS Extension API に登録するには、以下を呼び出します。

```
tdgchart.extensionManager.register(config);
```

参照　拡張機能構築のヒント

拡張機能を構築する最も簡単な方法は、Simple Bar のサンプルを複製し、これに変更を加えることです。新しい拡張機能の ID が com.foo.bar であると仮定します。

1. ルートフォルダの名前を com.foo.bar に変更します。com.ibi.simple_bar.js の名前を com.foo.bar.js に変更します。
2. com.foo.bar.js で、3 つのコールバック関数の内部コンテンツを削除します。
3. com.foo.bar.js で、config の各プロパティのエントリを拡張機能の要件に合わせて変更します。
4. 必要な外部リソースがあれば、lib および css に追加し、com.foo.bar.js で config.resources を設定してこれらをロードします。
5. com.foo.bar.js で renderCallback 関数を実行し、拡張機能を描画します。

グラフインターフェースの構成

各拡張機能には必ず properties.json ファイルが含まれ、これによりユーザインターフェースを描画する際に WebFOCUS に必要な情報が定義されます。

properties.json ファイルは、以下のブロックで構成されます。

- **info ブロック**　詳細は、1033 ページの 「 info ブロック 」 を参照してください。
- **properties ブロック**　詳細は、1034 ページの 「 properties ブロック 」 を参照してください。
- **propertyAnnotations ブロック**　詳細は、1034 ページの 「 propertyAnnotations ブロック 」 を参照してください。
- **dataBuckets ブロック**　詳細は、1034 ページの 「 dataBuckets ブロック 」 を参照してください。
- **translations ブロック**　詳細は、1036 ページの 「 translations ブロック 」 を参照してください。

例　properties.json ファイルの例

次の例は、Simple Bar 拡張機能の properties.json ファイルです。

```
{
    // Define some general extension configuration options
    "info": {
     "version": "1.0",  // version number of your extension.
     "implements_api_version": "2.0",  // version number of the WebFocus
    // API used by your extension.
      "author": "TIBCO",
      "copyright": "TIBCO",
      "url": "https://github.com/ibi/wf-extensions-chart/tree/master/simple_bar%20example",
      "icons": {
      "medium": "icons/medium.png"  // Reference to an image in the
    // extension, used in the WF chart picker
        }
    },
    // Define any properties of your extension that end user may want to change.
    "properties": {
        "exampleProperty": 50
    },
    // Define the possible values for each property in 'properties'.
    "propertyAnnotations": {
        "exampleProperty": "number"
    },
    // Define the available data buckets drawn in WF's 'Query' data bucket tree.
    "dataBuckets":  {
        // Choose whether or not to reuse existing WF data buckets.  All optional.
        "tooltip": false,
        "series_break": true,
```

```
        // Define your own custom data buckets.  Optional
        "buckets": [
            {
            "id": "value",
            "type": "measure",
            "count": {"min": 1, "max": 5}
           },
           {
           "id": "labels",
           "type": "dimension",
           "count": {"min": 1, "max": 5}
           }
           ]
         },
    // Define the set of labels used in the WF interface for buckets and
    // chart type picker.
    "translations": {
        "en": {
            "name": "My Simple Bar Chart",
            "description": "This chart is just a simple bar chart, nothing to see here.",
            "icon_tooltip": "This extension does ...",
            "value_name": "Value Bucket",
            "value_tooltip": "Drop a measure here",
            "labels_name": "Label Bucket",
            "labels_tooltip": "Drop a dimension here"
        },
        "fr": {
            "name": "Un Bar Chart tres simple",
            "description": "C'est un Bar Chart vraiment simple",
            "icon_tooltip": "This extension does ...",
            "value_name": "Value Bucket",
            "value_tooltip": "Drop a measure here",
            "labels_name": "Label Bucket",
            "labels_tooltip": "Drop a dimension here"
        }
    }
}
```

参照

info ブロック

このブロックは、グラフアイコンを含むファイルなど、いくつかの汎用構成オプションを定義します。

icons

このプロパティは、WebFOCUS ツール内に拡張機能を表示するアイコンとして使用されるファイルを定義します。拡張機能には、さまざまなサイズの複数のビットマップアイコンを含めることも、拡張性のある単一の SVG アイコンを含めることもできます。SVG アイコンには、ハードコードされた "width" または "height" の属性を含めることはできません。以下は、1 つのアイコンオブジェクトを含む info ブロックの構文です。

```
{
    "info": {
        "icons": {
            "size": "icons/filename"
        }
    }
}
```

説明

`"size"`

各アイコンファイルのサイズです。各アイコンファイルは、サイズに従って "icons" 下のエントリとして列挙します。サイズは次のいずれかです。

- "small" - 32x32 ピクセルのアイコン
- "medium" - 72x72 ピクセルのビットマップアイコン、または SVG アイコン

`"icons/filename"`

アイコンを含むファイルの名前です。

以下はその例です。

```
{
    "info": {
        "icons": {
            "medium": "icons/my_svg_icon.svg"
        }
    }
}
```

参照 properties ブロック

このブロックは、エンドユーザが変更することのできる拡張機能のプロパティを定義します。ユーザは、WebFOCUS グラフプロシジャの GRAPH_JS ブロックでこれらのプロパティを変更することができます。

参照 propertyAnnotations ブロック

このブロックは、properties ブロックのコンテンツを確認します。properties のすべてが propertyAnnotations ブロックに表示されます。properties のオブジェクト以外の (リーフ) プロパティは、"str"、"bool"、"number" のいずれかで表記されます。

"propertyAnnotations" ブロックの各プロパティの値は、WebFOCUS の [拡張のプロパティ] パネルに表示されるコントロールのタイプを指定します。これは、プロパティ値として直接設定することができます。また、プロパティ内の "typeAnnotation" プロパティを設定して、コントロールに追加のプロパティを指定することもできます。デフォルト設定では、"str" の値がテキストボックス、"bool" がチェックボックス、"color" がカラーパレット、"number" がスピナを生成します。

"typeAnnotation" が "stir" に設定されている場合、"stringValues" プロパティに値の配列を追加し、これらの値を使用したドロップダウンリストを作成することができます。
"typeAnnotation" が "number" に設定されている場合、デフォルトではスピナの値の間隔は 1 に設定され、最大値および最小値は設定されません。"numericRange" プロパティを使用すると、配列 (例、[0,100]) として最小値と最大値を設定できます。また、"numericStep" プロパティを使用すると、各値の間隔を設定できます。これら両方のプロパティが設定されている場合、デフォルトコントロールはスライダになります。スピナを使用する場合は、プロパティ設定で "uiType": "spinner" を指定します。たとえば、次の "propertyAnnotations" ブロックの一連のプロパティは、増分間隔が 0.05 のスピナを使用してパディングの値を 0 から 1 までに設定するコントロールを作成します。

```
"padding": {
    "typeAnnotation": "number",
    "numericRange": [0,1],
    "numericStep": 0.05,
    "uiType": "spinner"
},
```

参照 dataBuckets ブロック

このブロックは、グラフ作成時に WebFOCUS ユーザインターフェースの [クエリ] ウィンドウに表示されるグラフ属性カテゴリのセットを定義します。

API バージョン 2 を使用する場合、dataBuckets.buckets はオブジェクトではなく配列になります。この配列の各エントリは、1 つのデータバケットのコンテンツを表します。id プロパティはバケットを識別し、fields 配列がこのバケット内のエントリ数および各エントリの固有情報 (タイトル、フィールド名、数値フォーマット) を指定します。

dataBuckets オブジェクトには、getBucket() という名前のメソッドが含まれます。このメソッドでは、バケットの名前を渡すと、このバケットのコンテンツが返されます。

バケットには、組み込みバケットとカスタムバケットの 2 種類があります。組み込みバケットは、既存の WebFOCUS データバケットロジックを再利用する簡単な方法を提供します。現在使用できるのは tooltip と series_break の 2 つの組み込みバケットです。これらのバケットを使用するには、関連する dataBuckets プロパティを true に設定します。

API バージョン 1.0 では、dataBuckets ブロックに次のオブジェクトが含まれます。

bucket - 各バケットブロックが 1 つのカスタムグラフ属性カテゴリを定義します。カスタムバケットにはそれぞれ次のプロパティが必要です。

- **id** - このプロパティは dataArrayMap および data プロパティに対応し、カスタムグラフの render 関数に渡されます。
- **type** - このプロパティは、このバケットが受容するデータフィールドのタイプを定義します ("measure"、"dimension"、"both")。
- **count** - count.min および count.max で構成されます。このバケットが受容可能なフィールドの最小数と最大数をそれぞれ定義します。最小値がゼロ (0) の場合、このバケットがオプションであることを示します。

API バージョン 2.0 では、dataBuckets ブロックに次のオブジェクトが含まれます。

- **depth** - バケット配列のバケット数を指定します。
- **buckets** - すべてのバケットのすべてのフィールドのプロパティを指定します。
  - **id** - このプロパティは dataArrayMap および data プロパティに対応し、カスタムグラフの render 関数に渡されます。
  - **fields** - 各バケットのフィールド配列です。各フィールドのフィールドタイトル、名前、数値フォーマットが定義されます。

```
{"title": "fieldtitle", "fieldName": "fieldname", "numberFormat":
"format"}
```

API バージョン 2 では、拡張機能の各レンダーコールバックに渡される renderConfig.dataBuckets のコンテンツに、フィールドタイトルとフォーマットの情報が追加されます。次の例では、2 つのバケットが使用されています。labels バケットには、CAR.ORIGIN.COUNTRY という 1 つのフィールドが含まれています。このフィールドのタイトルは、COUNTRY です。value バケットには、CAR.SALES (タイトルは SALES) および CAR.BODY.DEALER_COST (タイトルは DEALER_COST) の 2 つのフィールドが含まれています。

```
"dataBuckets": {
   "getBucket(bucketName)",
   "depth": 2,
 "buckets": [
  {
   "id": "labels",
   "fields": [
   {"title": "COUNTRY", "fieldName": "CAR.ORIGIN.COUNTRY"}
            ]
  },
  {
  "id": "value",
  "fields": [
   {"title": "SALES", "fieldName": "CAR.SALES",
    "numberFormat": "#,###.00"},
   {"title": "DEALER_COST",
    "fieldName": "CAR.BODY.DEALER_COST",
    "numberFormat": "#,###"}
         ]
       }
   ]
}
```

参照 translations ブロック

このブロックは、WebFOCUS インターフェースで描画されるすべてのラベルの多言語翻訳を定義します。translation オブジェクトには、拡張機能がサポートする各言語に対し 1 つのプロパティが含まれます。2 バイトの ISO-639 言語ロケール文字列で指定します。

拡張機能のデータアクセス

拡張機能が生成されるたびに、renderConfig.data 引数を使用した拡張機能のレンダーコールバックが現在のデータセットに渡されます。データセットの全体構造は、properties.json ファイルにリストされた一連のバケットによって定義されますが、特定のデータコンテンツは、ユーザが各バケットに追加したデータフィールドによって定義されます。

拡張機能のバケットの定義と使用

データセットは、レンダーコールバックの最初の引数の data プロパティを使用して拡張機能に渡されます。この名前は通常 renderConfig です。各バケットに現在設定されているフィールドについての詳細は、renderConfig.dataBuckets に格納されています。

データセットは、オブジェクトの配列として JavaScript で表記されます。1 つの拡張機能がカスタムバケットのみ定義する場合は、データセットはフラットなオブジェクトの配列になります。1 つの拡張機能が複数の組み込みバケットを使用する場合は、データセットには深くネストされた配列の配列が含まれる可能性があります。renderConfig.dataBuckets.depth プロパティは、現在のデータセットの配列ディメンションの数に設定されます。

カスタムバケット

データ配列の最も内側のオブジェクト (datum と呼ばれる) は、1 つのフィールドを含む各データバケットに対しプロパティを 1 つ指定します。各プロパティは、カスタムバケットの id となり、properties.json の dataBuckets.buckets セクションで定義されます。これらプロパティの値のタイプは、バケットのタイプによって異なります。ディメンションのバケットでは値が文字列になりますが、メジャーのバケットでは値は数値になります。バケットにフィールドが複数含まれる場合は、最も内側のオブジェクトそれぞれの関連するプロパティが、文字列または数値の配列になります。

組み込みバケット

拡張機能は、WebFOCUS にビルトインされた定義済みのバケットを使用することができます。これらのバケットは、データセット以外にも影響します。各バケットには特定のグラフエンジンプロパティも設定されており、バケットに関連する追加情報を渡します。

WebFOCUS の組み込みバケットは、標準バケットまたは分割バケットのいずれかです。

- 標準バケットは、カスタムバケットとまったく同じ動作をします。データセットはそのまま単一配列として残り、各 datum オブジェクトにはバケットの名前が付いた別のプロパティが含まれます。
- 分割バケットは、データセットを別のデータ配列に分割します。分割バケットが使用された場合は、それぞれの datum オブジェクトが datum オブジェクトの完全な配列に変換されます。各配列の datum オブジェクトの数は変わりませんが、配列または datum 配列の数は、区切りフィールド内のエントリ数と一致します。

分割バケットのタイプ

分割バケットには 2 つのタイプがあります。

- シリーズ分割バケットはデータセットを分割し、ユーザが選択したシリーズ区切りフィールドのエントリごとに 1 つの配列にします。シリーズ分割バケットは、グラフエンジンで定義されたシリーズ固有のプロパティを使用し、データ名はそれらシリーズ固有のプロパティに示されます。シリーズ分割フィールドの各エントリは、対応するシリーズ固有のプロパティオブジェクトをグラフエンジンに生成し、renderConfig.moonbeamInstance.getSeries (*x*) で取得可能です。この場合の *x* は、取得されるシリーズの整数です。getSeries は、色やラベルなど選択されたシリーズに固有のプロパティを備えたオブジェクトを返します。
- マトリックス分割バケットは、マトリックスグラフの列と行を定義するソートフィールドに使用されます。また、マトリックス分割バケットは、データセットに配列ディメンションを追加します。マトリックス分割バケットは、列と行のサブバケットに分割されます。行バケットか列バケットのいずれかにフィールドが含まれる場合、どちらか一方のマトリックスバケットがブランクであっても、データセットには 2 つのデータディメンションが含まれます。つまり、データセットは、行データも列データも含まないか、行データと列データの両方を含むかのいずれかで、片方だけということはありません。bucket.depth は常に 3 以上です。

ツールヒントバケット

ツールヒントバケットは分割バケットではないため、データセットに配列ディメンションを追加しません。ツールヒントバケットの動作はカスタムバケットに類似しています。内部の各 datum オブジェクトには tooltip という名前のプロパティが与えられ、ディメンションのタイプ文字列の値、メジャーの数、およびバケット内の複数フィールドの値の配列がこれに含まれます。

このバケットの利便性は、データセットにツールヒント固有のデータを含めることができるうえ、WebFOCUS でも各シリーズに有意義なツールヒントのコンテンツを生成できることです。このツールヒントのコンテンツは、WebFOCUS の組み込みグラフタイプのすべてで使用されるコンテンツと同じです。ツールヒントバケットを使用すると、拡張機能で各ツールヒントに追加するコンテンツを指定する必要がありません。

例　シリーズ分割バケット定義の例

ここでは以下のサンプルデータを使用します。

Car	Country	Seats
BMW	ドイツ	5
Audi	ドイツ	4
Peugeot	フランス	5
Alfa Romeo	イタリア	4
Maserati	イタリア	2
Toyota	日本	4

次のコードは、シリーズ分割バケットを定義します。

```
dataBuckets:
        series_break: true,
        buckets: [
            {id: "label", type: "dimension"},
            {id: "value", type: "measure"}
        ]
```

各バケットに割り当てられた次のフィールドについて考察します。

- "Country" は、"series_break" バケットに割り当てられます。
- "Car" は、"label" バケットに割り当てられます。
- "Seats" は、"value" バケットに割り当てられます。

renderConfig 関数では、renderConfig.data オブジェクトは以下のようになります。ここでは、Country の値はデータ配列の一部でなくなっています。ただし、Country の値を変更するたびに新しい配列が開始されます。

```
 [{labels: "PEUGEOT", value: 5}],
        [{labels: "ALFA ROMEO", value: 4}, {labels: "MASERATI", value: 2],
        [{labels: "TOYOTA", value: 4}],
        [{labels: "AUDI" ,value: 4}, {labels: "BMW", value: 5}]
```

renderConfig.dataBuckets オブジェクトは次のように定義されます。

```
renderConfig.dataBuckets = {
        depth: 2,
        series_break: {title: "Country"},
        buckets: {
            label: {title: "Car"},
            value: {title: "Seats"}
```

拡張機能の部分データおよび NULL データの処理

多くの場合、拡張機能を扱うエンドユーザは、拡張機能バケットのすべてに即時入力することはできません。拡張機能は、これらの部分データを正確に処理する必要があり、1 つまたは複数のバケットがブランクの場合にクラッシュすることがあってはなりません。そのため、どのバケットが入力済みであるか、renderConfig.dataBuckets を確認し、適切に処理することが重要です。

また、ディメンションとメジャーの組み合わせで値の一部が欠落するなど、データセットが不完全の場合も多くあります。これらのミッシング値は、配列 (datum オブジェクトではなく) 内の NULL エントリ、または完全にブランクの配列としてデータセット内に存在する可能性があります。これらの欠落データを検出、処理し、そのような欠落データに対し適切なビジュアライゼーションを生成することが重要です。

拡張機能のほとんどでは、生成前に入力すべきバケットの最小数が決められています。これらの最小数要件を定義するには、properties.json の各 dataBuckets.bucket エントリの count.min プロパティを使用します。すべてのバケットのフィールドが最小数を満たさない場合は、拡張機能の renderCallback が呼び出されません。その代わり、同じ拡張機能の noDataPreRenderCallback が呼び出されます。この場合、拡張機能は特別な no data モードで生成されます。このモードでは、拡張機能はグレースケールで生成され、主色として renderCallback.baseColor を使用します。これは、非常に簡素化された拡張機能のサンプル生成となります。

例

noDataPreRenderCallback 関数の例

次の noDataPreRenderCallback 関数は、Simple Bar サンプル拡張機能から抜粋したものです。

```
function noDataRenderCallback(renderConfig) {
  var grey = renderConfig.baseColor;
  renderConfig.data = [{value: [3, 3]}, {value: [4, 4]}, {value: [5, 5]},
{value: [6, 6]}, {value: [7, 7]}];
  renderConfig.moonbeamInstance.getSeries(0).color = grey;
  renderConfig.moonbeamInstance.getSeries(1).color =
pv.color(grey).lighter(0.18).color;
  renderCallback(renderConfig);
 }
```

グラフ拡張機能のインストール

1. ローカルマシンの WebFOCUS インストールの拡張機能フォルダを確認します。通常は次のフォルダに格納されています。

   ```
   C:¥ibi¥install_dir¥config¥web_resource¥extensions
   ```

   説明

   ```
   install_dir
   ```

   WebFOCUS インストールのディレクトリです。

   注意：GitHub サイトの WebFOCUS 拡張機能セクションには、一般に利用可能な拡張機能やサポートされる拡張機能のリストが示されています。これらをインストールするには、インストールする拡張機能をクリックします。次に、その拡張機能の ZIP ファイル (例、com.ibi.xyz.zip) を右クリックし、[リンクに名前を付けて保存] を選択します。

2. ダウンロードした ZIP ファイルを WebFOCUS 拡張機能フォルダに解凍します。たとえば、com.ibi.xyz.zip ZIP ファイルの場合、次のフォルダが作成されます。

   ```
   C:¥ibi¥install_dir¥config¥web_resource¥extensions¥com.ibi.xyz
   ```

   ユーザ独自の環境からカスタム拡張機能をインストールする場合は、これを WebFOCUS 拡張機能フォルダにコピーまたはダウンロードします。フォルダ名および拡張機能 ID には、サンプル拡張機能と同じ命名規則を使用します。

3. C:¥ibi¥install_dir¥config¥web_resource¥extensions¥html5chart_extensions.json を編集します。新規拡張機能のためにフォームに新しい行を追加します。

   ```
   "com.ibi.abc": {"enabled": true},
   ```

   説明

   ```
   abc
   ```

   拡張機能の名前です。

4. WebFOCUS 管理コンソールで、[キャッシュのクリア] をクリックします。これにより、WebFOCUS がすべての拡張機能のリロードを強制的に実行します。

以下は、html5chart_extensions.json の例です。

```
{
       "com.ibi.simple_bar": {enabled: true},
       "com.ibi.liquid_gauge": {enabled: false},
       "com.ibi.sankey": {enabled: true}
 }
```

注意：WebFOCUS 管理コンソールは、グラフ拡張機能インストールのユーザインターフェースを備えています。詳細は、『ibi™ WebFOCUS® セキュリティ管理ガイド』の第 3 章「WebFOCUS Client の構成」を参照してください。

参照 WebFOCUS Client 再インストール時のカスタムグラフタイプの保持

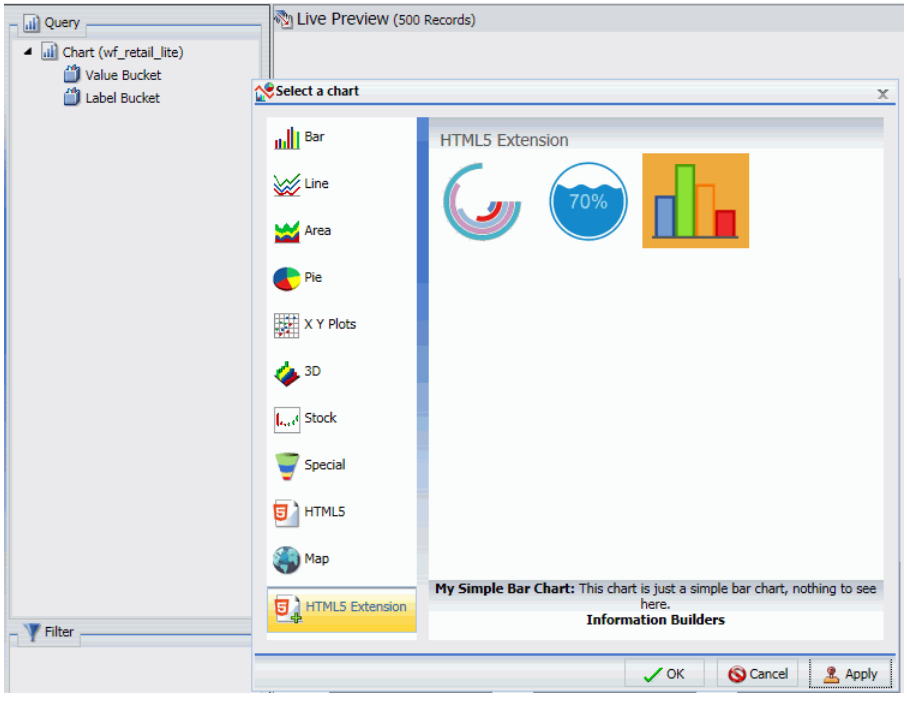
TIBCO WebFOCUS Client を再インストールすると、拡張機能フォルダは上書きされます。そのため、カスタムグラフ拡張機能をインストールしていた場合は、WebFOCUS Client の再インストール開始前にこれらを別の場所にコピーして保存し、WebFOCUS Client の再インストール終了後にこれらを拡張機能フォルダにコピーして戻す必要があります。

また、WebFOCUS Client の新しいバージョンでインストールされた新しい html5chart_extensions.json ファイルに、カスタム拡張機能のエントリをコピーする必要もあります。

注意：WebFOCUS の一部として同梱された拡張機能は自動的に再インストールされるため、保存する必要はありません。これらの拡張機能が強化された場合は、WebFOCUS Client の再インストール時に強化されたバージョンに自動的にアクセスできます。

ibi WebFOCUS リクエストでの拡張機能の使用

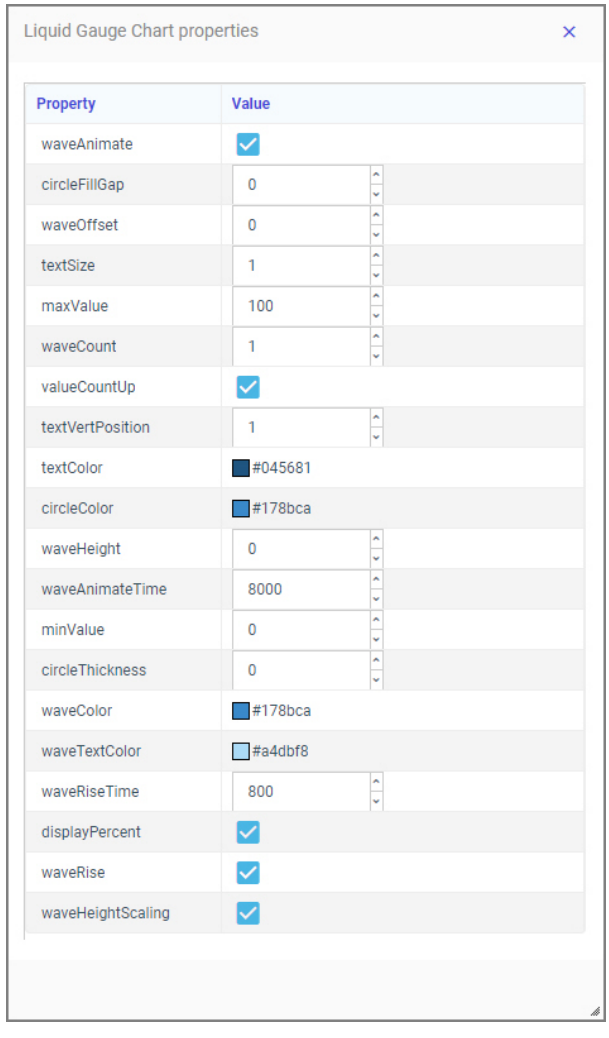
手順に従って拡張機能のインストールおよび構成が完了後、下図のように [その他] フォーマットカテゴリの [HTML5 拡張] グラフタイプとして、拡張機能を WebFOCUS ツールで使用することができます。



WebFOCUS を使用してグラフを作成する場合、下図のように、コンテンツの選択オプションを展開すると、[カスタム] カテゴリにグラフ拡張機能が表示されます。

拡張機能の dataBuckets オブジェクトで定義した属性カテゴリは、InfoAssist の [クエリ] ウィンドウまたは WebFOCUS の [表示] パネルで使用することができます。

また、WebFOCUS では、クイックアクセスメニューで [全般] を選択すると、propertyAnnotations オブジェクトで定義したプロパティが、[フォーマット] タブの [その他] セクションで使用できます。下図のように、[拡張のプロパティ] をクリックすると、[拡張のプロパティ] パネルが開き、使用可能なプロパティに変更を加えることができます。下図は、液体メータグラフ拡張機能のプロパティを示しています。



