



# CHART TABLE SNAPSHOT

## PRODUCT DOCUMENTATION (Revised 10/11/2001)

The chart table snapshot utilizes JavaScript to deliver a powerful user interface to the charting software. The table consists of a customized chart for