次の構文がプロシジャに追加されます。

❑ LOOKGRAPH 値は、EXTENSION です。

- 使用する実際の拡張機能は、スタイルシートの *GRAPH_JS ブロックの chartType プロパティで識別されます。以下は、液体メータグラフの例です。

```
*GRAPH_JS
    chartType: "com.ibi.liquid_gauge",
```

- カスタム属性カテゴリの名前にはそれぞれ、先頭に不等号 (> より大きい) が追加されます。以下は、1 つのラベルフィールドと 4 つの値フィールドを含む単純な棒グラフの例です。

```
TYPE=DATA, COLUMN=N1, BUCKET= >labels, $
TYPE=DATA, COLUMN=N2, BUCKET= >value, $
TYPE=DATA, COLUMN=N3, BUCKET= >value, $
TYPE=DATA, COLUMN=N4, BUCKET= >value, $
TYPE=DATA, COLUMN=N5, BUCKET= >value, $
```

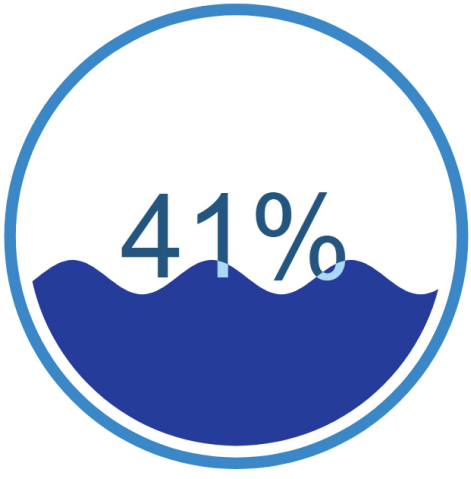
- グラフ拡張機能のプロパティは、"extensions": "extension_name" オブジェクトの下に記述されます。以下は、波の数を 3 、波の色を濃い青の色調に設定した液体メータグラフのプロパティを示しています。

```
"extensions": {
    "com.ibi.liquid_gauge": {
        "waveCount": 3,
        "waveColor": "#083f9e"
    }
}
```

以下は、液体メータグラフ拡張機能を使用した単純なリクエストの例です。

```
GRAPH FILE wf_retail_lite
SUM PCT.WF_RETAIL_LITE.WF_RETAIL_SALES.QUANTITY_SOLD
BY WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.BUSINESS_REGION
WHERE_GROUPED WF_RETAIL_LITE.WF_RETAIL_GEOGRAPHY_CUSTOMER.BUSINESS_REG
ION EQ 'North America'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH EXTENSION
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=N2, BUCKET= >value, $
*GRAPH_JS_FINAL
"chartType": "com.ibi.liquid_gauge",
"extensions": {
    "com.ibi.liquid_gauge": {
       "waveCount": 3,
       "waveColor": "#083f9e"
    }
}
*END
ENDSTYLE
END
```

グラフを実行します。下図は、出力結果を示しています。



A

TIBCO WebFOCUS グラフのパラメータ

ここでは、WebFOCUS グラフのパラメータおよびデフォルト値について説明します。

- JSON chartType プロパティによるグラフタイプの制御
- 複数グラフの作成
- SET パラメータによるグラフのカスタマイズ

JSON chartType プロパティによるグラフタイプの制御

この JSON プロパティは、グラフタイプを定義します。ただし、WebFOCUS LOOKGRAPH パラメータに対応する LOOKGRAPH 値が存在する場合は、LOOKGRAPH パラメータを使用してグラフタイプを設定することをお勧めします。LOOKGRAPH パラメータについての詳細は、40 ページの 「 LOOKGRAPH によるグラフタイプの制御 」 を参照してください。

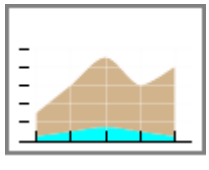
```
"chartType": "string"
```

説明

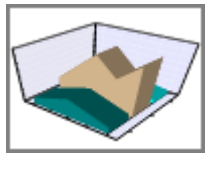
```
"chartType": "string"
```

グラフタイプを定義する文字列です。この文字列は、二重引用符 (") で囲みます。デフォルト値は "bar" です。

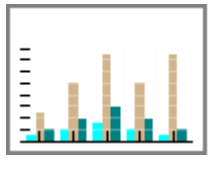
下図は、各グラフタイプと、有効な chartType 値を示しています。



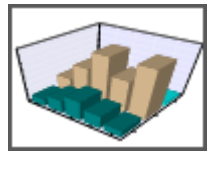
area



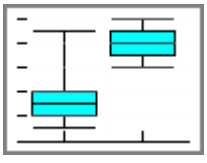
area3d



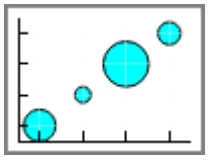
bar



bar3d

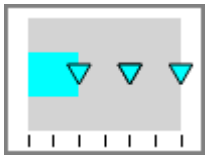


boxplot

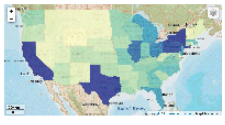


bubble

bubblemap



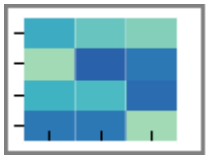
bullet



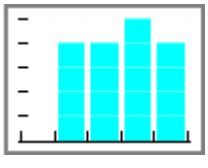
choropleth

funnel

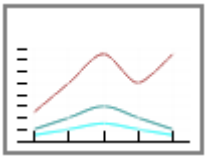
gauge



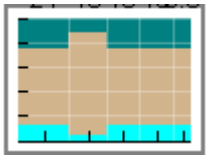
heatmap



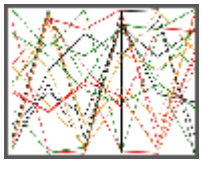
histogram



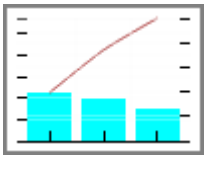
line



mekko

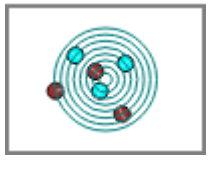


parabox

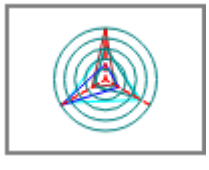


pareto

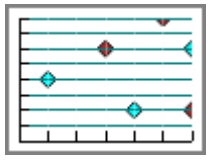
pie



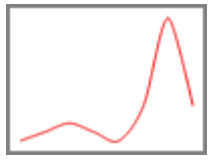
polar



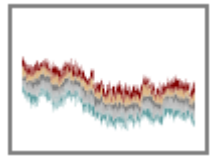
radar



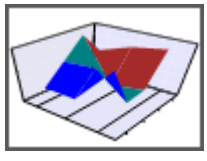
scatter



sparkline



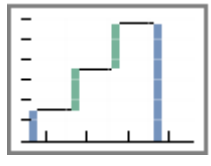
streamgraph



surface3d

tagcloud

treemap



waterfall

複数グラフの作成

2 つ目のソートディメンション (フィールド) を追加することで、複数のグラフを作成することができます。

デフォルト設定では、作成されるグラフ数は、ソート句 (BY、ACROSS) で指定したフィールドの値の個数によって決定されます。このデフォルト設定を変更するには、GRMERGE パラメータを使用します。

- GRMERGE OFF (デフォルト設定) では、リクエストに BY フィールドが 2 つ含まれる場合、最初の BY フィールドの値の個数と同数のグラフが生成されます。2 つ目の BY フィールドは、X 軸を決定します。たとえば、2 つの値を含む BY フィールドを選択した場合、2 つのグラフが生成されます。10 個の値を含むフィールドを選択した場合、10 個のグラフが生成されます。BY 句が 1 つで、ACROSS 句が 1 つの場合、BY フィールド内の値と同数のグラフが表示されます。ACROSS フィールドは、X 軸を決定します。2 つ目の横方向カテゴリは、同一リクエスト内に複数の BY 句または ACROSS 句と BY 句を 1 つずつ含めることにより選択することができます。
- GRMERGE ON では、統合された 1 つのグラフが作成されます。
- GRMERGE ADVANCED では、3 つのパラメータに基づいて次のことが決定されます。
  - 生成するグラフ数 (GRMULTIGRAPH)。
  - グラフ凡例に配置するソートフィールド数 (GRLEGEND)。

- X 軸に配置するソートフィールド数 (GRXAXIS)。

複数のグラフを、統合された形式または列形式で表示することができます。詳細は、1052 ページの 「 グラフの統合 」 および 1055 ページの 「 複数グラフの列表示 」 を参照してください。

構文　グラフの統合

```
SET GRMERGE={ON|OFF|ADVANCED}
```

説明

ON
: グラフの統合オプションをオンにします。

OFF
: グラフの統合オプションをオフにします。このオプションがデフォルト設定で選択されています。

ADVANCED
: 高度な統合オプションを有効にします。このオプションを選択すると、次の 3 つのパラメータに基づいて複数グラフの統合方法が決定されます。

- GRMULTIGRAPH - 複数グラフの作成に使用するソートフィールド数を指定します。
- GRLEGEND - グラフ凡例に配置するソートフィールド数を指定します。
- GRXAXIS - X 軸に表示するソートフィールド数を指定します。グラフを描画するには、GRXAXIS の値を少なくとも 1 に設定する必要があります。1 より大きい値を指定すると、ネストされた X 軸が作成されます。

注意：GRMULTIGRAPH、GRLEGEND、GRXAXIS で使用するソートフィールドの総数は、グラフリクエストで使用するソートフィールド数に一致させる必要があります。

GRMULTIGRAPH、GRLEGEND、GRXAXIS パラメータの構文は次のとおりです。

```
ON GRAPH SET GRMULTIGRAPH n
```

出力を複数のグラフに分割する際に使用するソートフィールド数 (0 から 2) を指定します。グラフの分割には、最も外側のソートフィールドが使用されます。n が 0 (ゼロ) より大きい場合、GRMERGE=OFF と同様に機能しますが、この設定ではソートフィールドを追加することが可能です。

```
ON GRAPH SET GRLEGEND n
```

GRMULTIGRAPH に使用する最も外側のソートフィールドを除いた残りのソートフィールドのうち、グラフ凡例に追加するソートフィールド数 (0 から 2) を指定します。n が 0 (ゼロ) より大きい場合、GRMERGE=ON と同様に機能しますが、この設定ではソートフィールドを追加することが可能です。

```
ON GRAPH SET GRXAXIS n
```

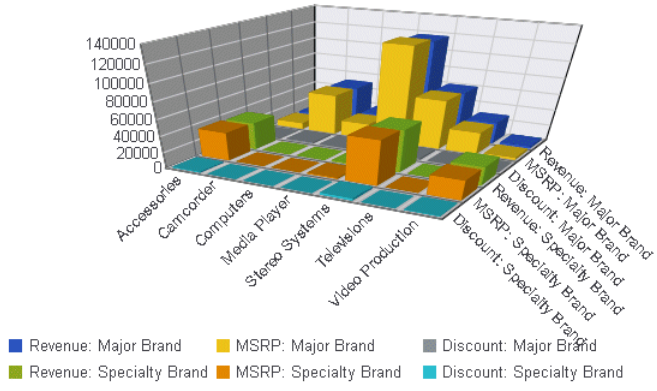
X 軸に表示する残りのソートフィールド数 (1 から 3) を指定します。n が 1 より大きい場合、ネストされた X 軸が作成されます。

例　GRMERGE ON による複数グラフの統合

次のリクエストは、2 つの横方向 (X 軸) カテゴリ (PRODUCT_CATEGORY および BRANDTYPE) が統合されたグラフを生成します。

```
SET GRMERGE=ON
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US MSRP_US DISCOUNT_US
ACROSS PRODUCT_CATEGORY
BY BRANDTYPE
ON GRAPH SET LOOKGRAPH 3D_BAR
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/ENIADefault_combine.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。

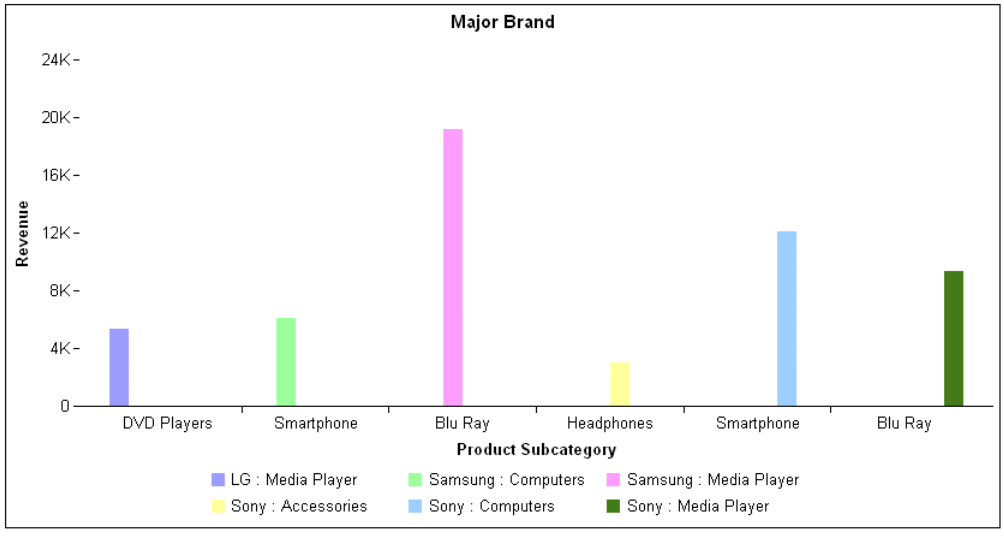


例　GRMERGE ADVANCED による複数グラフの統合

次のリクエストは、縦棒グラフを生成し、最も外側のソートフィールド (BRANDTYPE) の値ごとに、それぞれ個別のグラフを作成します。また、グラフ凡例に 2 つのソートフィールド (BRAND および PRODUCT_CATEGORY) を統合することでこれらを識別し、X 軸上に PRODUCT_SUBCATEG ソートフィールドを配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM  REVENUE_US
BY BRANDTYPE BY BRAND BY PRODUCT_CATEGORY BY PRODUCT_SUBCATEG
WHERE BRAND EQ 'Sony' OR 'Samsung' OR 'LG'
WHERE PRODUCT_SUBCATEG EQ 'Blu Ray' OR 'Smartphone' OR 'DVD Players' OR
'Headphones'
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 2
ON GRAPH SET GRXAXIS 1
ON GRAPH SET LOOKGRAPH VBAR
ON GRAPH PCHOLD FORMAT JSCHART
END
```

下図は、Major Brand グループのグラフを示しています。凡例では、Brand-Category の組み合わせが色分けされ、X 軸上では Brand-Category の組み合わせごとに Subcategory ソートフィールド値が繰り返されています。



構文　複数グラフの列表示

```
SET GRWIDTH=nn
```

説明
nn

複数のグラフを表示する列数です。0 (ゼロ) から 512 までの任意の値を指定します。デフォルト値は 0 です。

1 から 512 の値を指定すると、グラフは HTML テーブル内に表示されます。この HTML テーブルの列数は、指定した値に対応します。デフォルト値の 0 (ゼロ) では、グラフは Java アプレット内で上下に表示されます。

SET パラメータによるグラフのカスタマイズ

GRAPH 環境には、グラフの外観を制御し、リクエストの実行時にも制御可能なパラメータのセットが含まれます。

たとえば、BSTACK パラメータを使用すると、棒グラフの棒が横並びで表示されるのではなく、積み重ねられて表示されます。

構文　GRAPH リクエストでの SET パラメータの使用

GRAPH 環境を制御するパラメータを設定するには、SET パラメータの適当な組み合わせを使用します。

```
SET parameter=value,parameter=value...
```

サポートする GRAPH パラメータのリストは、1056 ページの 「 グラフの SET パラメータ値および関数 」 を参照してください。

注意

- それぞれの新しい行で SET コマンドを繰り返します。
- 複数のパラメータを 1 行に入力するときは、区切り文字にカンマ (,) を使用します。
- パラメータの一意の短縮名を使用することができます。それらの一意性を確認する必要があります。

例　GRAPH リクエストでの SET パラメータの使用

次のリクエストは、グラフの高さ (Y 軸) と幅 (X 軸) の設定方法を示しています。

```
SET HAXIS=75,VAXIS=40
GRAPH FILE filename
.
.
.
END
```

参照　グラフの SET パラメータ値および関数

グラフの SET パラメータ	値	パラメータ関数
3D	ON|OFF	ON に設定すると、3D グラフが生成されます。OFF に設定すると、2D グラフが生成されます。デフォルト値は ON です。

グラフの SET パラメータ	値	パラメータ関数
AUTOTICK	ON|OFF	ON に設定すると、目盛り間隔が自動的に設定されます。デフォルト値は ON です。HTICK および VTICK も参照してください。
BARNUMB	ON|OFF	ON に設定すると、棒グラフでは棒の末端に、円グラフでは円項目に集計値を表示します。デフォルト値は OFF です。
BSTACK	ON|OFF	ON に設定すると、棒グラフの棒が並べて表示されるのではなく、積み重ねられます。デフォルト値は OFF です。
EMBEDHEADING	ON|OFF	ON に設定すると、見出しがグラフに埋め込まれます。OFF に設定すると、見出しが HTML ページのグラフの上側に配置されます。デフォルト値は OFF です。
GRAPHDEFAULT	ON|OFF	ON に設定すると、WebFOCUS のデフォルトスタイルが使用されます。OFF に設定すると、デフォルトスタイルがレガシーグラフ API から取得されます。デフォルト値は ON です。
GRAPHEDIT	graphedit	WebFOCUS バージョン 8.0 以降では、このパラメータは廃止されました。
GRID	ON|OFF	ON に設定すると、グラフの横および縦クラスマークで罫線が描画されます。VGRID も参照してください。デフォルト値は OFF です。

グラフの SET パラメータ	値	パラメータ関数
HAUTO	ON|OFF	ON に設定すると、ユーザが上書きしない限り、横軸の目盛りが自動的に指定されます。OFF に設定すると、ユーザが HMAX および HMIN の値を指定する必要があります。デフォルト値は ON です。
HAXIS		横軸の幅をバイト数で指定します。このパラメータは、オフラインで生成したグラフに対して調整することができます。HAXIS は、グラフの幅が表示領域の幅に応じて自動的に調整されることから、オンライン表示では無視されます。
HCLASS	nnn	AUTOTICK=OFF に設定されている場合、横間隔マークを指定します。HTICK も参照してください。
HMAX	nnn	自動目盛りが使用されない (HAUTO=OFF) 場合、横軸上の最大値を指定します。
HMIN	nnn	自動目盛りが使用されない (HAUTO=OFF) 場合、横軸上の最小値を指定します。
HSTACK	ON|OFF	ON に設定すると、ヒストグラムの棒が並べて表示されるのではなく、積み重ねられます。デフォルト値は OFF です。

グラフの SET パラメータ	値	パラメータ関数
HTICK	nnn	AUTOTICK が OFF に設定されている場合、横軸の目盛り間隔を指定します。HCLASS も参照してください。
LOOKGRAPH	option	グラフのタイプを指定します。詳細は、40 ページの「LOOKGRAPH によるグラフタイプの制御」を参照してください。
VAUTO	ON|OFF	ON に設定すると、ユーザが上書きしない限り、縦軸の目盛りが自動的に指定されます。OFF に設定にすると、ユーザが VMAX および VMIN の値を指定する必要があります。デフォルト値は ON です。
VAXIS		ページの長さを行数で指定します。このパラメータは、オフラインで生成したグラフに対して調整することができます。VAXIS は、グラフの高さが表示領域の高さに応じて自動的に調整されることから、オンライン表示では無視されます。
VCLASS	nnn	AUTOTICK=OFF に設定されている場合、縦間隔マークを指定します。VTICK も参照してください。

グラフの SET パラメータ	値	パラメータ関数
VGRID	ON|OFF	ON に設定すると、グラフの横および縦クラスマークで罫線が描画されます。GRID も参照してください。デフォルト値は OFF です。
VMAX	nnn	自動目盛りが使用されない (VAUTO=OFF) 場合、縦軸上の最大値を指定します。
VMIN	nnn	自動目盛りが使用されない (VAUTO=OFF) 場合、縦軸上の最小値を指定します。
VTICK	nnn	AUTOTICK が OFF に設定されている場合、縦軸の目盛り間隔を指定します。VCLASS も参照してください。
VZERO	ON|OFF	Y 軸のミッシング値を格納するか、無視するかを指定します。ON に設定すると、Y 軸のミッシングデータが 0 (ゼロ) として処理されます。OFF に設定すると、Y 軸のミッシングデータは無視され、値はプロットマトリックスに格納されません。デフォルト値は OFF です。

B

リクエストのグラフ属性構文への変換

この章では、従来の WebFOCUS グラフリクエストをグラフ属性構文に変換する方法について説明します。

- 変換規則の概要
- 棒グラフ、折れ線グラフ、面グラフのリクエストのグラフ属性構文への変換
- ボックスプロットリクエストのグラフ属性構文への変換
- バブルグラフリクエストのグラフ属性構文への変換
- じょうごグラフリクエストのグラフ属性構文への変換
- マップグラフリクエストのグラフ属性構文への変換
- マリメッコチャートリクエストのグラフ属性構文への変換
- 円グラフリクエストのグラフ属性構文への変換
- 散布図リクエストのグラフ属性構文への変換
- ブロック地図リクエストのヒートマップリクエストへの変換
- ストリームグラフリクエストのグラフ属性構文への変換
- タグクラウドリクエストのグラフ属性構文への変換
- ツリーマップリクエストのグラフ属性構文への変換
- ネストされた X 軸を含むリクエストのグラフ属性構文への変換

変換規則の概要

従来の WebFOCUS グラフ構文では、BY 句および ACROSS 句を使用することができます。グラフ属性構文で使用できるのは BY 句のみとなり、リクエスト内の ACROSS 句は BY 句に変換する必要があります。変換がグラフの出力結果に影響することはありません。

また、従来のグラフリクエストが HTML5 グラフを生成しない場合は、グラフの出力フォーマットを JSCHART に変更する必要があります。たとえば、リクエストにより PNG フォーマットでグラフ出力が生成される場合、次のように記述されます。

```
ON GRAPH PCHOLD FORMAT PNG
```

次の構文を使用して、HTML5 グラフを生成するようフォーマットを変更する必要があります。

```
ON GRAPH PCHOLD FORMAT JSCHART
```

デフォルト設定では、リクエストに複数のソートフィールドが含まれる場合は、複数のグラフが生成されます。従来の WebFOCUS グラフ構文では、複数の WebFOCUS パラメータを使用してグラフコンポーネントへのソートフィールドの割り当て方法を指定することで、複数のグラフを統合することができます。GRMERGE パラメータは、単純な統合や高度な統合を有効にします。GRMERGE ON では、統合された 1 つのグラフが生成されます。GRMERGE ADVANCED では、次のパラメータに基づいて生成されるグラフの数が決定されます。

- GRMULTIGRAPH - 複数グラフの作成に使用するソートフィールド数を指定します。
- GRLEGEND - グラフ凡例に配置するソートフィールド数を指定します。
- GRXAXIS - X 軸に表示するソートフィールド数を指定します。1 より大きい値を指定すると、ネストされた X 軸が作成されます。ただし、緯度と経度のソートフィールドに基づくバブルマップは例外です。X 軸ラベルの連結シンボルを指定することで、ネストされた X 軸ラベルを生成するか、連結された X 軸ラベルを生成するかを制御することができます。詳細は、315 ページの 「 軸のプロパティ 」 を参照してください。

リクエスト内のソートフィールド数は、統合フィールドの合計数に一致します。統合パラメータがサポートされないグラフタイプもあります。グラフタイプによっては、特定の統合パラメータがサポートされません。また、GRLEGEND と GRXAXIS が交換可能で、ソートフィールド数と統合フィールド数を一致させるためにのみ GRLEGEND と GRXAXIS を必要とするグラフタイプもあります。これらのインスタンスについての詳細は、次のセクションで説明します。

グラフ属性構文では、統合パラメータは必要ありません。その代わりに、生成されるグラフ数と統合方法を制御する属性カテゴリに割り当てるソートフィールドをリクエスト内に追加することができます。これらの割り当てについての詳細は、次のセクションで説明します。グラフ属性構文についての詳細は、123 ページの 「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

注意：この章で説明するグラフリクエストには、WebFOCUS ツールより多くの統合パラメータを生成するものがあります。たとえば、WebFOCUS ツールでは GRLEGEND フィールドを 1 つしか許可しない場合でも、リクエストで GRLEGEND を 2 に設定することができます。この違いは、グラフ属性カテゴリへのパラメータの変換を表すものであり、WebFOCUS ツールが作成する実際のグラフリクエストを表すものではありません。

棒グラフ、折れ線グラフ、面グラフのリクエストのグラフ属性構文への変換

従来の棒グラフ、折れ線グラフ、面グラフは、縦方向、横方向、および並列、積み上げ、絶対、パーセントなどのレイアウトに対して、それぞれ異なる LOOKGRAPH パラメータ値を使用します。

グラフ属性構文では、同じグラフタイプのすべてのレイアウトと方向で 1 つの LOOKGRAPH 値が使用されます。レイアウトと方向は、WebFOCUS スタイルシートの TYPE=REPORT 宣言で指定されます。

棒グラフ、折れ線グラフ、面グラフでは、LOOKGRAPH パラメータを正しい値に変換し、グラフの方向およびレイアウトが WebFOCUS スタイルシートで正しく割り当てられていること、またメジャーとソートフィールドが適切な属性カテゴリに割り当てられていることを確認します。

グラフ属性構文は、類似した X 軸の値をクラスタ化するため、出力結果の外観は GRMERGE 構文で生成された出力結果と異なります。GRMERGE 構文による出力では、X 軸のソートフィールド値の集合はクラスタ化されません。

参照

棒グラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を、新しい CHART-ORIENTATION および CHART-SERIES-LAYOUT の値とともに示しています。

LOOKGRAPH のパラメータ値		TYPE=REPORT のスタイルシート属性	
HBAR	BAR	horizontal	side-by-side (デフォルト)
VBAR, BAR	BAR	vertical (デフォルト)	side-by-side (デフォルト)
HBRSTK1	BAR	horizontal	stacked
VBRSTK1, STACK	BAR	vertical (デフォルト)	stacked
HBRSTKPC	BAR	horizontal	percent
VBRSTKPC	BAR	vertical (デフォルト)	percent

参照

折れ線グラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を、新しい CHART-ORIENTATION および CHART-SERIES-LAYOUT の値とともに示しています。

LOOKGRAPH のパラメータ値		TYPE=REPORT のスタイルシート属性	
HLINE	LINE	horizontal	absolute (デフォルト)

LOOKGRAPH のパラメータ値		TYPE=REPORT のスタイルシート属性	
VLINE, LINE	LINE	vertical (デフォルト)	absolute (デフォルト)
HLINSTK	LINE	horizontal	stacked
VLINSTK	LINE	vertical (デフォルト)	stacked
HLNSTKPC	LINE	horizontal	percent
VLNSTKPC	LINE	vertical (デフォルト)	percent

参照

面グラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を、新しい CHART-ORIENTATION および CHART-SERIES-LAYOUT の値とともに示しています。

LOOKGRAPH のパラメータ値		TYPE=REPORT のスタイルシート属性	
HAREA	AREA	horizontal	absolute (デフォルト)
VAREA	AREA	vertical (デフォルト)	absolute (デフォルト)
HAREASTK	AREA	horizontal	stacked
VAREATK	AREA	vertical (デフォルト)	stacked
HARESTKP	AREA	horizontal	percent
VARESTKP	AREA	vertical (デフォルト)	percent

参照

棒グラフ、折れ線グラフ、面グラフの属性カテゴリの割り当て

下表は、棒グラフ、折れ線グラフ、面グラフの属性カテゴリの変換を示しています。

フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	y-axis

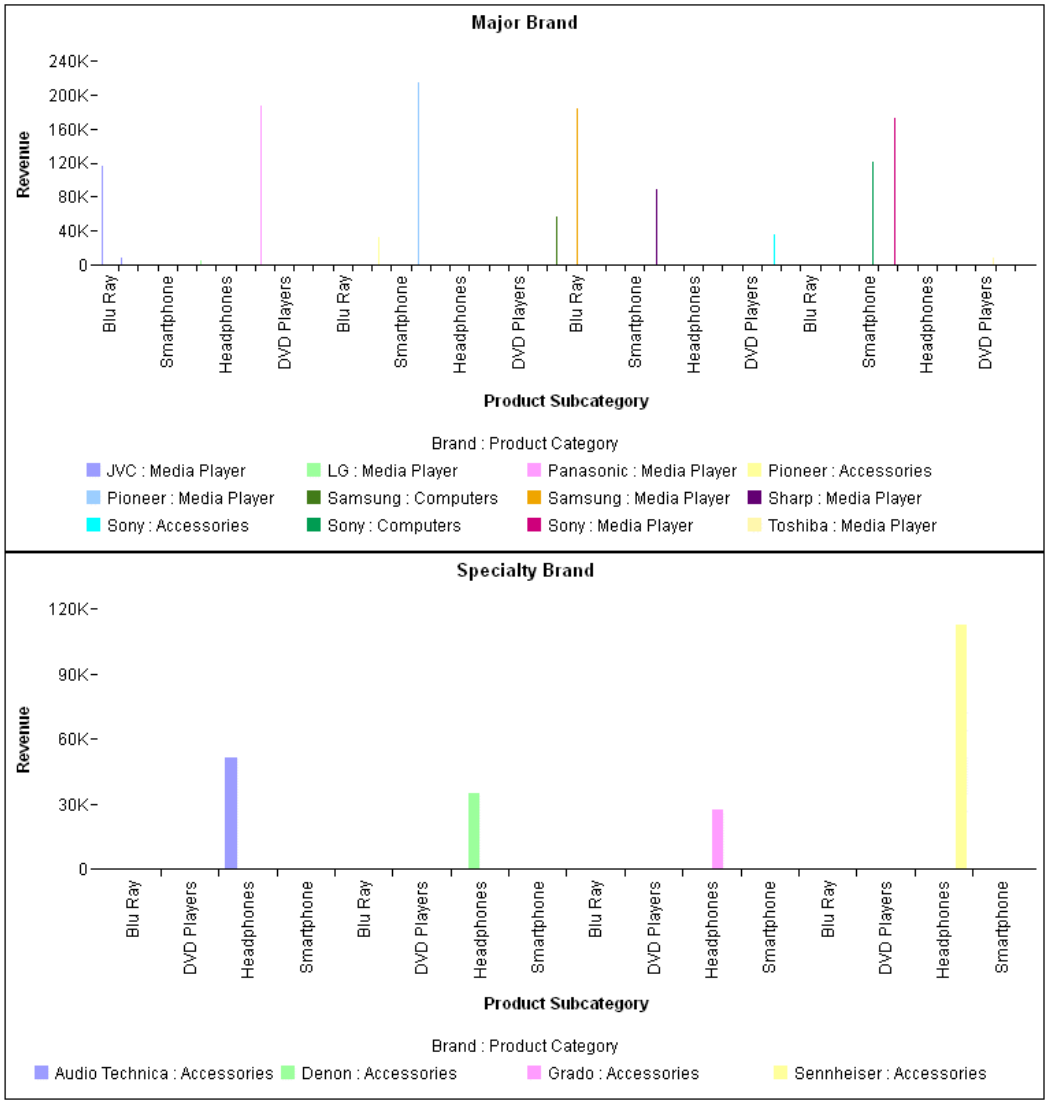
フィールドまたはパラメータのタイプ	属性カテゴリ
GRXAXIS ソートフィールド	x-axis
GRLEGEND ソートフィールド	color
GRMULTIGRAPH ソートフィールド	page

例　棒グラフリクエストのグラフ属性構文への変換

次のリクエストは、縦棒グラフを生成し、最も外側のソートフィールド (BRANDTYPE) の値ごとに、それぞれ個別のグラフを作成します。また、グラフ凡例に 2 つのソートフィールド (BRAND および PRODUCT_CATEGORY) を統合することでこれらを識別し、X 軸上に PRODUCT_SUBCATEG ソートフィールドを配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY BRANDTYPE
BY BRAND
BY PRODUCT_CATEGORY
ACROSS PRODUCT_SUBCATEG
WHERE PRODUCT_SUBCATEG EQ 'Blu Ray' OR 'Smartphone' OR 'DVD Players' OR
'Headphones'
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 2
ON GRAPH SET GRXAXIS 1
ON GRAPH SET LOOKGRAPH VBAR
ON GRAPH PCHOLD FORMAT JSCHART
END
```

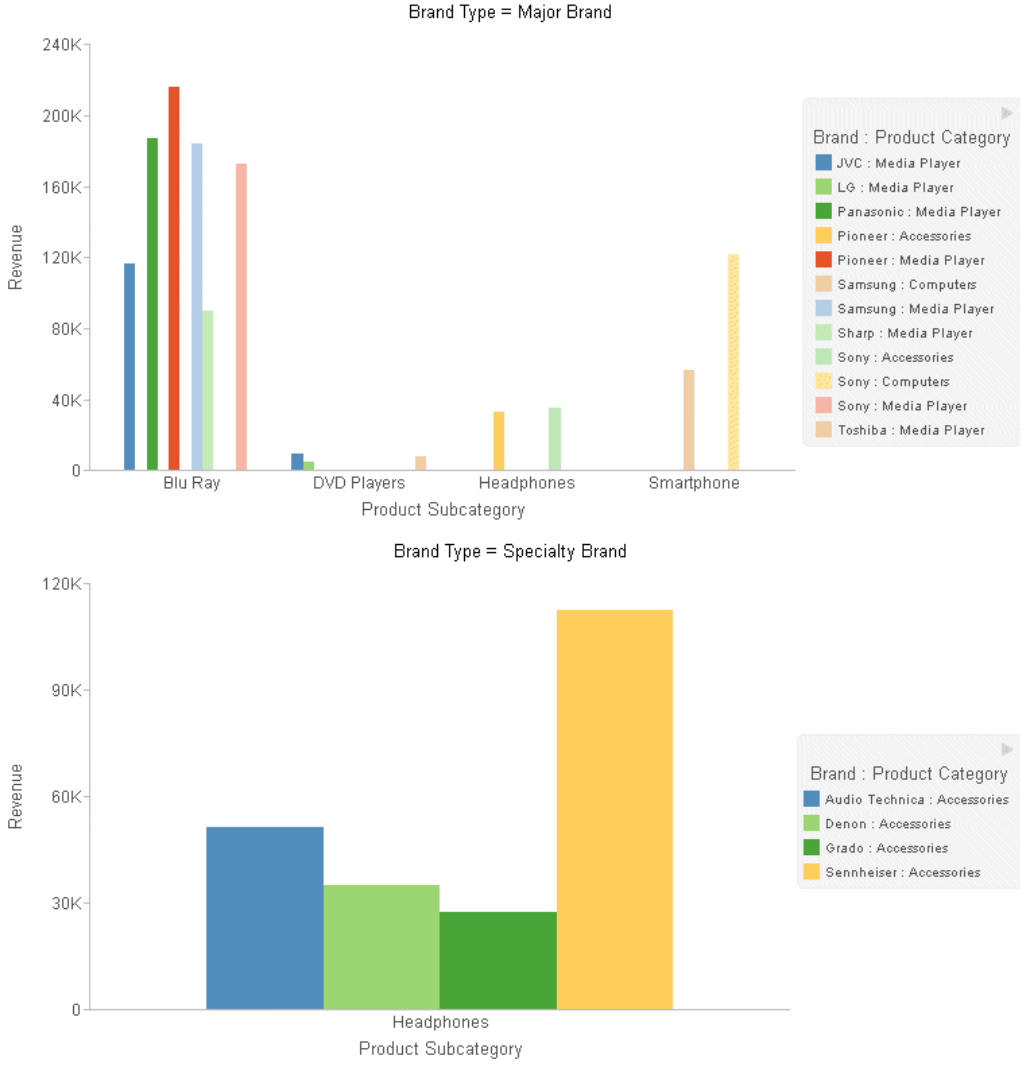
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。BRANDTYPE ソートフィールドが page、BRAND および PRODUCT_CATEGORY のソートフィールドが color、PRODUCT_SUBCATEG ソートフィールドが x-axis、REVENUE_US のメジャーが y-axis のカテゴリにそれぞれ割り当てられています。ACROSS 句は BY 句に置き換えられており、グラフタイプは BAR です。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"Brand Type = <BRANDTYPE"
SUM REVENUE_US
BY BRANDTYPE
BY BRAND
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
WHERE PRODUCT_SUBCATEG EQ 'Blu Ray' OR 'Smartphone' OR 'DVD Players' OR
'Headphones'
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=page,$
TYPE=DATA, COLUMN=BRAND, BUCKET=COLOR,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color,$
TYPE=DATA, COLUMN=PRODUCT_SUBCATEG, BUCKET=x-axis,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=y-axis,$
END
```

下図の出力結果は、GRMERGE の出力と外観が異なります。これは、グラフ属性構文が類似した X 軸の値をクラスタ化するためです。この例では、すべてのブランドのメディアプレイヤがクラスタ化されて表示されています。

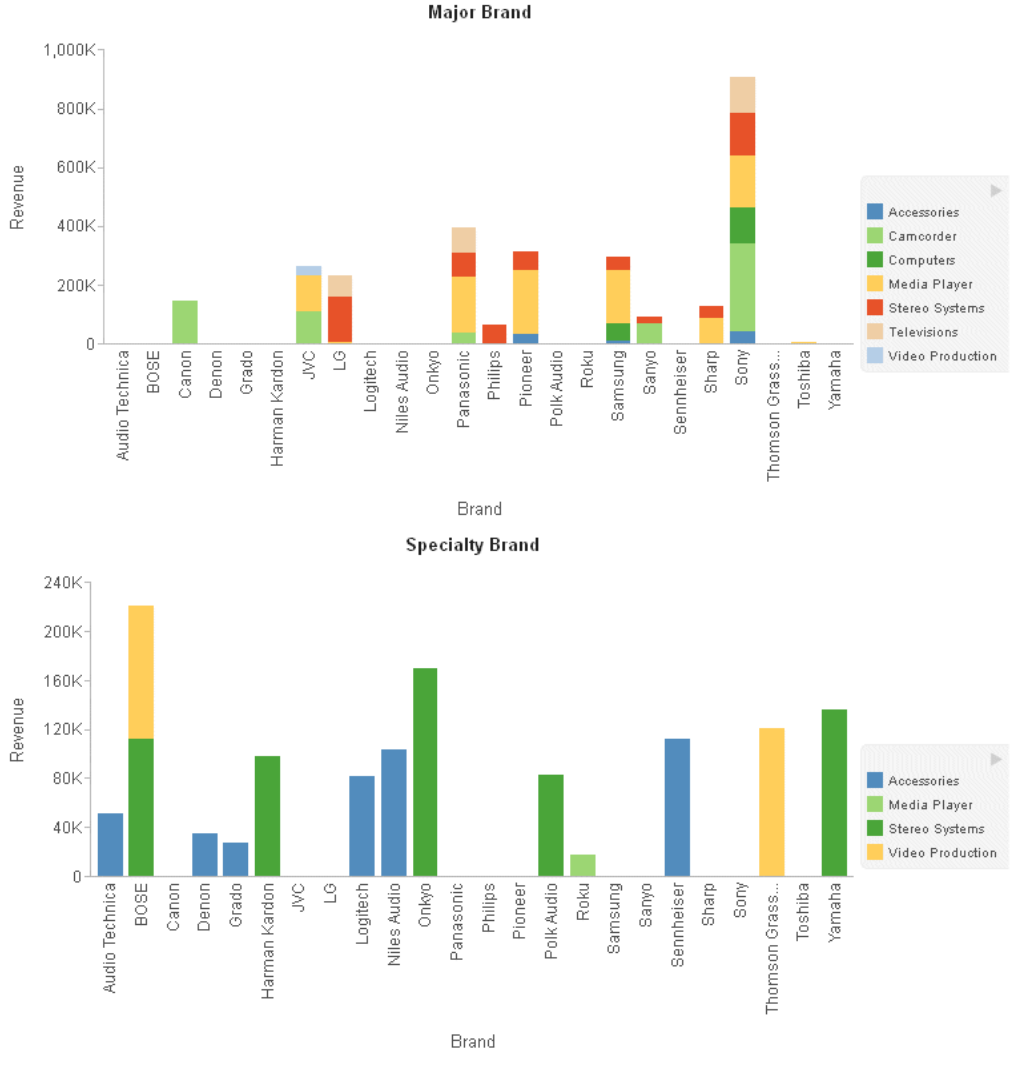


例　積み上げ棒グラフリクエストのグラフ属性構文への変換

次のリクエストは、複数の積み上げ棒グラフを生成します。BRANDTYPE (GRMULTIGRAPH フィールド) の値ごとに新しいグラフが生成されます。PRODUCT_CATEGORY フィールドが GRLEGEND フィールド、BRAND フィールドが GRXAXIS フィールドになります。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ACROSS BRAND
ON GRAPH SET AUTOFIT ON
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 1
ON GRAPH SET LOOKGRAPH VBRSTK1
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

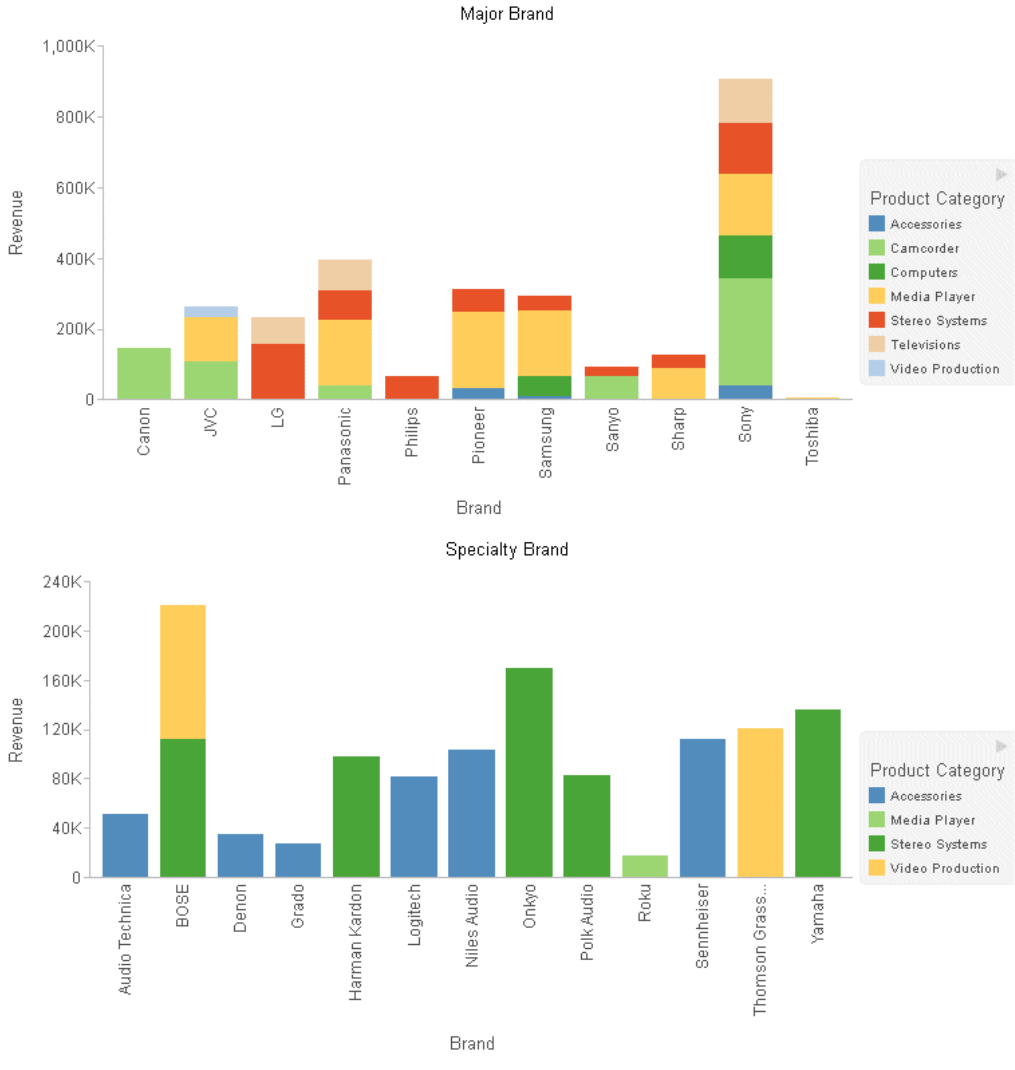
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。CHART-SERIES-LAYOUT は stacked です。メジャー REVENUE_US は y-axis、BRANDTYPE ソートフィールドは page、PRODUCT_CATEGORY ソートフィールドは color、BRAND ソートフィールドは x-axis のカテゴリにそれぞれ割り当てられています。元のリクエストの ACROSS ソート句は、BY 句に置き換えられています。グラフタイプは BAR (棒グラフ) です。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE"
SUM REVENUE_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
BY BRAND
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET AUTOFIT ON
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=report, chart-series-layout=stacked,$
type=data, column=brandtype, bucket=page,$
type=data, column=brand, bucket=x-axis,$
type=data, column=product_category, bucket=color,$
type=data, column=revenue_us, bucket=y-axis,$
END
```

出力結果は次のとおりです。



ボックスプロットリクエストのグラフ属性構文への変換

ボックスプロットは、データを四分位に分けて示します。ボックスプロットでは、このような分割を視覚化するために、各シリーズとグループに 5 つの値が必要です。

従来の構文では、正しい四分位に割り当てられるよう、メジャーは昇順で指定する必要があります。グラフ属性構文では、カテゴリの割り当てによりこの条件が不要になります。

参照

ボックスプロットの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
BOXPLOT	BOXPLOT

参照

ボックスプロットの属性カテゴリの割り当て

下表は、ボックスプロットの属性カテゴリの変換を示しています。

フィールドまたはパラメータのタイプ	属性カテゴリ
最小値メジャーフィールド	min
ボックス下端メジャーフィールド	lower
中央値メジャーフィールド	median
ボックス上端メジャーフィールド	upper
最大値メジャーフィールド	max
ソートフィールド	x-axis

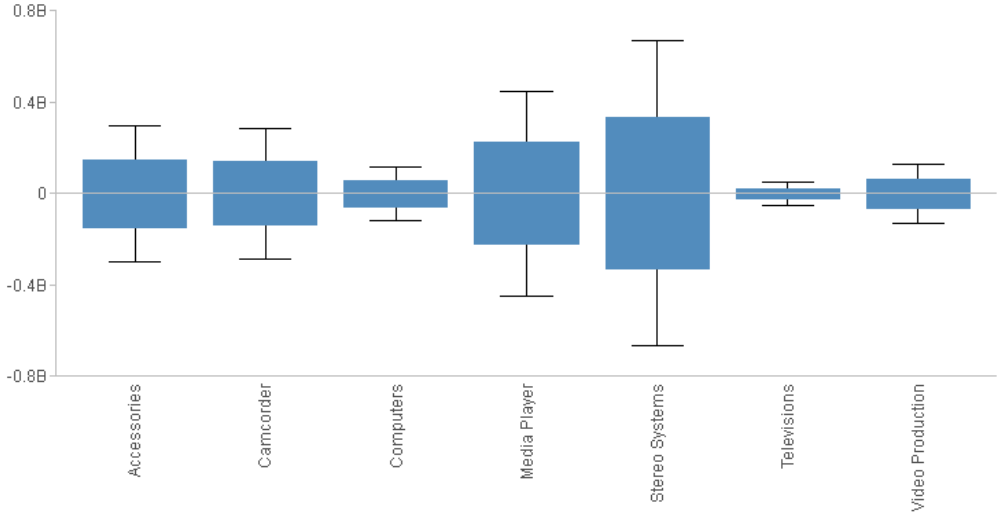
注意：ボックスプロットでは統合パラメータは使用できません。

例　ボックスプロットリクエストのグラフ属性構文への変換

次のリクエストは、ボックスプロットを生成します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"boxplotProperties": {"drawHatAsBox": false}
*END
ENDSTYLE
END
```

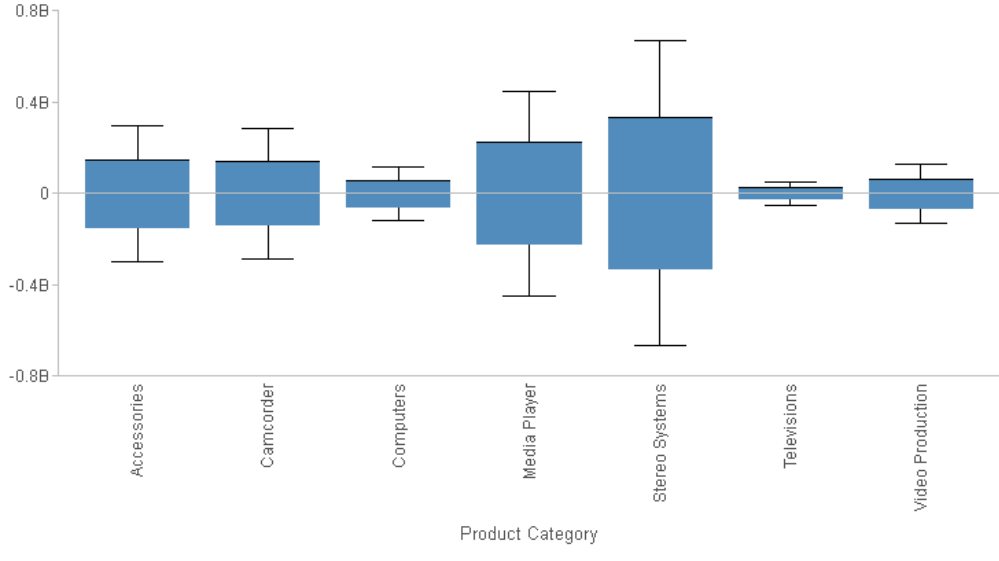
出力結果は次のとおりです。



次のリクエストでは、グラフ属性構文を使用します。

```
DEFINE FILE WF_RETAIL_LITE
DIFF1 = COGS_US -100000;
DIFF2 = COGS_US -200000;
DIFF3 = COGS_US +100000;
DIFF4 = COGS_US +200000;
END
GRAPH FILE WF_RETAIL_LITE
SUM DIFF1 DIFF2 MDN.COGS_US DIFF3 DIFF4
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BOXPLOT
ON GRAPH SET STYLE *
type=data, column=product_category, bucket=x-axis,$
type=data, column=diff2, bucket=min,$
type=data, column=diff1, bucket=lower,$
type=data, column=C4, bucket=median,$
type=data, column=diff3, bucket=upper,$
type=data, column=diff4, bucket=max,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"boxplotProperties": {"drawHatAsBox": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



バブルグラフリクエストのグラフ属性構文への変換

バブルグラフは散布図で、各マーカーのサイズはメジャー値に基づきます。

参照 バブルグラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
BUBBLE	BUBBLE

参照 バブルグラフの属性カテゴリの割り当て

下表は、バブルグラフの属性カテゴリの変換を示しています。

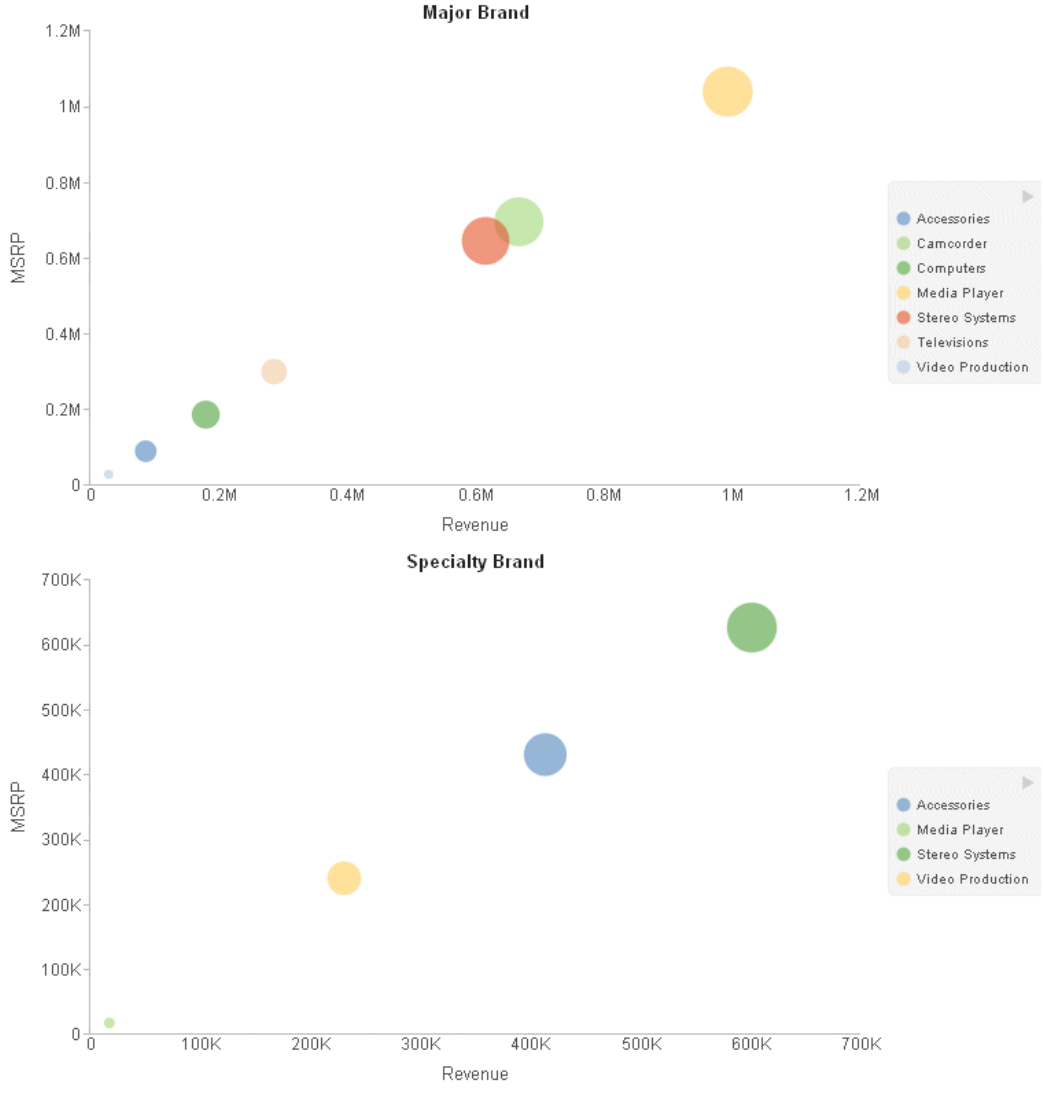
フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	y-axis
メジャーフィールド	x-axis
メジャーフィールド	size
GRLEGEND または GRXAXIS ソートフィールド	color
GRMULTIGRAPH ソートフィールド	page

例　バブルグラフリクエストのグラフ属性構文への変換

次のリクエストは、従来の構文を使用して複数のバブルグラフを生成します (LOOKGRAPH 値は BUBBLE) 。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のバブルグラフが生成されます。PRODUCT_CATEGORY ソートフィールドは、GRLEGEND ソートフィールドです。デフォルト設定では、凡例が表示されないため、"legend": には JSON "visible": true プロパティを設定します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
MSRP_US
GROSS_PROFIT_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 0
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"legend": {"visible": true}
*END
ENDSTYLE
END
```

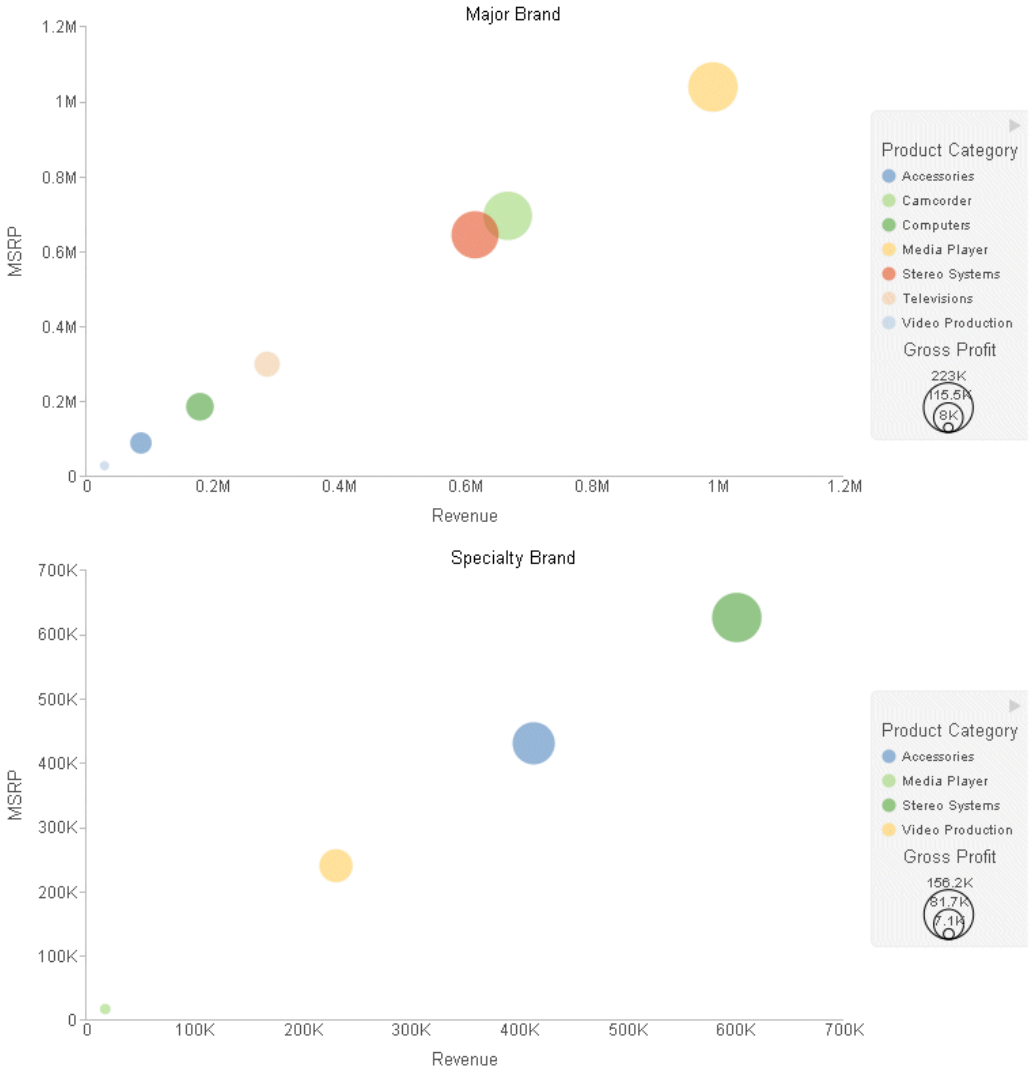
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、BUBBLE です。メジャー (REVENUE_US、MSRP_US、GROSS_PROFIT_US) は、x-axis、y-axis、size の属性カテゴリにそれぞれ割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (PRODUCT_CATEGORY) は color の属性カテゴリに割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
SUM REVENUE_US
MSRP_US
GROSS_PROFIT_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH SET LOOKGRAPH BUBBLE
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN = BRANDTYPE, BUCKET=PAGE,$
TYPE=DATA, COLUMN = PRODUCT_CATEGORY, BUCKET=COLOR,$
TYPE=DATA, COLUMN = REVENUE_US, BUCKET=x-axis,$
TYPE=DATA, COLUMN = MSRP_US, BUCKET=y-axis,$
TYPE=DATA, COLUMN = GROSS_PROFIT_US, BUCKET=SIZE,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
END
```

出力結果は次のとおりです。



じょうごグラフリクエストのグラフ属性構文への変換

じょうごグラフは、ソートフィールドに基づいてメジャーを項目化します。リクエストに従来の WebFOCUS 構文を使用するか、グラフ属性構文を使用するか、スタイルシートを使用するかによって、いくつかのプロパティではデフォルト設定が異なります。

参照

じょうごグラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
FUNNEL	FUNNEL

参照

じょうごグラフの属性カテゴリの割り当て

下表は、じょうごグラフの属性カテゴリの変換を示しています。

フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	measure
GRLEGEND または GRXAXIS ソートフィールド (いずれか 1 つのみサポート)	color
GRMULTIGRAPH ソートフィールド	page

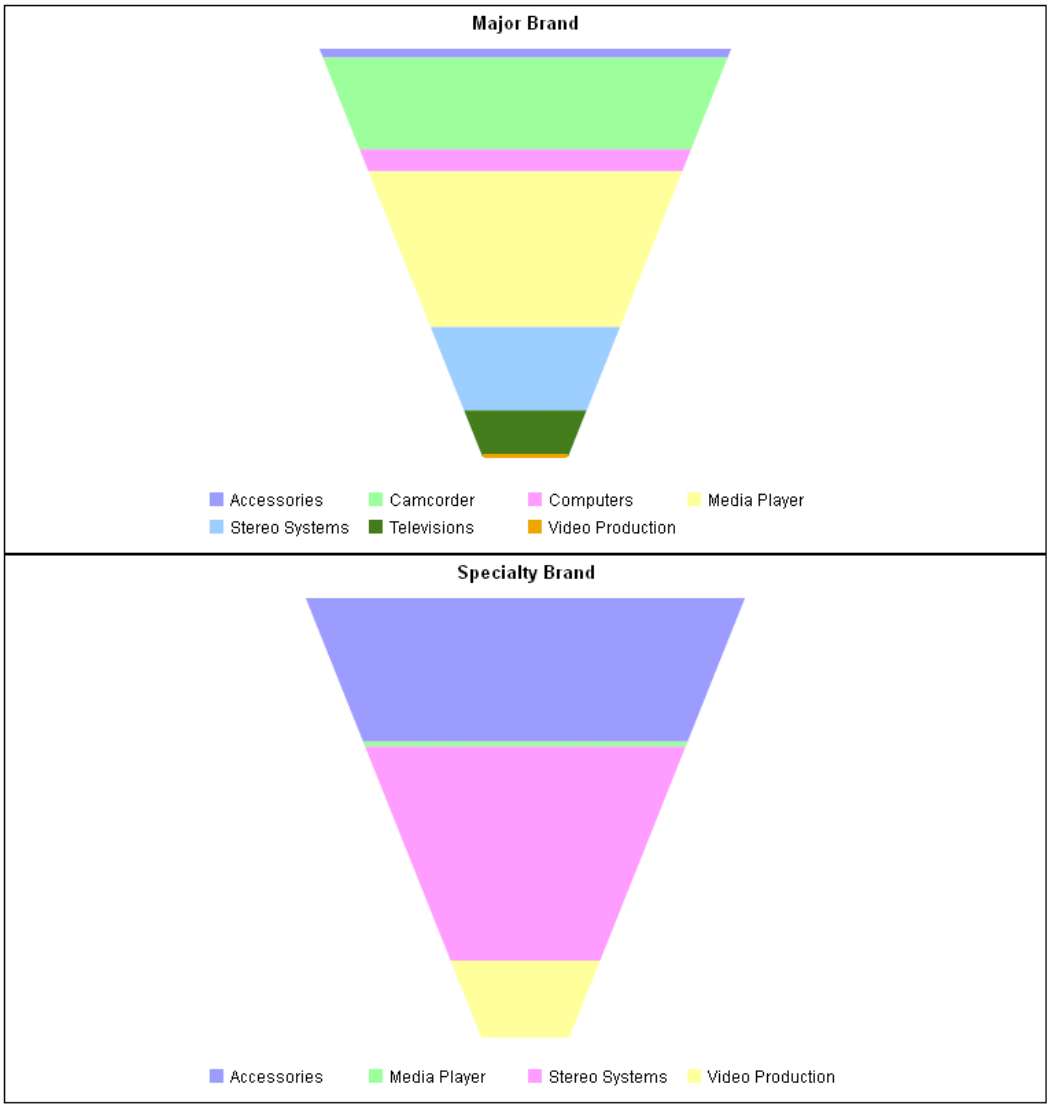
例

じょうごグラフリクエストのグラフ属性構文への変換

次のリクエストは、従来の構文を使用してじょうごグラフを生成します (LOOKGRAPH 値は FUNNEL) 。BRANDTYPE の値ごとに個別のじょうごグラフが生成されます。PRODUCT_CATEGORY ソートフィールドは、GRLEGEND ソートフィールドです。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 0
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET AUTOFIT ON
END
```

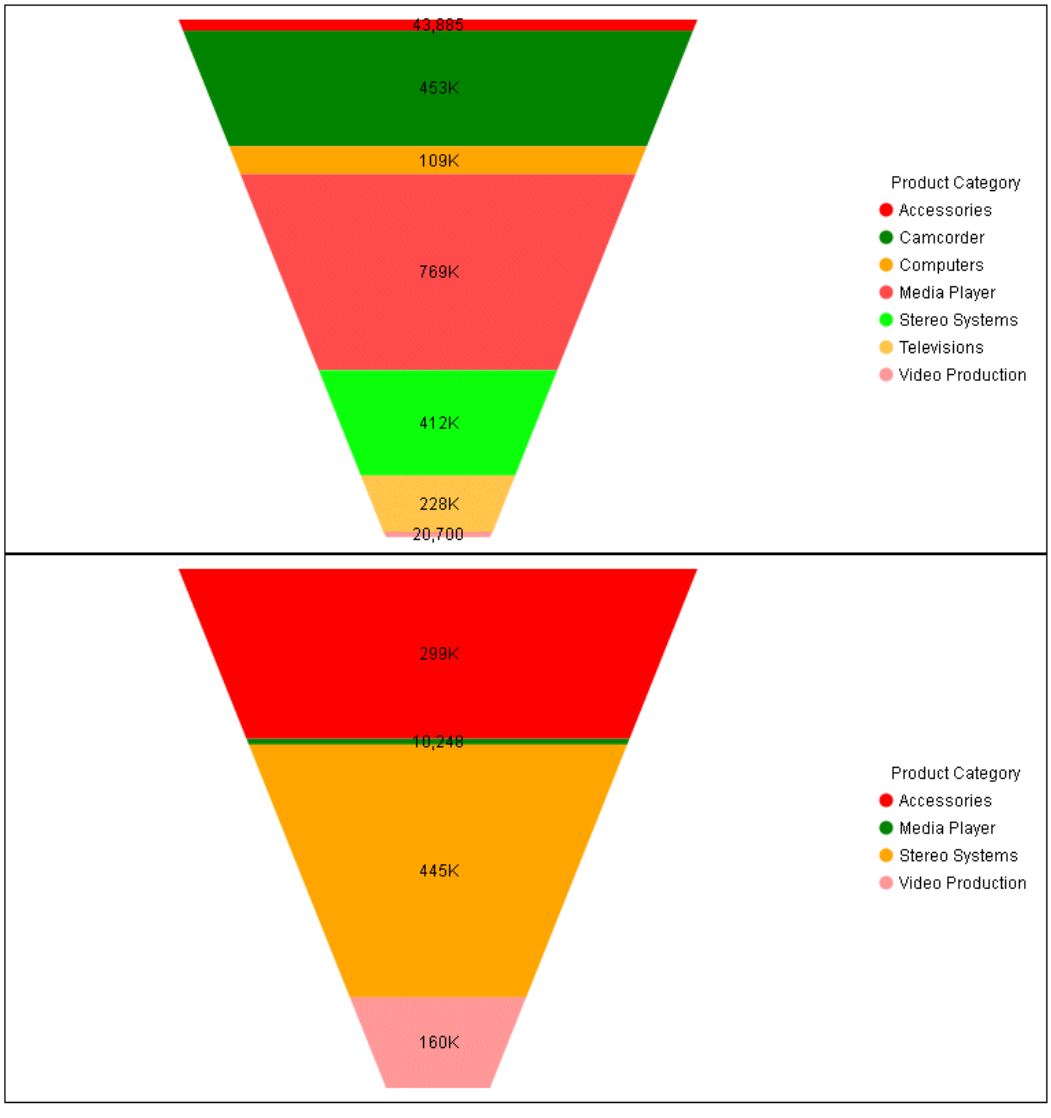
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、FUNNEL です。メジャー (COGS_US) は measure 属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (PRODUCT_CATEGORY) は color の属性カテゴリに割り当てられます。色やデータテキストラベルの作成などいくつかのデフォルトプロパティは、従来の構文のデフォルト設定とは異なります。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH FUNNEL
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=measure,$
type=data, column=brandtype, bucket=page,$
type=data, column=product_category, bucket=color,$
ENDSTYLE
END
```

出力結果は次のとおりです。



マップグラフリクエストのグラフ属性構文への変換

コロプレスマップでは、場所を表す多角形がメジャー値に基づいて色分けされます。バブルマップでは、地図上の場所や地点 (緯度と経度の値で定義される) に、プロポーショナルシンボルが表示されます。

従来の WebFOCUS 構文は、HTML5 グラフに対して Leaflet マップエンジンのみサポートします。WebFOCUS グラフ属性構文は、Leaflet と ESRI 両方のマップグラフをサポートします。このセクションでは、従来の WebFOCUS 構文をグラフ属性構文に変換する方法について説明します。マップエンジンの変換方法についてはここでは説明しません。マップグラフについての詳細は、871 ページの 「マップサポート」 を参照してください。

参照

マップグラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
CHOROPLETH	CHOROPLETH
BUBBLEMAP	BUBBLEMAP

参照

マップグラフの属性カテゴリの割り当て

下表は、マップグラフの属性カテゴリの変換を示しています。

フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド (コロプレスマップ)	color
メジャーフィールド (バブルマップ)	size
GRMULTIGRAPH ソートフィールド	page
GRLEGEND (マップではサポートされません)	なし
GRXAXIS ソートフィールド コロプレスマップでは、ソートフィールドが location のみであるため、GRXAXIS の値は常に 1 になります。バブルマップでは、ソートフィールドが location または latitude および longitude のいずれかであるため、GRXAXIS の値は 1 または 2 になります。	GRXAXIS の値が 1 である場合、location の属性カテゴリに変換します。 GRXAXIS の値が 2 である場合、latitude および longitude の属性カテゴリに変換します。

例　コロプレスマップリクエストのグラフ属性構文への変換

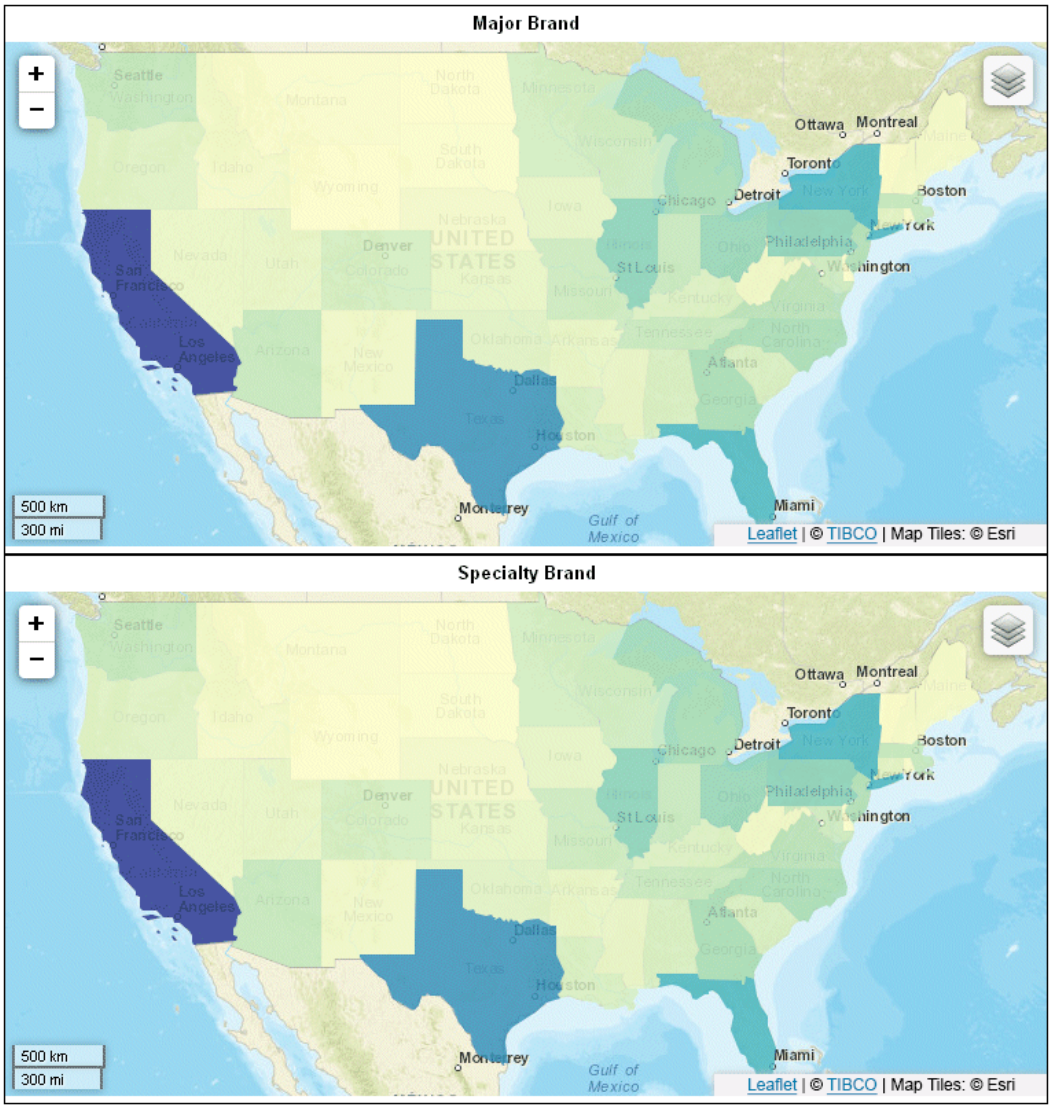
次のリクエストは、従来の構文を使用して複数のコロプレスマップを生成します (LOOKGRAPH 値は CHOROPLETH)。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のコロプレスマップが生成されます。メジャー (MIN.STATE_PROV_POPULATION) が多角形の色に使用され、ソートフィールド (STATE_PROV_NAME) はロケーションフィールド (GRXAXIS フィールド) です。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.STATE_PROV_POPULATION
BY BRANDTYPE
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 0
ON GRAPH SET GRXAXIS 1
ON GRAPH SET STYLE *
*GRAPH_JS
"mapProperties": {
"engine": "leaflet",
"leaflet": {
"initPos": {
"center": [37.8, -96],
"level": 4
},
```

```
"overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],
"controls": [
{"control": "L.Control.Layers"},
{
"control": "L.Control.Scale",
"options": {
"imperial": true,
"metric": true }
}
],
"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 17,
"minZoom": 0,
"attribution": function(){ return "&|copy; <a target='_blank'href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles:&|copy; Esri";}
},
"url": function(){return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

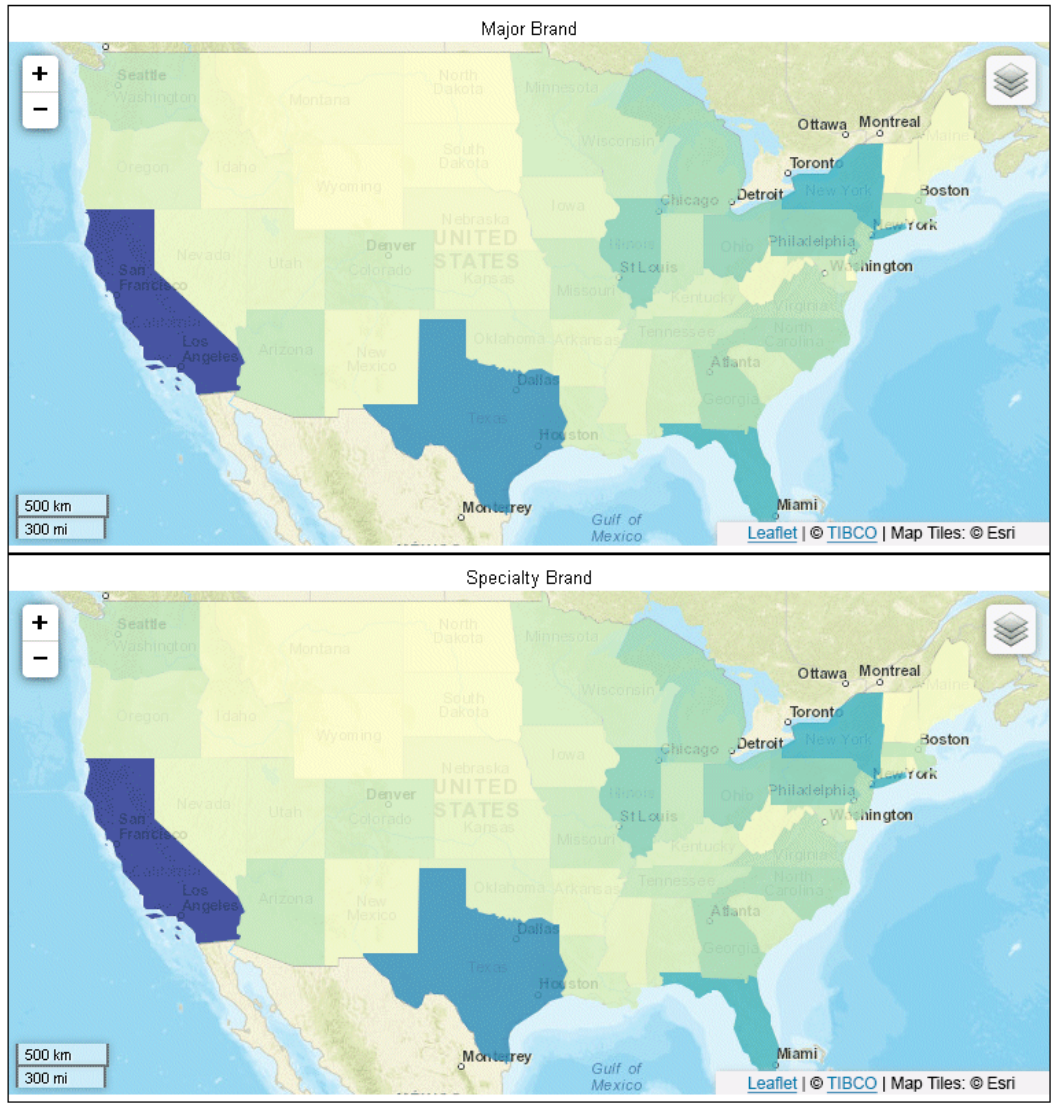
以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、CHOROPLETH です。メジャー (MIN.STATE_PROV_POPULATION) は color の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (STATE_PROV_NAME) は location の属性カテゴリにそれぞれ割り当てられます。出力結果をデフォルト設定の従来の構文による出力結果と同様にするため、"legend" に "visible": false のプロパティが設定されています。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
SUM MIN.STATE_PROV_POPULATION
BY BRANDTYPE
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=PAGE,$
TYPE=DATA, COLUMN=MIN.STATE_PROV_POPULATION, BUCKET=color,$
TYPE=DATA, COLUMN=STATE_PROV_NAME, bucket=location,$
*GRAPH_JS
"legend":{"visible":false},
"legend":{"visible":false},
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
           },
```

```
"overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){ return tdgchart.getScriptPath() + 'map/US.json'}
}],
  "controls": [
    {"control": "L.Control.Layers"},
    {
     "control": "L.Control.Scale",
     "options": {
      "imperial": true,
      "metric": true }
    }
   ],
 "baselayers": [{
 "title": "ArcGIS_World_Street_Map",
 "layerInfo": {
 "maxZoom": 17,
 "minZoom": 0,
 "attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
  "url": function(){ return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';}
}]
}
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

例　位置ベースのバブルマップリクエストのグラフ属性構文への変換

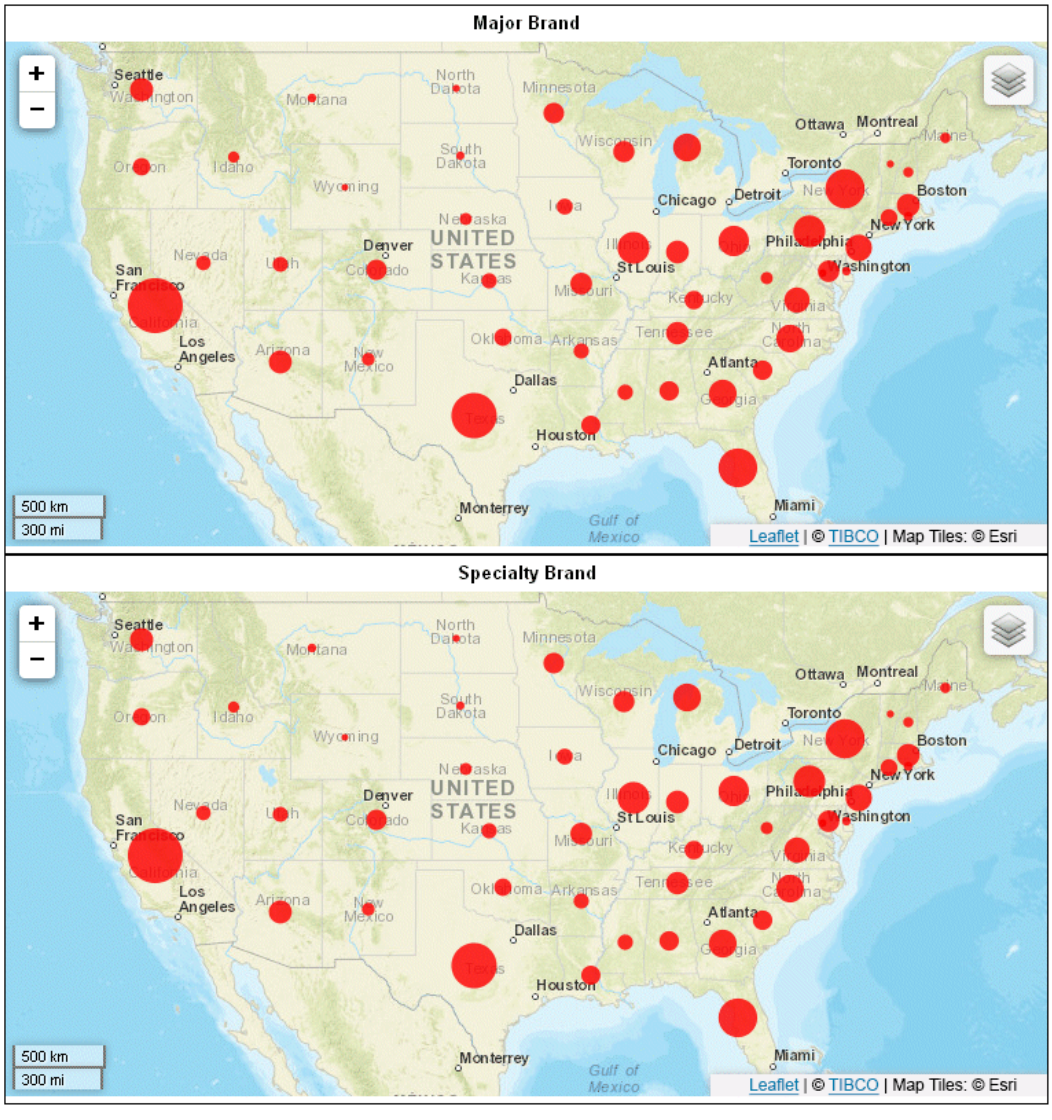
次のリクエストは、従来の構文を使用して複数のバブルマップを生成します (LOOKGRAPH 値は BUBBLEMAP)。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のバブルマップが生成されます。メジャー (MIN.STATE_PROV_POPULATION) が、プロポーショナルシンボルのサイズに使用され、ソートフィールド (STATE_PROV_NAME) は位置フィールド (GRXAXIS フィールド) です。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
SUM MIN.STATE_PROV_POPULATION
BY BRANDTYPE
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET AUTOFIT ON
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 0
ON GRAPH SET GRXAXIS 1
ON GRAPH SET STYLE *
*GRAPH_JS
"bubbleMarker":{"maxSize":"10%"},
"mapProperties": {
  "engine": "leaflet",

"leaflet": {
"initPos": {
                  "center": [37.8, -96],
                  "level": 4
           },
```

```
"overlayLayers": [{
"title": "United States of America",
"dataLookup": "properties.state_name",
"layerInfo": {
"maxZoom": -1,
"minZoom": -1,
"type": "regions"
},
"type": "tdg",
"url": function(){ return tdgchart.getScriptPath() + 'map/US.json'}
}],
  "controls": [
    {"control": "L.Control.Layers"},
    {
     "control": "L.Control.Scale",
     "options": {
      "imperial": true,
      "metric": true }
    }
   ],
 "baselayers": [{
 "title": "ArcGIS_World_Street_Map",
 "layerInfo": {
 "maxZoom": 17,
 "minZoom": 0,
 "attribution": function(){ return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri";}
},
  "url": function(){ return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';}
}]
}
},
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

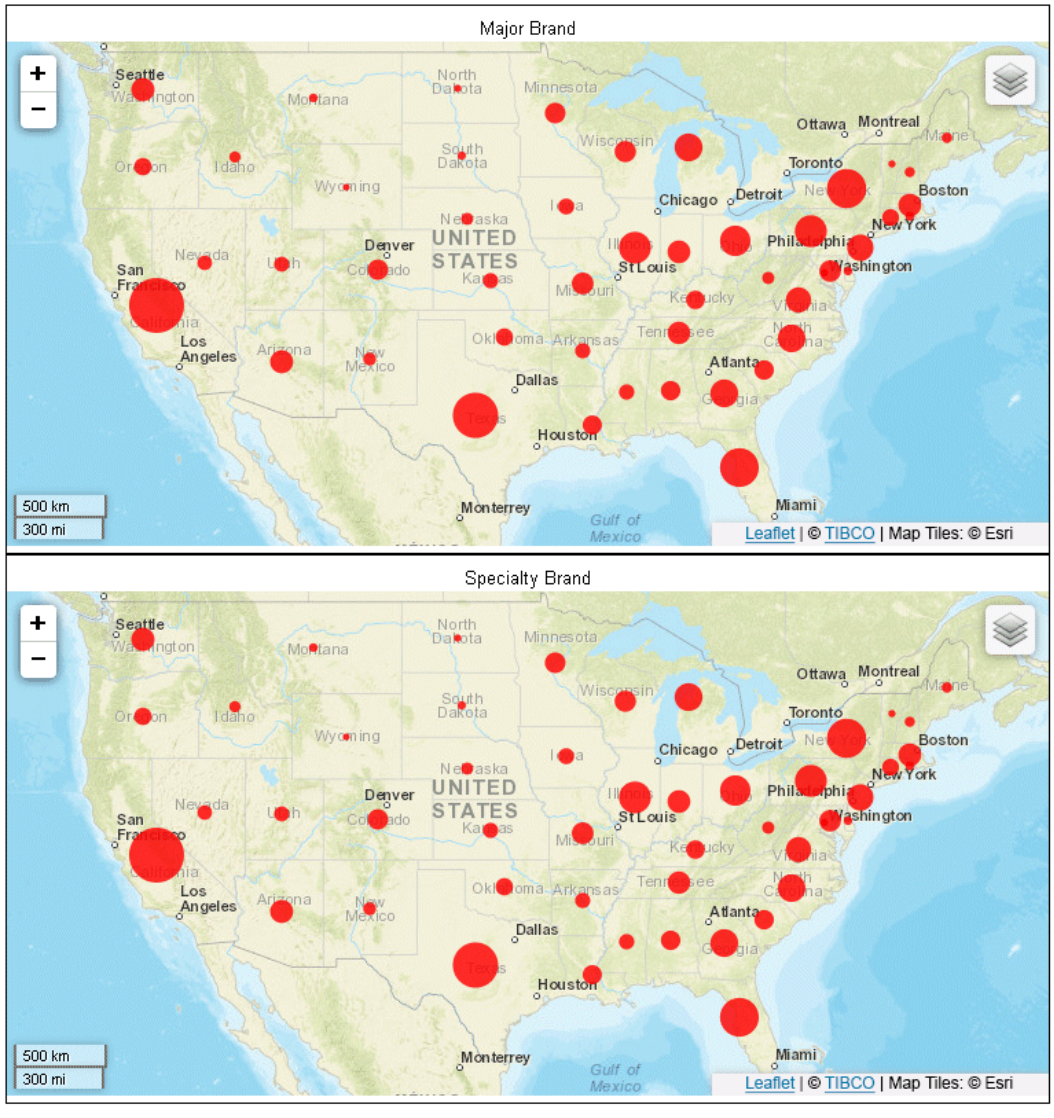
以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、BUBBLEMAP です。メジャー (MIN.STATE_PROV_POPULATION) は size の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (STATE_PROV_NAME) は location の属性カテゴリにそれぞれ割り当てられます。出力結果を従来の構文による出力結果と同様にするため、"legend" に "visible": false のプロパティが設定されています。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
SUM MIN.STATE_PROV_POPULATION
BY BRANDTYPE
BY STATE_PROV_NAME
WHERE COUNTRY_NAME EQ 'United States'
WHERE STATE_PROV_NAME NE 'Puerto Rico'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET AUTOFIT ON
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=PAGE,$
TYPE=DATA, COLUMN=STATE_PROV_NAME, BUCKET=LOCATION,$
TYPE=DATA,COLUMN= MIN.STATE_PROV_POPULATION, BUCKET=SIZE,$
*GRAPH_JS
"legend":{"visible":false},
"bubbleMarker":{"maxSize":"10%"},"mapProperties": {
"leaflet": {
"initPos": {
"center": [37.8, -96],
"level": 4
},
```

```
"overlayLayers": [{
"title": "US",
"layerInfo": {
"type": "regions"
},
"type": "tdg",
"url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
}],
"baselayers": [{
"title": "ArcGIS_World_Street_Map",
"layerInfo": {
"maxZoom": 16,
"attribution": function() { return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles:&|copy; Esri"; }
},
"url": function() { return 'https://services.arcgisonline.com/ArcGIS/rest/services/
World_Street_Map/MapServer/tile/{z}/{y}/{x}';
}
}]
}
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

例　緯度/経度ベースのバブルマップリクエストのグラフ属性構文への変換

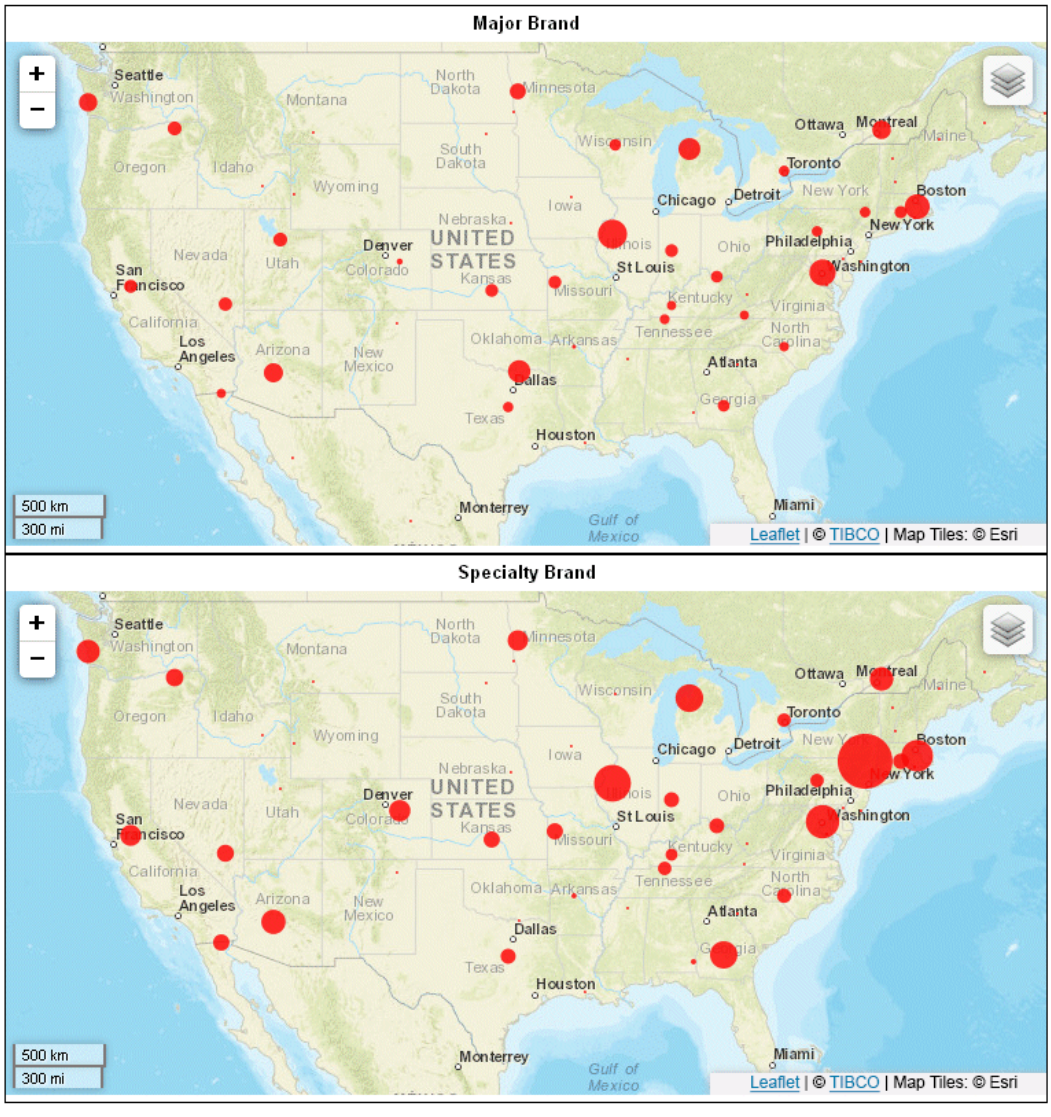
次のリクエストは、従来の構文を使用して複数のバブルマップを生成します (LOOKGRAPH 値は BUBBLEMAP) 。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のバブルマップが生成されます。メジャー (MIN.CITY_POPULATION) が、プロポーショナルシンボルのサイズに使用され、緯度と経度のソートフィールド (STATE_PROV_LATITUDE および STATE_PROV_LONGITUDE) は GRXAXIS の値 (2) に使用されます。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE wf_retail_lite
SUM MIN.CITY_POPULATION
BY BRANDTYPE
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 0
ON GRAPH SET GRXAXIS 2
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
*GRAPH_JS
"bubbleMarker":{"maxSize":"10%"},
"mapProperties": {
  "leaflet": {
   "initPos": {
    "center": [37.8, -96],
     "level": 4
              },
```

```
   "overlayLayers": [{
      "title": "US",
      "layerInfo": {
       "type": "regions"
                },
      "type": "latlng",
       "url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
             }],
   "baselayers": [{
    "title": "ArcGIS_World_Street_Map",
    "layerInfo": {
     "maxZoom": 16,
     "attribution": function() {  return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri"; }
                },
     "url": function() { return 'https://services.arcgisonline.com/ArcGIS/rest/
services/World_Street_Map/MapServer/tile/{z}/{y}/{x}'; }
             }]
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

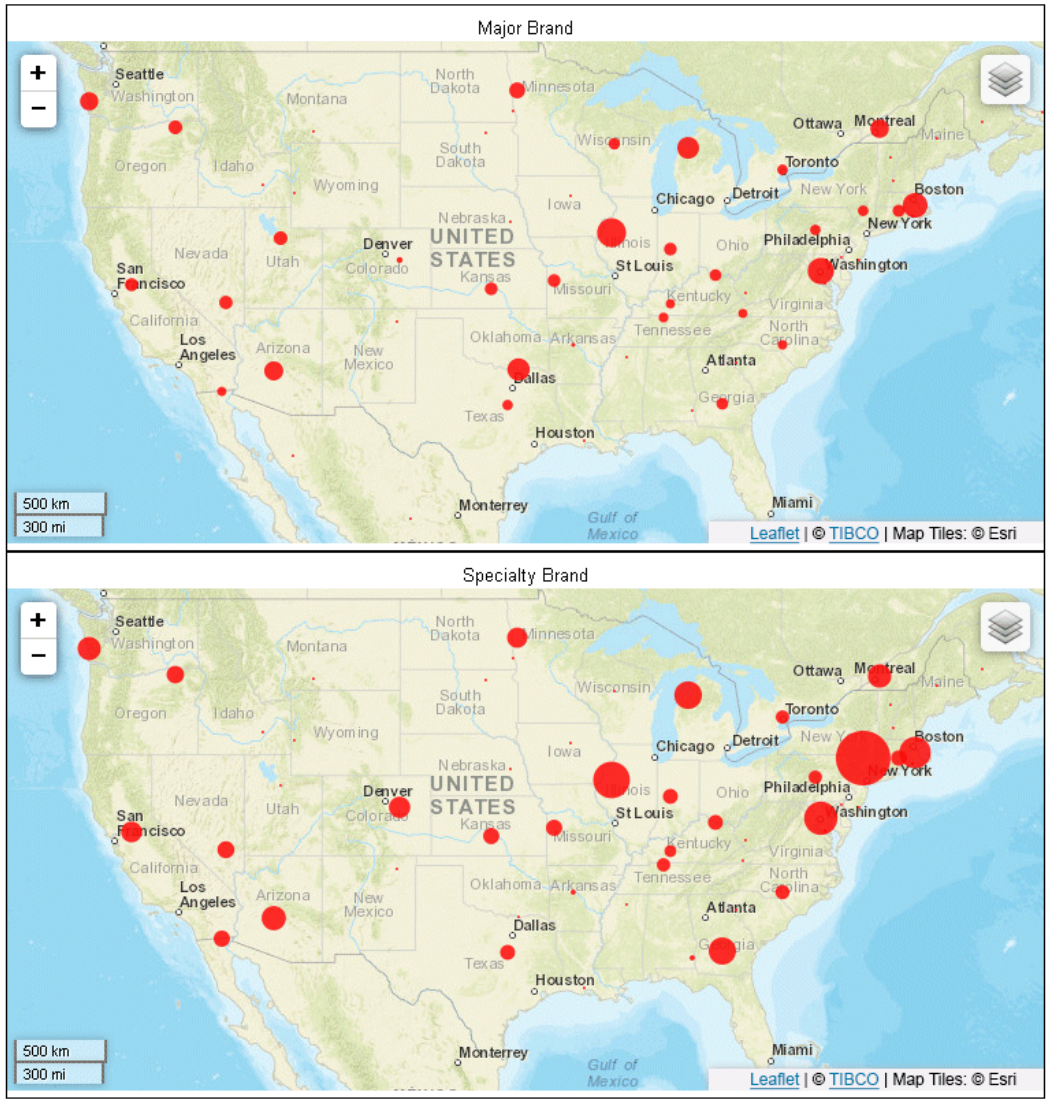
以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、BUBBLEMAP です。メジャー (MIN.CITY_POPULATION) は size の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (STATE_PROV_LATITUDE および STATE_PROV_LONGITUDE) は latitude および longitude の属性カテゴリにそれぞれ割り当てられます。出力結果をデフォルト設定の従来の構文による出力結果と同様にするため、"legend" に "visible": false のプロパティが設定され、brand types を表示するために埋め込み見出しが追加されています。

注意：以下のサンプルリクエストでは、行の長さに制限があるため、一部の構文行が次の行に折り返されている場合があります。行が折り返された場合、文字列または URL 参照内に改行が挿入されるため、このリクエストをコピーして実行する際にエラーが発生することがあります。このサンプルリクエストをコピーする場合は、実行する前に改行を削除する必要があります。

```
GRAPH FILE wf_retail_lite
HEADING CENTER
"<BRANDTYPE "
SUM MIN.CITY_POPULATION
BY BRANDTYPE
BY STATE_PROV_LATITUDE
BY STATE_PROV_LONGITUDE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH BUBBLEMAP
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=PAGE,$
TYPE=DATA, COLUMN=MIN.CITY_POPULATION, BUCKET=SIZE,$
TYPE=DATA, COLUMN=STATE_PROV_LATITUDE, BUCKET=LATITUDE,$
TYPE=DATA, COLUMN=STATE_PROV_LONGITUDE, BUCKET=LONGITUDE,$
*GRAPH_JS
"legend":{"visible":false},
"bubbleMarker":{"maxSize":"10%"},
"mapProperties": {
  "leaflet": {
   "initPos": {
    "center": [37.8, -96],
     "level": 4
             },
```

```
   "overlayLayers": [{
      "title": "US",
      "layerInfo": {
       "type": "regions"
                },
      "type": "latlng",
       "url": function(){return tdgchart.getScriptPath() + 'map/US.json'}
             }],
   "baselayers": [{
    "title": "ArcGIS_World_Street_Map",
    "layerInfo": {
     "maxZoom": 16,
     "attribution": function() {  return "&|copy; <a target='_blank' href='http://
www.tibco.com'>TIBCO</a> | " + "Map Tiles: &|copy; Esri"; }
                },
     "url": function() { return 'https://services.arcgisonline.com/ArcGIS/rest/
services/World_Street_Map/MapServer/tile/{z}/{y}/{x}'; }
             }]
    }
}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



地図に含まれる政治的境界線または地名は、必ずしも特定の係争中の境界線または地名に関する TIBCO の見解を反映するものではなく、TIBCO は、そのような境界線や地名の正確さを保証するものではありません。

マリメッコチャートリクエストのグラフ属性構文への変換

マリメッコチャートは、パーセント棒グラフに類似していますが、積み上げ棒の全体値に基づいて棒ライザの幅が決定されるという点が異なります。

参照 マリメッコチャートの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
MEKKO	MEKKO

参照 マリメッコチャートの属性カテゴリの割り当て

下表は、マリメッコチャートの属性カテゴリの変換を示しています。

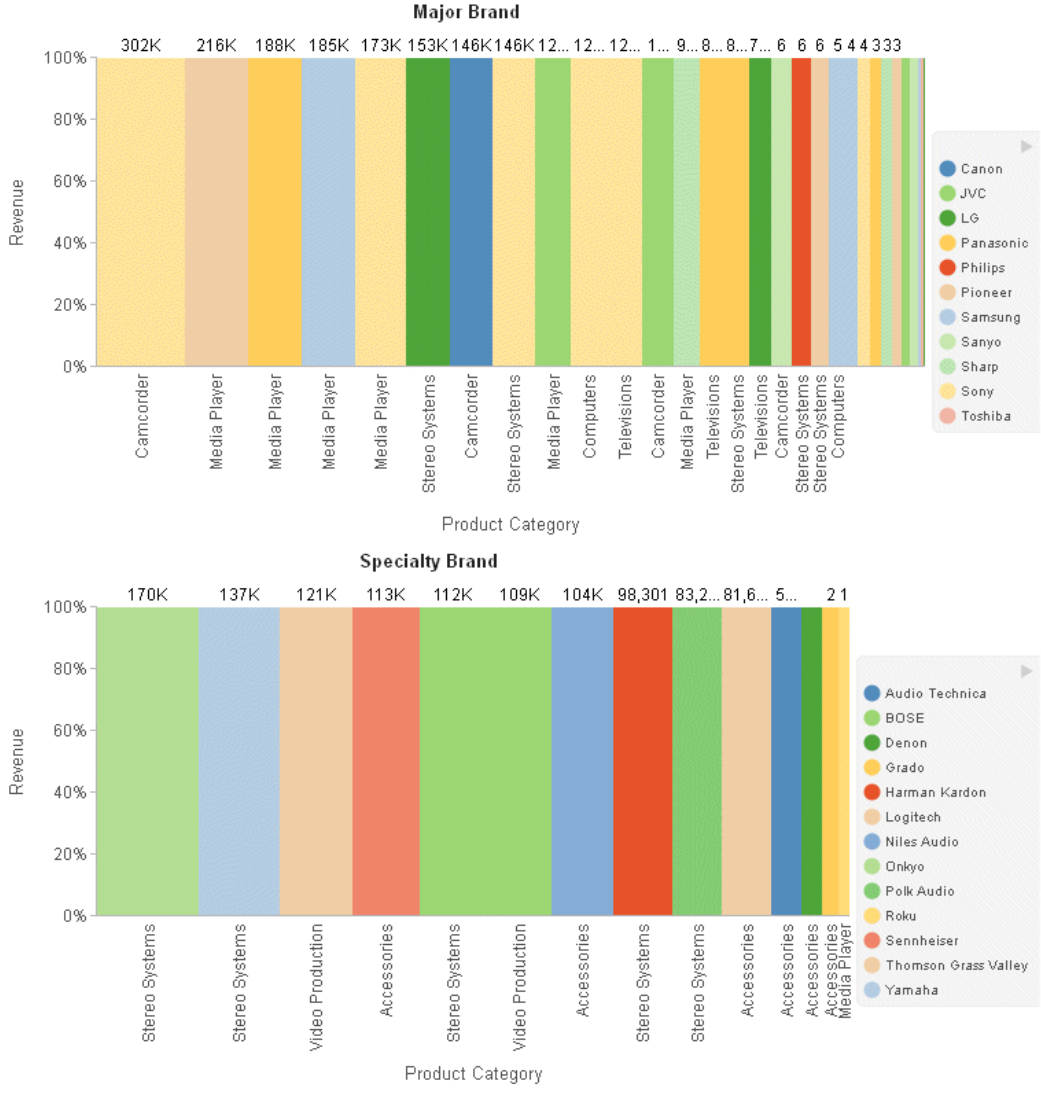
フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	y-axis
GRXAXIS ソートフィールド	x-axis
GRLEGEND ソートフィールド	color
GRMULTIGRAPH ソートフィールド	page

例　マリメッコチャートリクエストのグラフ属性構文への変換

次のリクエストは、マリメッコチャートを生成し、最も外側のソートフィールド (BRANDTYPE) の値ごとに、それぞれ個別のグラフを作成します。また、グラフ凡例に次のソートフィールド (BRAND) 配置して識別し、X 軸上には PRODUCT_SUBCATEG ソートフィールドを配置します。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US
BY BRANDTYPE
BY BRAND
BY PRODUCT_CATEGORY
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 1
ON GRAPH SET LOOKGRAPH MEKKO
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

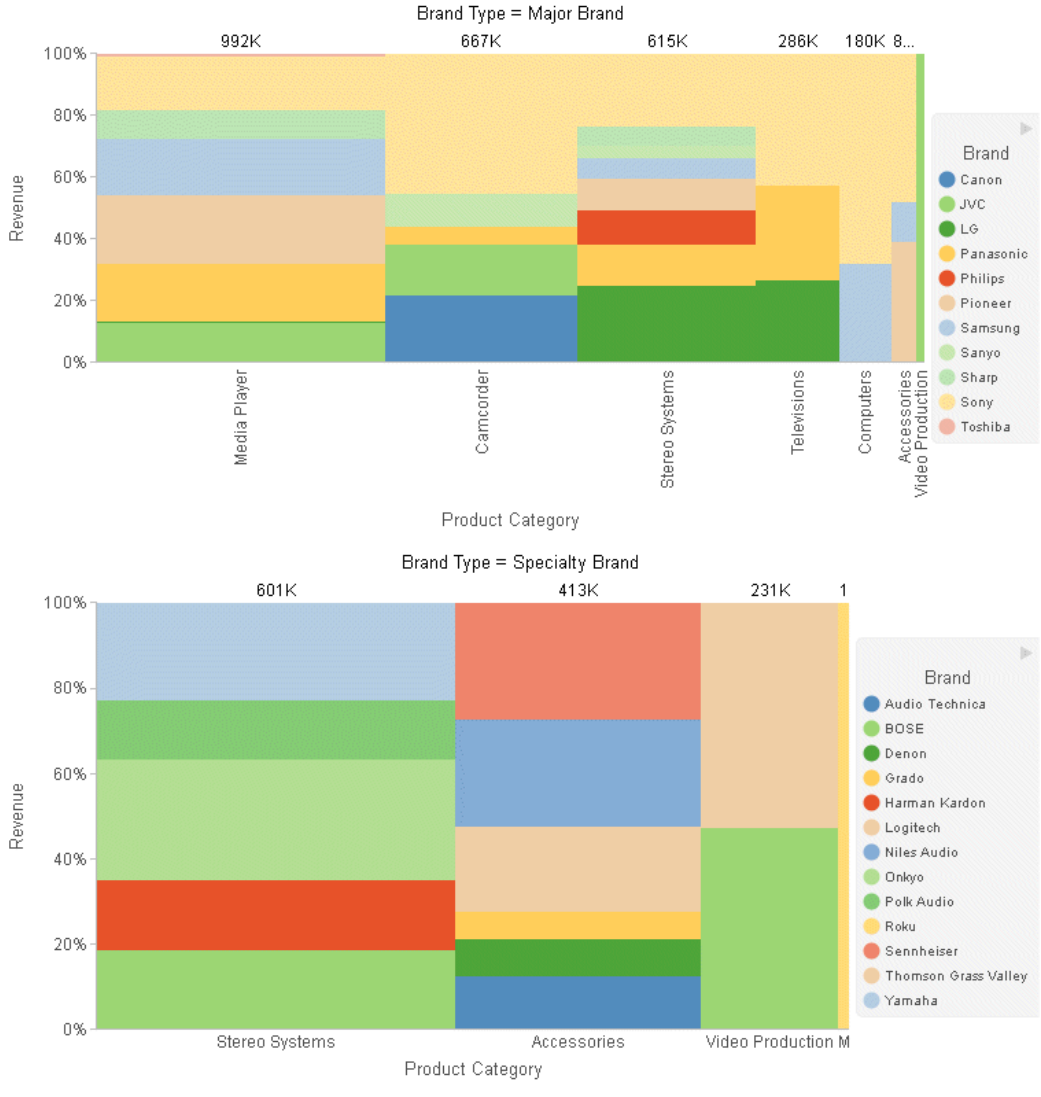
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。BRANDTYPE ソートフィールドは page、BRAND ソートフィールドは color、PRODUCT_CATEGORY ソートフィールドは x-axis のカテゴリにそれぞれ割り当てられ、メジャー REVENUE_US は y-axis のカテゴリに割り当てられます。グラフタイプは MEKKO (マリメッコチャート) です。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"Brand Type = <BRANDTYPE"
SUM REVENUE_US
BY BRANDTYPE
BY BRAND
BY PRODUCT_CATEGORY
ON GRAPH SET LOOKGRAPH MEKKO
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=page,$
TYPE=DATA, COLUMN=BRAND, BUCKET=COLOR,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=x-axis,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET=y-axis,$
END
```

下図の出力結果は、GRMERGE の出力と外観が異なります。これは、グラフ属性構文が類似した X 軸の値をクラスタ化するためです。この例では、全ブランドのメディアプレイヤが 1 本の積み上げ棒として表示されています。



円グラフリクエストのグラフ属性構文への変換

円グラフは、ソートフィールドに基づいてメジャーを項目化します。中心に穴の開いたリンググラフも円グラフの一種です。リクエストに従来の WebFOCUS 構文を使用するか、グラフ属性構文を使用するか、スタイルシートを使用するかによって、いくつかのプロパティではデフォルト設定が異なります。

参照 円グラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を、円グラフに必要な追加のプロパティとともに示しています。

LOOKGRAPH パラメータ	
PIE	PIE
PIEMULTI	PIE

PIERING	PIE
	次の JSON プロパティを追加します。
	`*GRAPH_JS`
	`"pieProperties": {`
	`"holeSize": "10%"`
	`}`
	パーセンテージを変えることで、穴の大きさを変更することができます。

PIEMULTR	PIE
	次の JSON プロパティを追加します。
	`*GRAPH_JS`
	`"pieProperties": {`
	`"holeSize": "10%"`
	`}`
	パーセンテージを変えることで、穴の大きさを変更することができます。

参照　円グラフの属性カテゴリの割り当て

下表は、円グラフの属性カテゴリの変換を示しています。

フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	measure
GRLEGEND または GRXAXIS ソートフィールドのいずれか	color、column、または row
GRMULTIGRAPH ソートフィールド	page

円グラフでは、従来の構文を使用した場合でも WebFOCUS ツールで使用できる複数ソートフィールドは限られていました。この場合、ユーザがこれらのフィールドを Category および Slices の構成要素にドラッグしていました。今後、WebFOCUS ツールでは、これらのリクエストを実行前にグラフ属性構文に変換します。これらの追加ソートフィールドの数とタイプによって、下表のように WebFOCUS ツールが column および color の属性カテゴリにソートフィールドを割り当てます。column カテゴリによって円グラフのマトリックスが生成され、単一行 (列) マトリックスの各列 (または行。構成オプションにより異なる) に 1 つの円グラフが配置されます。グラフ属性カテゴリについての詳細は、123 ページの「 TIBCO WebFOCUS グラフ構文 」 を参照してください。

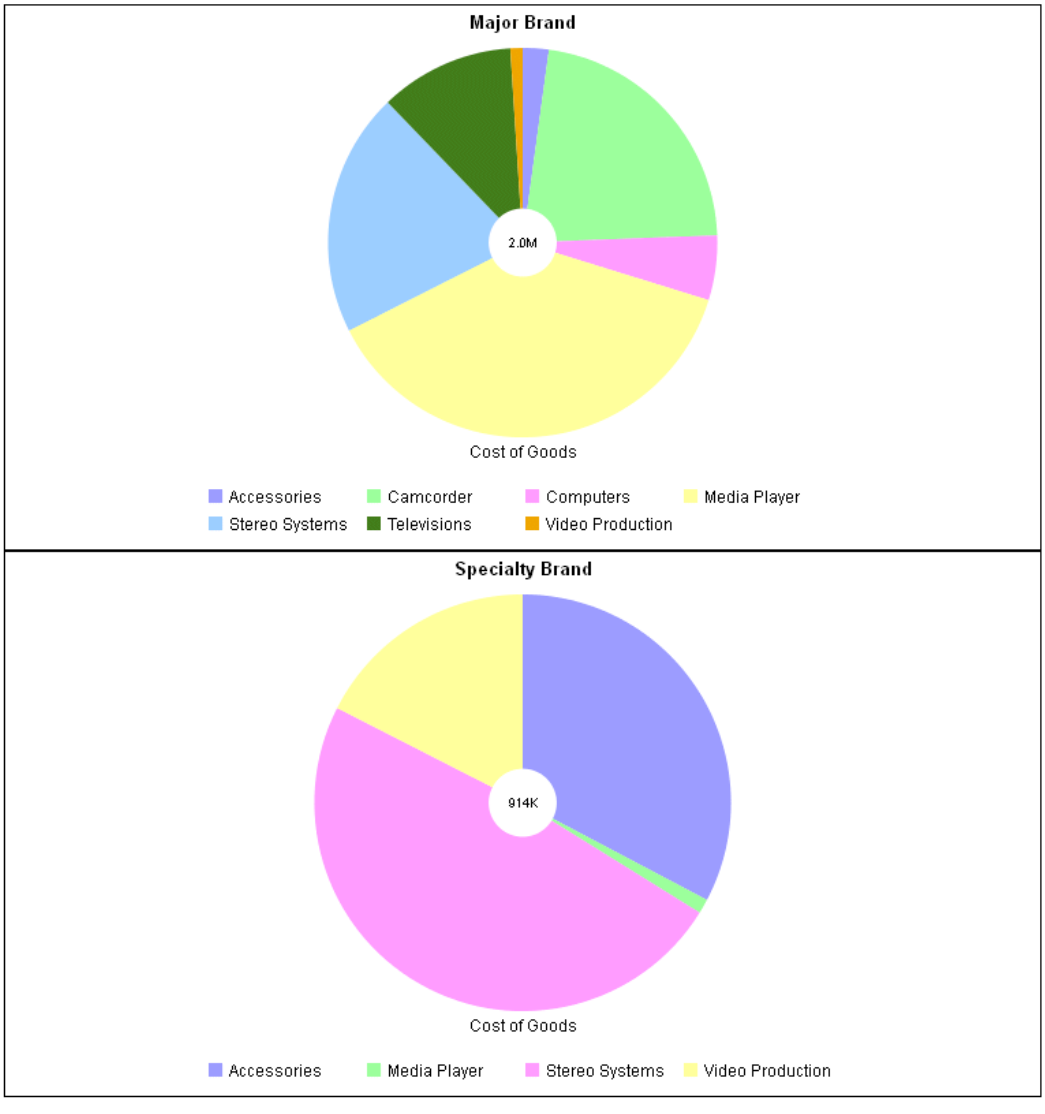
カテゴリフィールドの数	スライスフィールドの数	列の属性カテゴリへの割り当て	色の属性カテゴリへの割り当て
1	0	0	カテゴリフィールド
0	1	0	スライスフィールド
1	1	カテゴリフィールド	スライスフィールド
2	0	1 つ目のフィールド	2 つ目のフィールド
2	1	スライスフィールドの後に、1 つ目のカテゴリフィールド	2 つ目のカテゴリフィールド

例　円グラフリクエストのグラフ属性構文への変換

次のリクエストは、従来の構文を使用して複数の円グラフ (リンググラフ) を生成します (LOOKGRAPH 値は PIEMULTR) 。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別の円グラフが生成されます。PRODUCT_CATEGORY ソートフィールドは、GRLEGEND ソートフィールドです。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 0
ON GRAPH SET LOOKGRAPH PIEMULTR
END
```

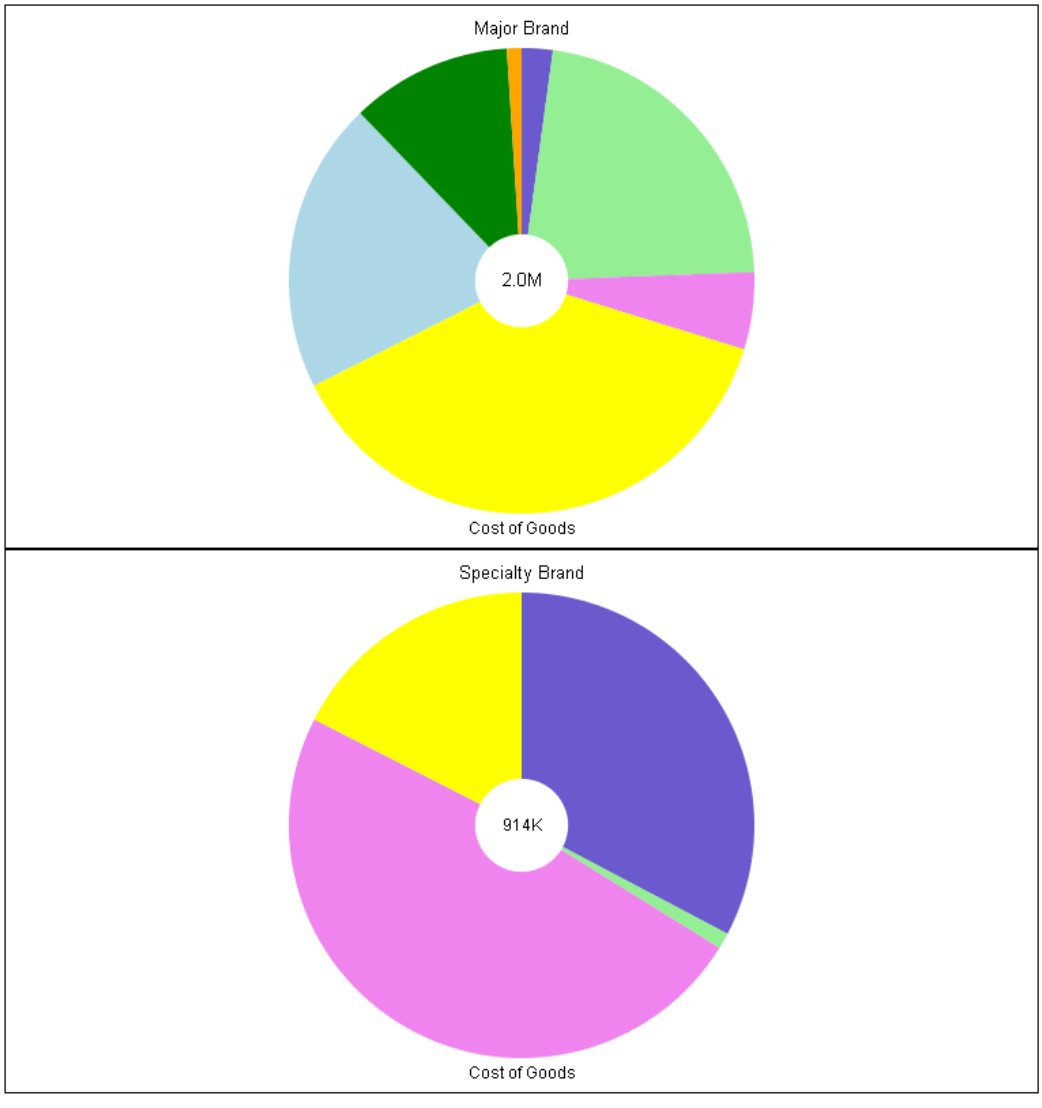
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、PIE です。メジャー (COGS_US) は measure の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (PRODUCT_CATEGORY) は color の属性カテゴリに割り当てられます。"pieProperties": "holeSize" プロパティは 20% に設定されています。円項目の色分けや凡例の非表示のために、その他いくつかの JSON プロパティが設定されています。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE"
SUM COGS_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US, BUCKET=measure,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color,$
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=page,$
*GRAPH_JS
"legend": {"visible": false},
"series": [
    {"series": 0, "color": "slateblue"},
    {"series": 1, "color": "lightgreen"},
    {"series": 2, "color": "violet"},
    {"series": 3, "color": "yellow"},
    {"series": 4, "color": "lightblue"},
    {"series": 5, "color": "green"},
    {"series": 6, "color": "orange"}],
"pieProperties": {
    "holeSize": "20%",
    "label": {"visible": true},
    "totalLabel": {"visible": true}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。



例　円グラフリクエストの円柱グラフレイアウトへの変換

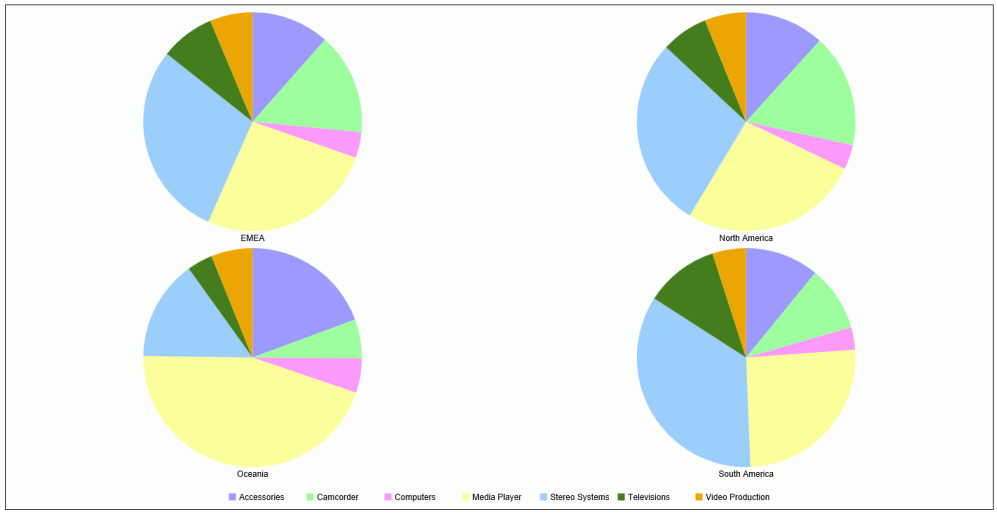
円グラフでは、従来の構文を使用した場合でも WebFOCUS ツールで使用できる複数ソートフィールドは限られていました。この場合、ユーザがこれらのフィールドを Category および Slices の構成要素にドラッグしていました。今後、WebFOCUS ツールでは、これらのリクエストを実行前にグラフ属性構文に変換します。

これらの追加ソートフィールドの数とタイプに従って、WebFOCUS ツールが列および色の属性カテゴリにソートフィールドを割り当てます。column カテゴリによって円グラフのマトリックスが生成され、単一行 (列) マトリックスの各列 (または行。構成オプションにより異なる) に 1 つの円グラフが配置されます。これにより、従来の構文で生成された、すべての円グラフが同一の枠内に収まるレイアウトとは外観が異なる複数円グラフのレイアウトが生成されます。

次のリクエストは、従来の WebFOCUS グラフ構文を使用して複数の円グラフを生成します。この場合、PRODUCT_CATEGORY ソートフィールドを GRLEGEND パラメータに、BUSINESS_REGION ソートフィールドを GRXAXIS パラメータにそれぞれ割り当てます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
ACROSS BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH SET AUTOFIT ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 0
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 1
ON GRAPH SET LOOKGRAPH PIEMULTI
END
```

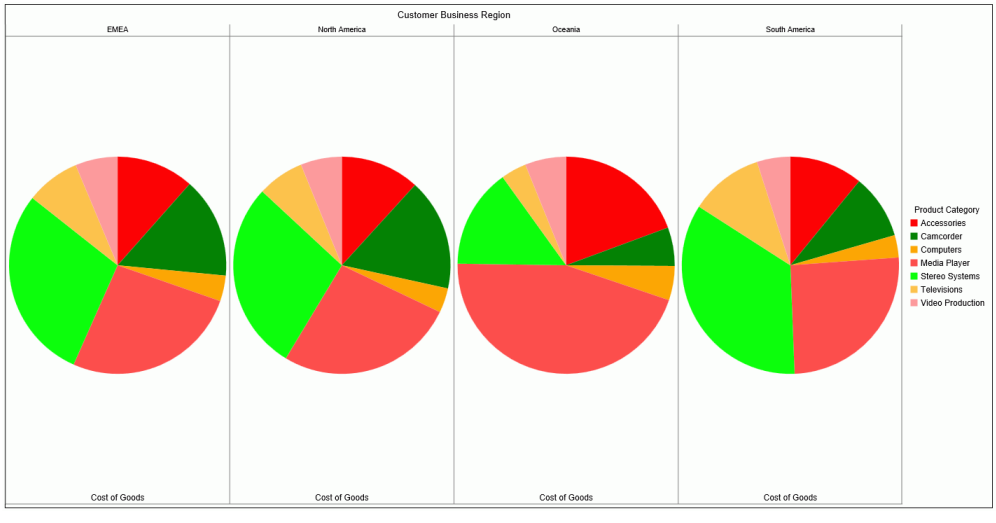
出力結果は次のとおりです。



次のリクエストでは、グラフ属性構文を使用します。PRODUCT_CATEGORY ソートフィールドは color、BUSINESS_REGION ソートフィールドは column のカテゴリにそれぞれ割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET AUTOFIT ON
ON GRAPH SET LOOKGRAPH PIE
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=COGS_US, BUCKET=measure,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color,$
TYPE=DATA, COLUMN=BUSINESS_REGION, BUCKET=column,$
END
```

出力結果は次のとおりです。円グラフはそれぞれの列に収まっています。



散布図リクエストのグラフ属性構文への変換

散布図は通常 1 つのメジャーをもう 1 つのメジャーに対してプロットしますが、WebFOCUS の LOOKGRAPH 値はディメンションもサポートします。

参照　散布図の LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
SCATTER	SCATTER
SCATTERN	SCATTER
SCATTERS	SCATTER

参照　散布図の属性カテゴリの割り当て

下表は、散布図の属性カテゴリの変換を示しています。LOOKGRAPH 値が SCATTER の場合、X 軸上には数値ソートフィールド、Y 軸上には数値メジャーが表示されます。

注意：散布図では統合パラメータは使用できません。

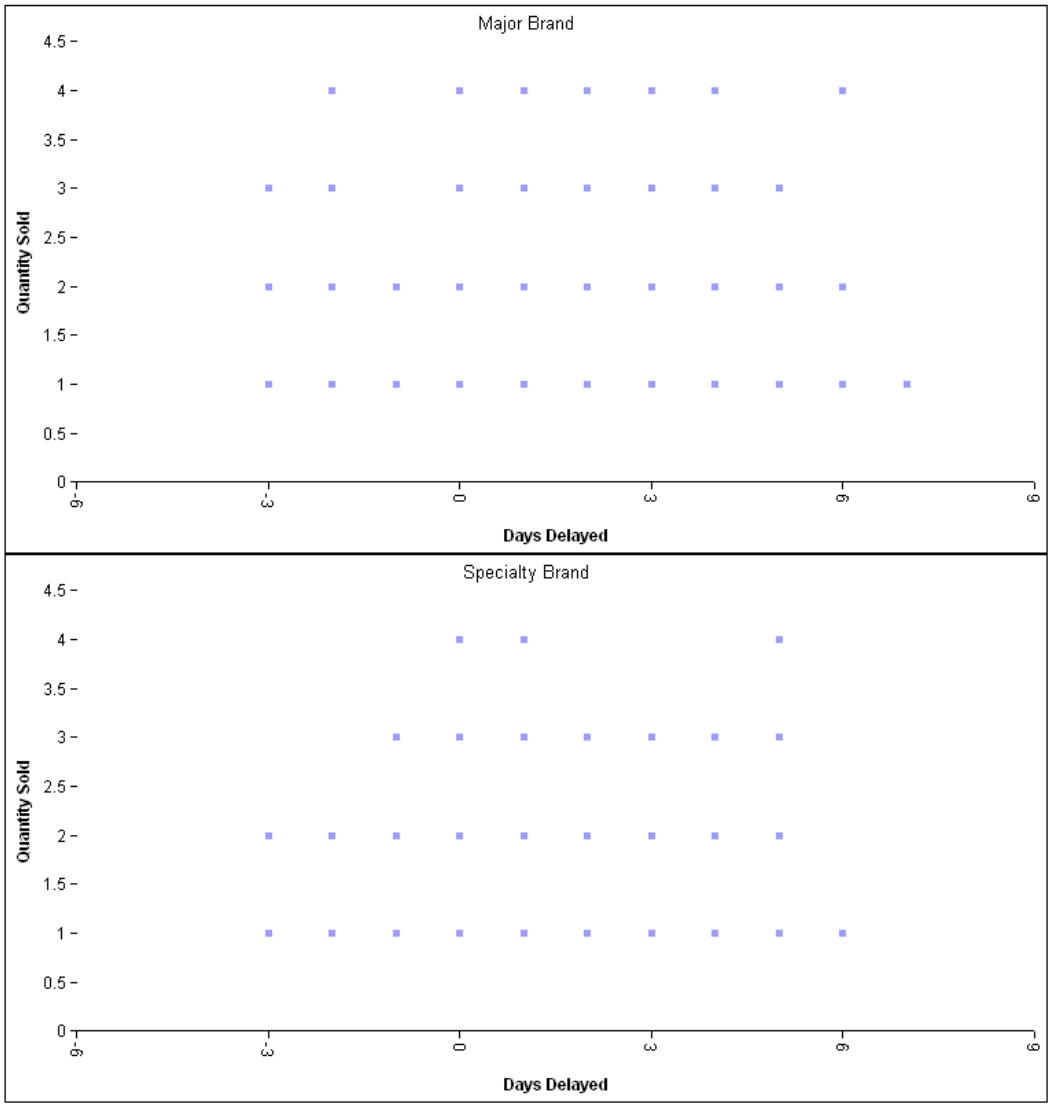
フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	y-axis
ソートフィールド SCATTER の場合は数値フォーマットにする必要があります。	x-axis
上位ソートフィールド	page

例　散布図リクエストのグラフ属性構文への変換

次のリクエストは、従来の構文を使用して複数の散布図を生成します (LOOKGRAPH 値は SCATTER)。上位ソートフィールド (BRANDTYPE) は、複数のグラフを生成するソートフィールドです。メジャー (QUANTITY_SOLD) は Y 軸、数値ソートフィールド (DAYSDELAYED) は X 軸にプロットされます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
PRINT QUANTITY_SOLD
BY BRANDTYPE
BY DAYSDELAYED
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH PCHOLD FORMAT JSCHART
END
```

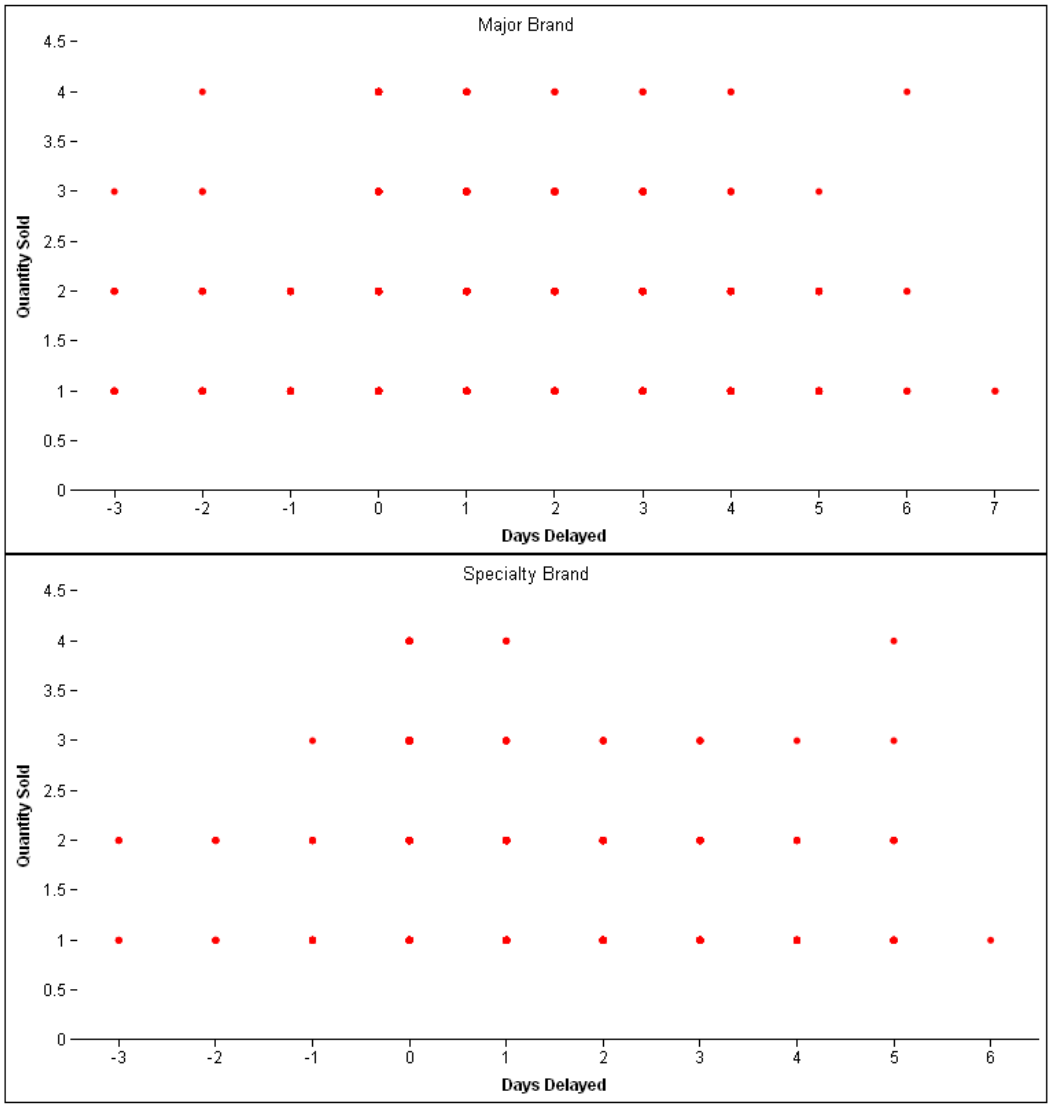
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、SCATTER です。メジャー (QUANTITY_SOLD) は y-axis、数値ソートフィールド (DAYSDELAYED) は x-axis の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は、page の属性カテゴリに割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
PRINT QUANTITY_SOLD
BY BRANDTYPE
BY DAYSDELAYED
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
type=data, column=quantity_sold, bucket=y-axis,$
type=data, column=daysdelayed, bucket=x-axis,$
type=data, column=brandtype, bucket=page,$
*GRAPH_JS
"xaxis": {"majorGrid": {"visible": false}},
"yaxis": {"majorGrid": {"visible": false}}
*END
END
```

出力結果は次のとおりです。



参照　散布図 (SCATTERN) の属性カテゴリの割り当て

LOOKGRAPH 値が SCATTERN の場合、グラフには 2 つのメジャーおよびプロットされたデータ点を表すソートフィールドが表示されます。

下表は、散布図 (SCATTERN) の属性カテゴリの変換を示しています。

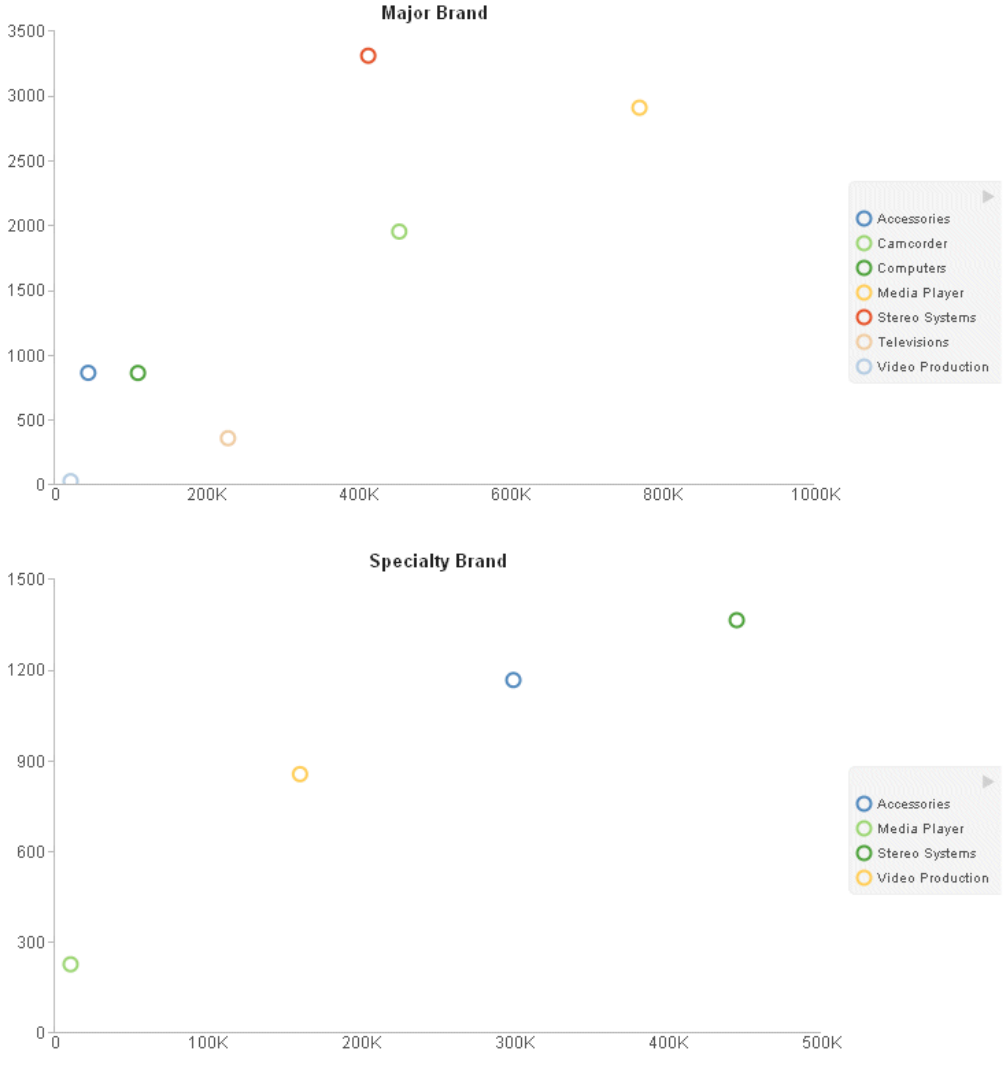
フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	x-axis
メジャーフィールド	y-axis
GRLEGEND または GRXAXIS ソートフィールド (ディメンションまたはメジャー) LOOKGRAPH 値が SCATTERN の場合、2 つのメジャーが X と Y の両軸に割り当てられ、ソートフィールドはプロットされたデータ点を表します。	color
GRMULTIGRAPH ソートフィールド	page

例 散布図 (SCATTERN) リクエストのグラフ属性構文への変換

次のリクエストは、従来の構文を使用して複数の散布図を生成します (LOOKGRAPH 値は SCATTERN) 。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別の散布図が生成されます。PRODUCT_CATEGORY ソートフィールドは、GRLEGEND ソートフィールドです。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE"
SUM COGS_US
QUANTITY_SOLD
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTERN
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 0
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

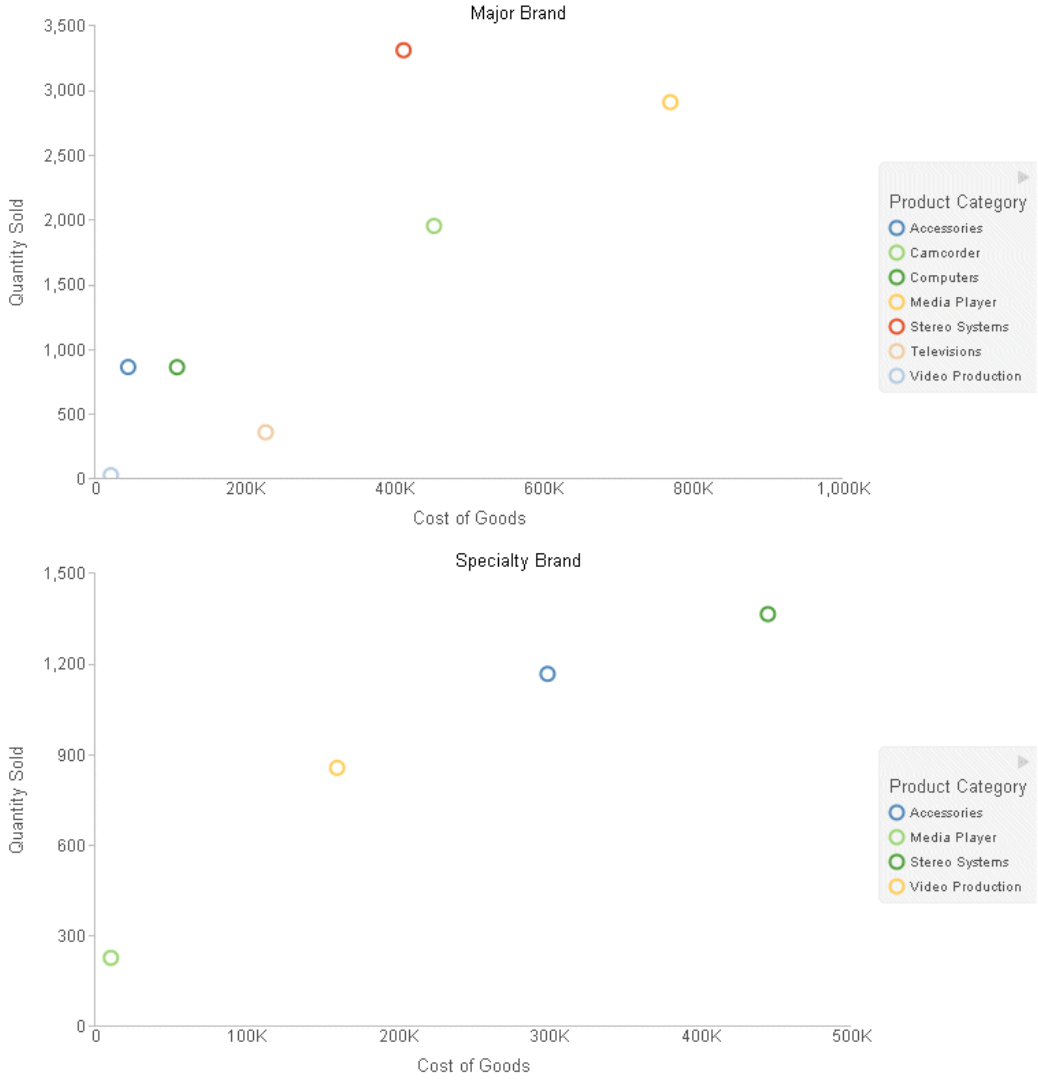
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、SCATTER です。メジャー (COGS_US および QUANTITY_SOLD) は、x-axis および y-axis の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (PRODUCT_CATEGORY) は color の属性カテゴリにそれぞれ割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
SUM COGS_US QUANTITY_SOLD
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=QUANTITY_SOLD, BUCKET=y-axis,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=x-axis,$
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=page,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=color,$
END
```

出力結果は次のとおりです。



参照　散布図 (SCATTERS) の属性カテゴリの割り当て

LOOKGRAPH 値が SCATTERS の場合、Y 軸上に任意の数のメジャーをプロットすることができ、ソートフィールドは X 軸 になります。ただし、グラフ属性構文では Y 軸のメジャーは 1 つしかサポートされません。

注意：LOOKGRAPH 値が SCATTERS の散布図では統合パラメータは使用できません。

下表は、散布図 (SCATTERS) の属性カテゴリの変換を示しています。

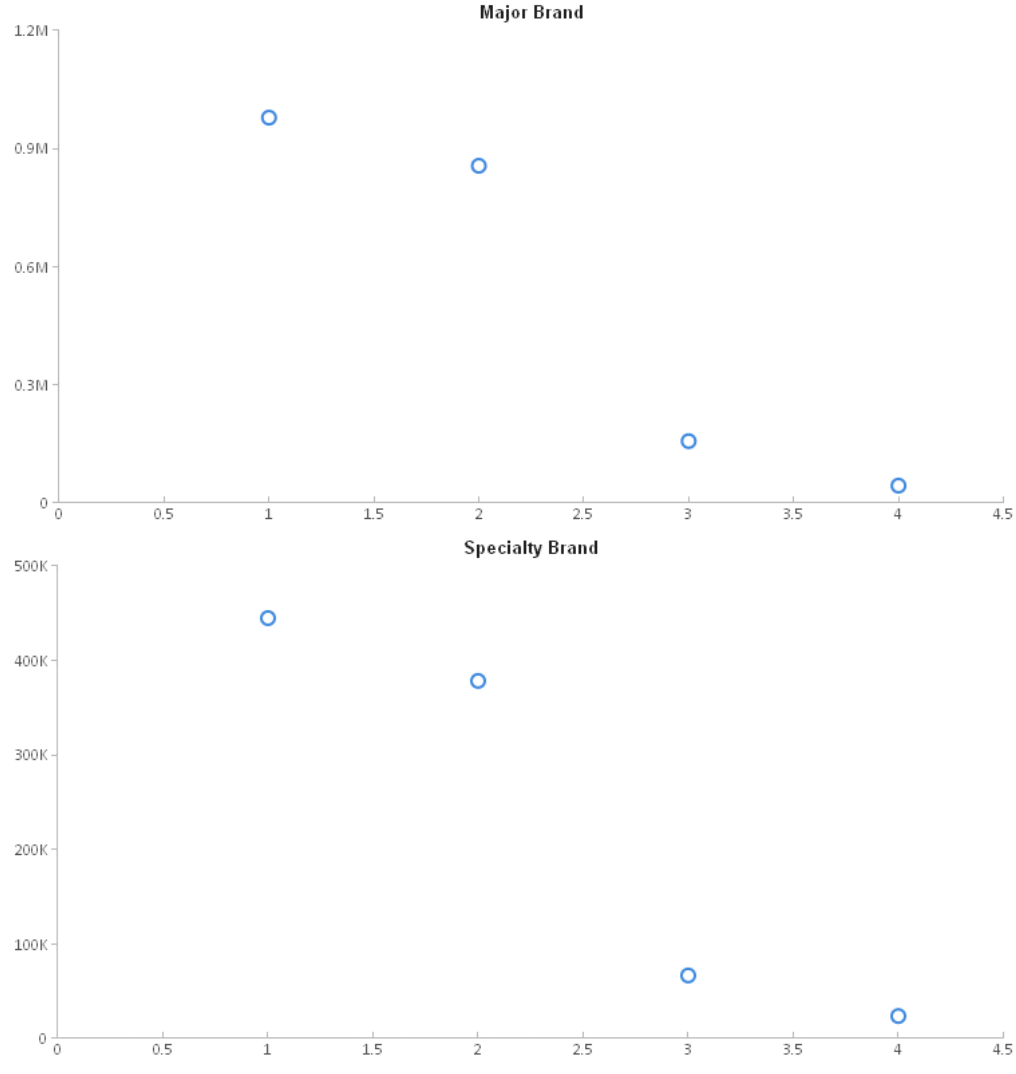
フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド (任意の数)	y-axis サポートされる Y 軸のメジャーは 1 つのみです。
ソートフィールド (メジャーまたはディメンション) SCATTERS では、ソートフィールドが X 軸になります。	x-axis
上位ソートフィールド	page

例　散布図 (SCATTERS) リクエストのグラフ属性構文への変換

次のリクエストは、従来の構文を使用して複数の散布図を生成します (LOOKGRAPH 値は SCATTERS)。上位ソートフィールド (BRANDTYPE) は、複数のグラフを生成するソートフィールドです。QUANTITY_SOLD ソートフィールドは X 軸を表します。メジャー COGS_US は、Y 軸にプロットされます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE"
SUM COGS_US
BY BRANDTYPE
BY QUANTITY_SOLD
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTERS
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

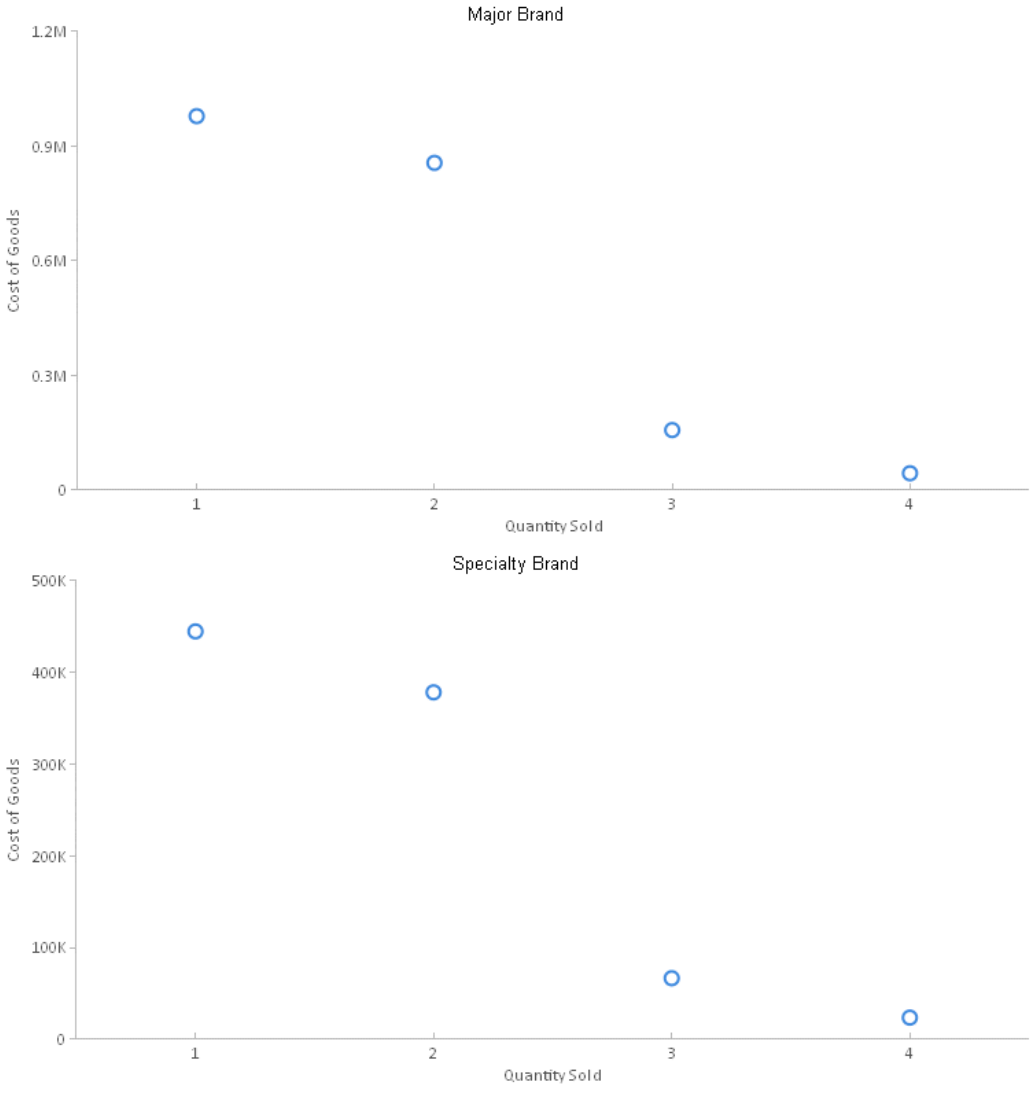
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、SCATTER です。メジャー (COGS_US) は y-axis 属性カテゴリに割り当てられ、下位ソートフィールド (QUANTITY_SOLD) は x-axis、上位ソートフィールド (BRANDTYPE) は page の属性カテゴリにそれぞれ割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE"
SUM COGS_US
BY BRANDTYPE
BY QUANTITY_SOLD
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH SET STYLE *
type=data, column=cogs_us, bucket=y-axis,$
type=data, column=quantity_sold, bucket=x-axis,$
type=data, column=brandtype, bucket=page,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。



ブロック地図リクエストのヒートマップリクエストへの変換

ヒートマップは、メジャー値を表す色で各セルを塗りつぶしたマトリックスです。

従来の WebFOCUS グラフ構文は、ヒートマップのグラフタイプをサポートしません。その代わりに類似のブロック地図をサポートしています。ブロック地図は、関連性のない複数のメジャーをサポートしますが、ヒートマップは単一のメジャーのみサポートします。複数のメジャーを含むブロック地図を変換するには、各メジャーに対して個別のヒートマップリクエストを発行する必要があります。

参照 ヒートマップの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
SPECTRAL	HEATMAP

参照 ヒートマップの属性カテゴリの割り当て

下表は、ヒートマップの属性カテゴリの変換を示しています。GRLEGEND と GRXAXIS のパラメータは同じ意味で使用されます。

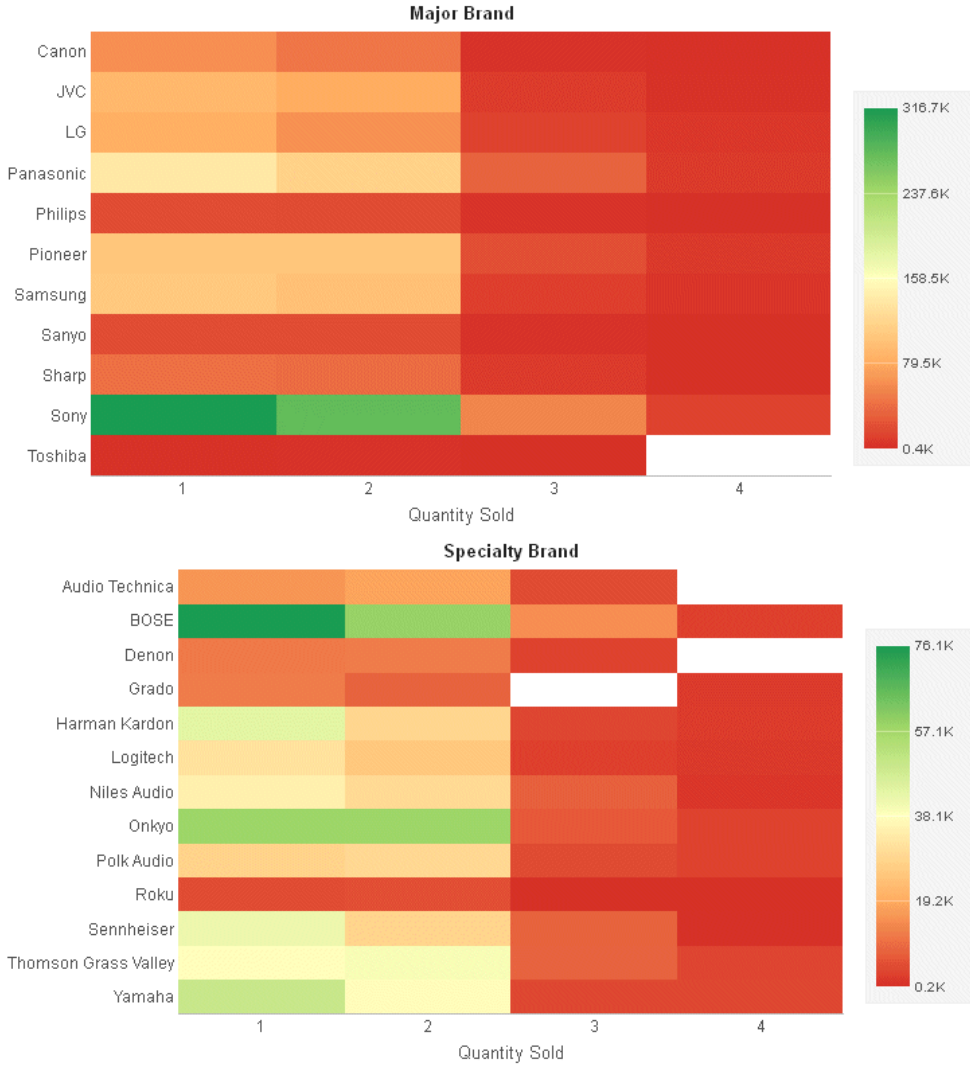
フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	color
GRLEGEND ソートフィールド	y-axis
GRXAXIS (ACROSS) ソートフィールド	x-axis
GRMULTIGRAPH ソートフィールド	page

例　ブロック地図リクエストのヒートマップリクエストへの変換

次のリクエストは、従来の構文を使用して複数のブロック地図を生成します (LOOKGRAPH 値は SPECTRAL)。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のブロック地図が生成されます。BRAND ソートフィールドは Y 軸を生成し、QUANTITY_SOLD ソートフィールドは X 軸を生成します。メジャー (COGS_US) は、長方形の色に使用されます。VZERO パラメータは、グラフ属性構文リクエストのデフォルト値に合わせて設定されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY BRANDTYPE
BY BRAND
ACROSS QUANTITY_SOLD
ON GRAPH SET VZERO OFF
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 1
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH SPECTRAL
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

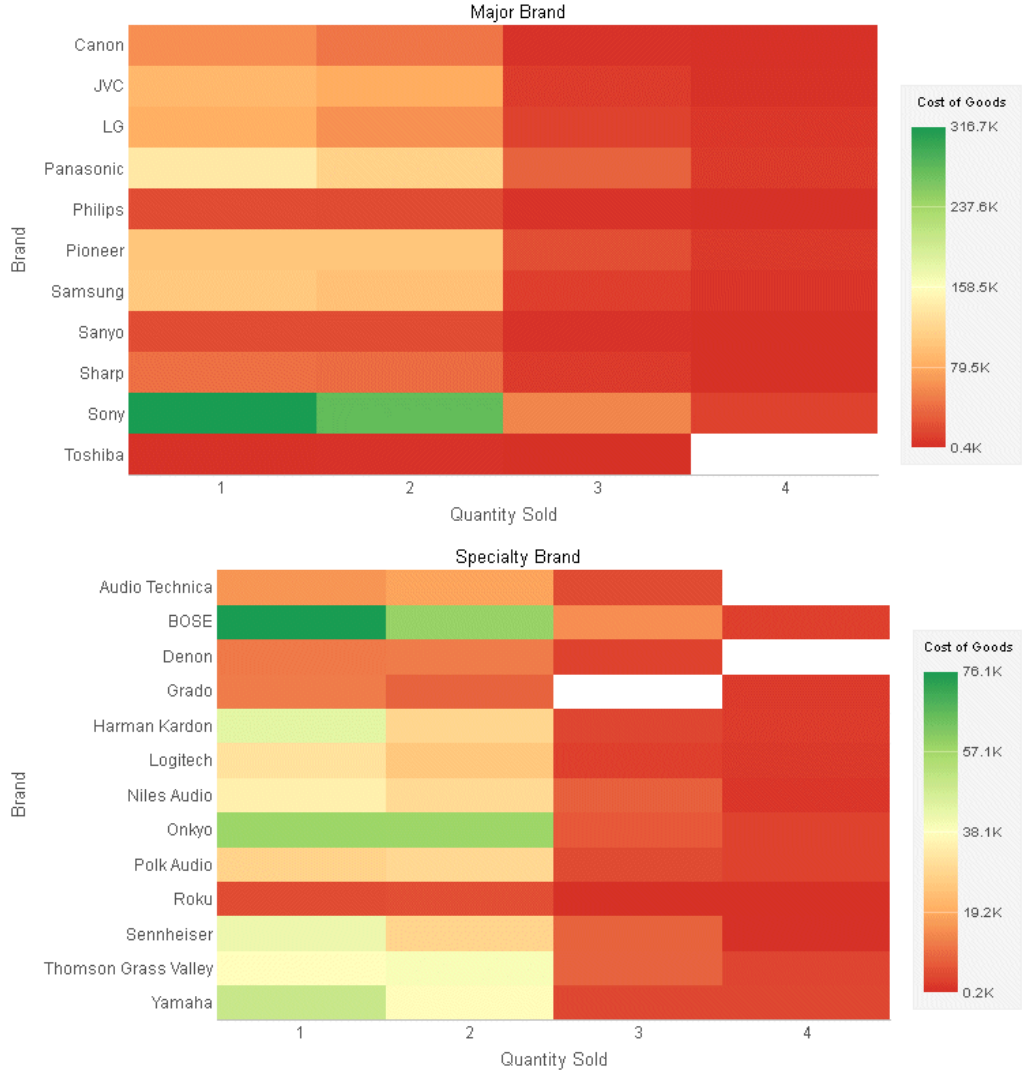
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、HEATMAP です。メジャー (COGS_US) は color、上位ソートフィールド (BRANDTYPE) は page の属性カテゴリに割り当てられ、BRAND ソートフィールドは y-axis、QUANTITY_SOLD ソートフィールドは x-axis の属性カテゴリにそれぞれ割り当てられます。ACROSS 句は BY 句に置き換えられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE "
SUM COGS_US
BY BRANDTYPE
BY BRAND
BY QUANTITY_SOLD
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET EMBEDHEADING ON
ON GRAPH SET LOOKGRAPH HEATMAP
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
type=data, column=brandtype, bucket=page,$
type=data, column=cogs_us, bucket=color,$
type=data, column=brand, bucket=y-axis,$
type=data, column=quantity_sold, bucket=x-axis,$
ENDSTYLE
END
```

出力結果は次のとおりです。



ストリームグラフリクエストのグラフ属性構文への変換

ストリームグラフは、積み上げ面グラフを簡素化したグラフです。

参照　ストリームグラフの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
STREAM	STREAM

参照　ストリームグラフの属性カテゴリの割り当て

下表は、ストリームグラフの属性カテゴリの変換を示しています。

フィールドまたはパラメータのタイプ	属性カテゴリ
メジャーフィールド	y-axis
ソートフィールド	x-axis

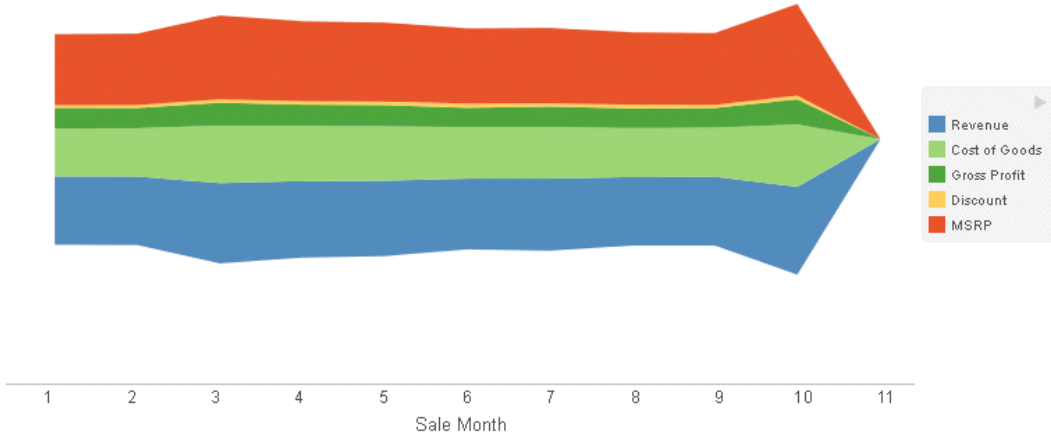
注意：ストリームグラフでは統合パラメータは使用できません。

例　ストリームグラフリクエストのグラフ属性構文への変換

次のリクエストは、ストリームグラフを生成します。メジャーは、REVENUE_US、COGS_US、GROSS_PROFIT_US、DISCOUNT_US、MSRP_US です。ソートフィールドは、TIME_MTH です。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US COGS_US GROSS_PROFIT_US DISCOUNT_US MSRP_US
BY TIME_MTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STREAM
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

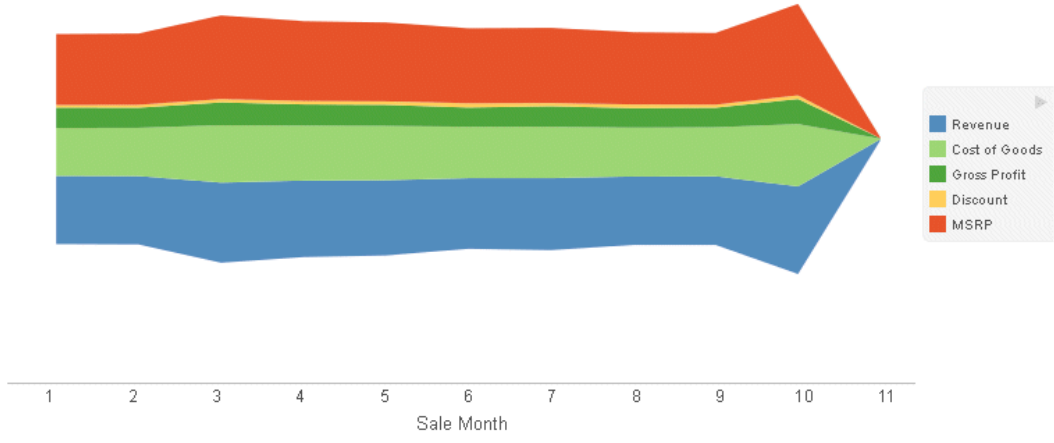
出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。TIME_MTH ソートフィールドが x-axis、メジャーが y-axis のカテゴリにそれぞれ割り当てられています。

```
GRAPH FILE WF_RETAIL_LITE
SUM REVENUE_US COGS_US GROSS_PROFIT_US DISCOUNT_US MSRP_US
BY TIME_MTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH STREAM
ON GRAPH SET STYLE *
type=data, column=revenue_us, bucket=y-axis,$
type=data, column=COGS_US, bucket=y-axis,$
type=data, column=GROSS_PROFIT_US, bucket=y-axis,$
type=data, column=DISCOUNT_US, bucket=y-axis,$
type=data, column=MSRP_US, bucket=y-axis,$
type=data, column=time_mth, bucket=x-axis,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。



タグクラウドリクエストのグラフ属性構文への変換

タグクラウドは、グループのラベルをグラフ化したものです。データ値に比例して、各ラベルのサイズが変化します。2 番目のメジャーはラベルの色に使用されます。

参照 タグクラウドの LOOKGRAPH 変換

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
TAGCLOUD	TAGCLOUD

参照 タグクラウドの属性カテゴリの割り当て

下表は、タグクラウドの属性カテゴリの変換を示しています。GRLEGEND と GRXAXIS のパラメータは同じ意味で使用されます。

フィールドまたはパラメータのタイプ	属性カテゴリ
1 番目のメジャーフィールド	size

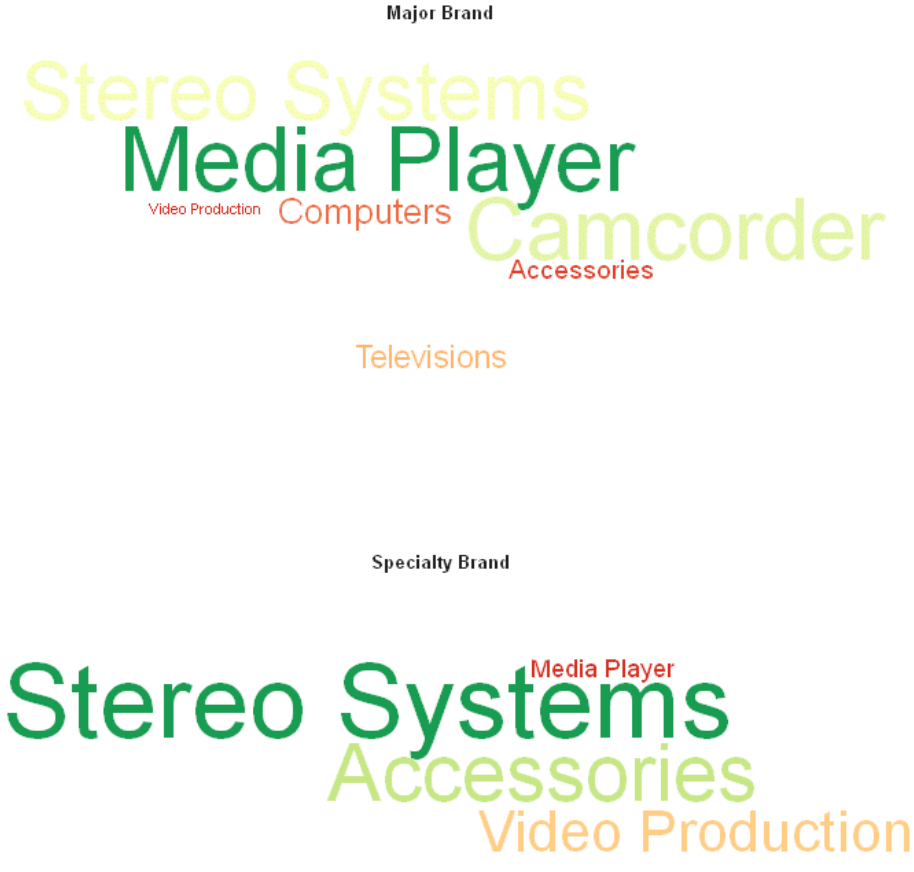
フィールドまたはパラメータのタイプ	属性カテゴリ
2 番目のメジャーフィールド (存在する場合)	color
GRLEGEND または GRXAXIS ソートフィールド	detail
GRMULTIGRAPH ソートフィールド	page

例　タグクラウドリクエストのグラフ属性構文への変換

次のサンプルリクエストは、従来の構文を使用して複数のタグクラウドを生成します (LOOKGRAPH 値は TAGCLOUD)。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のタグクラウドが生成されます。PRODUCT_CATEGORY ソートフィールドは、ラベルを生成します。1 番目のメジャー (GROSS_PROFIT_US) は、ラベルのサイズに使用され、2 番目のメジャー (COGS_US) は、ラベルの色に使用されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM GROSS_PROFIT_US COGS_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TAGCLOUD
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、TAGCLOUD です。メジャー (GROSS_PROFIT_US および COGS_US) は、size および color の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (PRODUCT_CATEGORY) は detail の属性カテゴリに割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
" <BRANDTYPE </1"
SUM GROSS_PROFIT_US COGS_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
ON GRAPH SET EMBEDHEADING ON
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TAGCLOUD
ON GRAPH SET STYLE *
type=data, column=brandtype, bucket=page,$
type=data, column=product_category, bucket=detail,$
type=data, column=gross_profit_us, bucket=size,$
type=data, column=cogs_us, bucket=color,$
type=heading, font=arial, style=bold,$
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。

ツリーマップリクエストのグラフ属性構文への変換

ツリーマップは、ネストされた長方形の集合体として階層データを色分けして表示します。

参照 **ツリーマップの LOOKGRAPH 変換**

下表は、従来の LOOKGRAPH 値と新しい LOOKGRAPH 値を示しています。

LOOKGRAPH パラメータ	
TREEMAP	TREEMAP

参照　ツリーマップの属性カテゴリの割り当て

下表は、ツリーマップの属性カテゴリの変換を示しています。GRLEGEND と GRXAXIS のパラメータは同じ意味で使用されます。

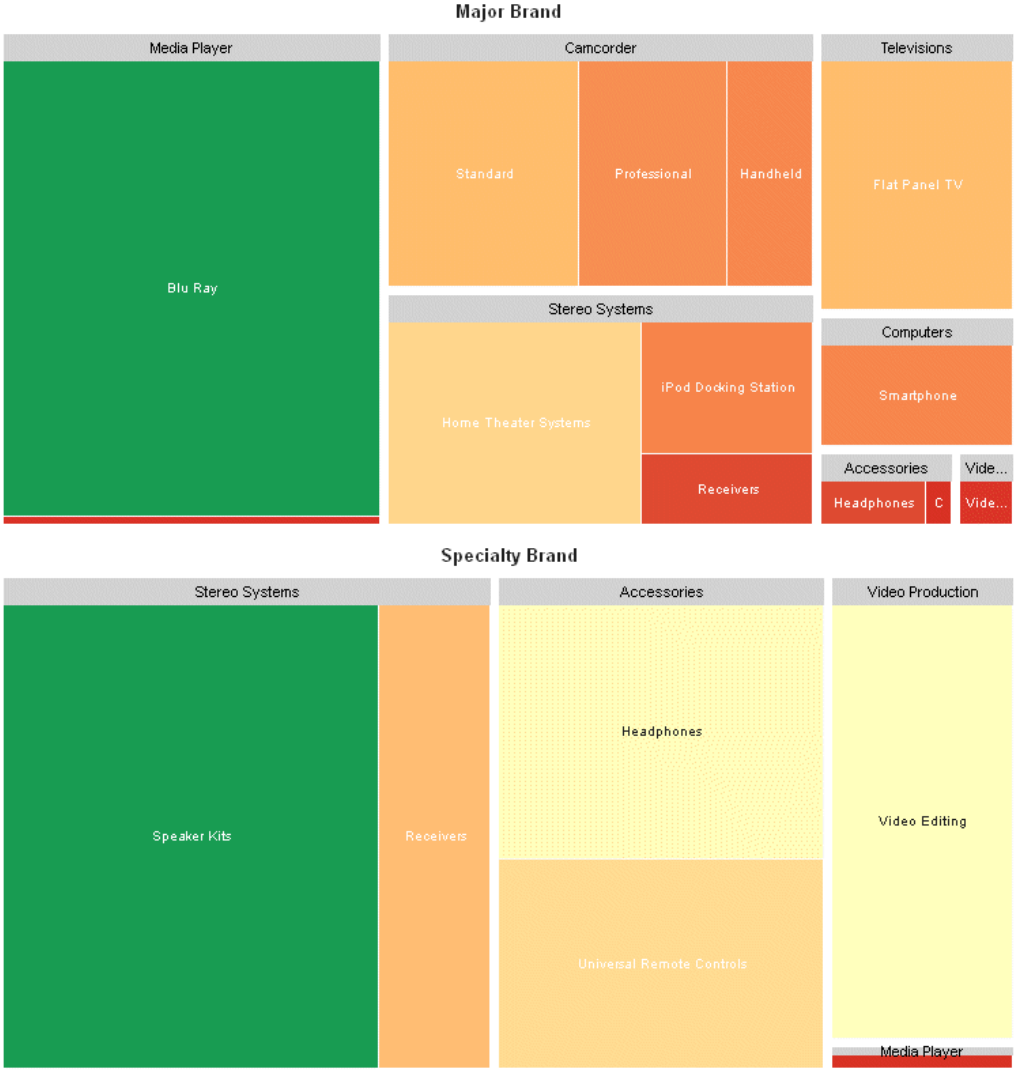
フィールドまたはパラメータのタイプ	属性カテゴリ
1 番目のメジャーフィールド	size
2 番目のメジャーフィールド (存在する場合)	color
GRLEGEND ソートフィールド	detail
GRXAXIS ソートフィールド	detail
GRMULTIGRAPH ソートフィールド	page

例　ツリーマップリクエストのグラフ属性構文への変換

次のサンプルリクエストは、従来の構文を使用して複数のツリーマップを生成します (LOOKGRAPH 値は TREEMAP)。上位ソートフィールド (BRANDTYPE) は、GRMULTIGRAPH ソートフィールドです。BRANDTYPE の値ごとに個別のツリーマップが生成されます。PRODUCT_CATEGORY と PRODUCT_SUBCATEG のソートフィールドは、ネストされた長方形を生成します。1 番目のメジャー (COGS_US) は、長方形のサイズに使用され、2 番目のメジャー (REVENUE_US) は、長方形の色に使用されます。

```
GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
REVENUE_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 1
ON GRAPH SET GRLEGEND 1
ON GRAPH SET GRXAXIS 1
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
ENDSTYLE
END
```

出力結果は次のとおりです。



以下は同じリクエストをグラフ属性構文に変換したものです。LOOKGRAPH 値は、TREEMAP です。メジャー (COGS_US および REVENUE_US) は、size および color の属性カテゴリに割り当てられ、上位ソートフィールド (BRANDTYPE) は page、下位ソートフィールド (PRODUCT_CATEGORY および PRODUCT_SUBCATEG) は detail の属性カテゴリに割り当てられます。

```
GRAPH FILE WF_RETAIL_LITE
HEADING CENTER
"<BRANDTYPE"
SUM COGS_US
REVENUE_US
BY BRANDTYPE
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH TREEMAP
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
TYPE=DATA, COLUMN=BRANDTYPE, BUCKET=PAGE,$
TYPE=DATA, COLUMN=COGS_US, BUCKET=size,$
TYPE=DATA, COLUMN=REVENUE_US, BUCKET = color,$
TYPE=DATA, COLUMN=PRODUCT_CATEGORY, BUCKET=DETAIL,$
TYPE=DATA, COLUMN=PRODUCT_SUBCATEG, BUCKET=DETAIL,$
TYPE=HEADING, STYLE=BOLD,$
*GRAPH_JS
"legend": {"visible": false}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。

ネストされた X 軸を含むリクエストのグラフ属性構文への変換

従来の WebFOCUS グラフ構文では、複数のソートフィールドが存在し、GRXAXIS パラメータが 1 より大きい数に設定されている場合に、ネストされた X 軸が生成されます。グラフ属性構文では、複数の階層ソートフィールドが X 軸の属性カテゴリに割り当てられた場合に、ネストされた X 軸が生成されます。

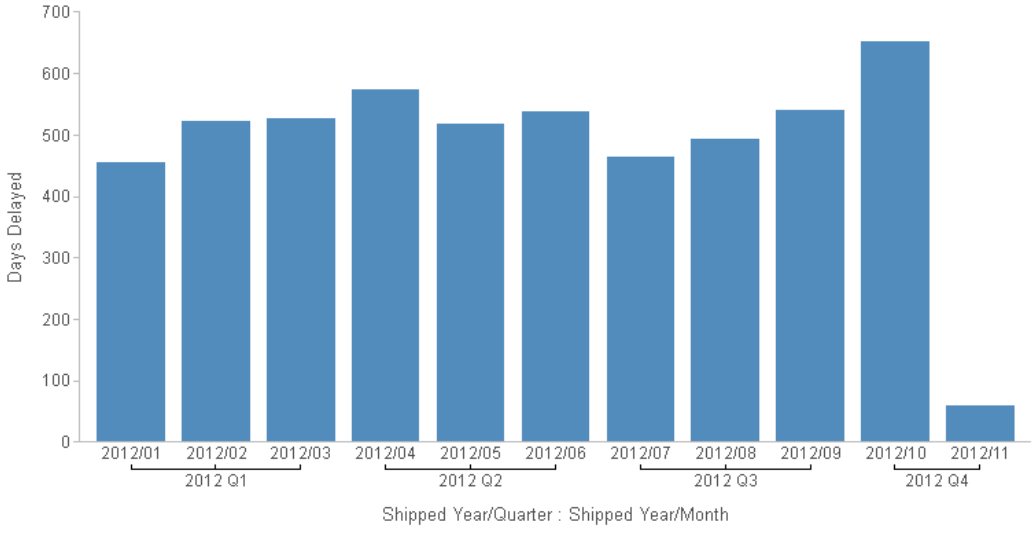
X 軸ラベルに連結文字列を設定することで、ネストされた X 軸を連結されたフィールド値として表示することができます。詳細は、315 ページの 「 軸のプロパティ 」 を参照してください。

例　ネストされた X 軸を含むリクエストのグラフ属性構文への変換

次のリクエストは、従来の WebFOCUS グラフ構文を使用してネストされた X 軸を生成します。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY TIME_DATE_QTR
BY TIME_DATE_MONTH
ON GRAPH SET GRMERGE ADVANCED
ON GRAPH SET GRMULTIGRAPH 0
ON GRAPH SET GRLEGEND 0
ON GRAPH SET GRXAXIS 2
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VBAR
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"xaxis": {"labels": {"nestingConcatSymbol": null}}
*END
ENDSTYLE
END
```

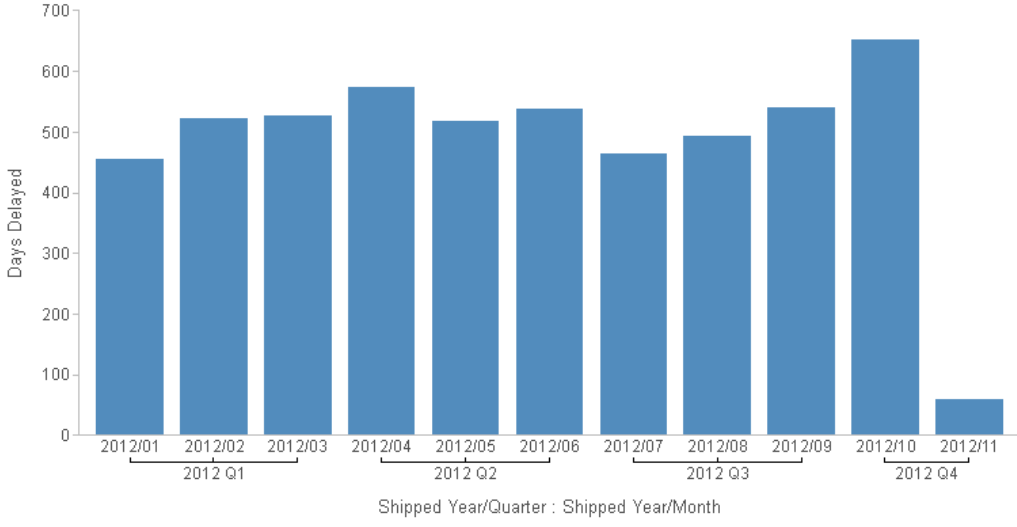
出力結果は次のとおりです。



次のリクエストでは、X 軸の属性カテゴリに両方のソートフィールドを割り当てます。

```
GRAPH FILE WF_RETAIL_LITE
SUM DAYSDELAYED
BY TIME_DATE_QTR
BY TIME_DATE_MONTH
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH BAR
ON GRAPH SET STYLE *
TYPE=DATA, COLUMN=DAYSDELAYED, BUCKET=y-axis, $
TYPE=DATA, COLUMN=TIME_DATE_QTR, BUCKET=x-axis, $
TYPE=DATA, COLUMN=TIME_DATE_MONTH, BUCKET=x-axis, $
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENWarm.sty,$
*GRAPH_JS
"xaxis": {"labels": {"nestingConcatSymbol": null}}
*END
ENDSTYLE
END
```

出力結果は次のとおりです。


700
600
500
400
300
200
100
0
Days Delayed
2012/01
2012/02
2012/03
2012/04
2012/05
2012/06
2012/07
2012/08
2012/09
2012/10
2012/11
2012 Q1
2012 Q2
2012 Q3
2012 Q4
Shipped Year/Quarter : Shipped Year/Month

C

JSON プロパティの変更累積リスト

ここでは、最新リリースを含むすべてのバージョンで変更または廃止になった JSON プロパティについて記載します。

- WebFOCUS バージョン 8.2 での JSON プロパティの変更
- WebFOCUS バージョン 8.1 での JSON プロパティの変更

WebFOCUS バージョン 8.2 での JSON プロパティの変更

以下は、WebFOCUS バージョン 8.2 で実装された JSON プロパティの変更を示したものです。

- データテキストラベルのプロパティ (dataLabels) は、グラフ全体のプロパティからシリーズ固有のプロパティに変更されました。"dataLabels": "displayMode" および "dataLabels": "callback" のプロパティは、content プロパティに置き換えられました。詳細は、427 ページの 「 シリーズ固有のプロパティ 」 を参照してください。
- シリーズ固有の showDataValues プロパティは廃止されました。
- グラフ全体の axisAutoLayout プロパティは廃止されました。
- バージョン 8.2.02 では、グローバルな傾向線を設定する line プロパティが、lineStyle プロパティに置き換えられました。
- バージョン 8.2.02 では、凡例の位置を設定する free プロパティおよび xy プロパティが廃止されました。凡例の位置に、JSON オブジェクトを使用できるようになりました。詳細は、290 ページの 「 凡例の位置の制御 」 を参照してください。
- バージョン 8.2.05 では、"colorScale": "colorBands" プロパティが廃止され、すべての色モードで "colorScale": "colors" への開始値と終了値の追加に置き換えられました。

WebFOCUS バージョン 8.1 での JSON プロパティの変更

以下は、WebFOCUS バージョン 8.1 で実装された JSON プロパティの変更を示したものです。

- xaxisNumeric および xaxisOrdinal は、xaxis に置き換えられました。
- zaxisOrdinal は、zaxis に置き換えられました。

- fillMode は、fillEffect に置き換えられました。
- "pieProperties": "feelerLine" は、"dataLabels": "feelerLine" に置き換えられました。
- "yaxis": "colorScale" および "heatmapProperties": "dataColors" は、colorScale に置き換えられました。

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

ibi, the ibi logo, ActiveMatrix BusinessWorks, BusinessConnect, Enterprise Message Service, FOCUS, Hawk, iWay, Maporama, Omni-Gen, Omni-HealthData, TIBCO, the TIBCO logo, the TIBCO O logo, TIBCO Administrator, TIBCO Designer, and WebFOCUS are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (*https://www.cloud.com/legal*) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: *https://scripts.sil.org/OFL*.

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at *https://www.tibco.com/patents*.

Copyright © 2023. Cloud Software Group, Inc. All Rights Reserved.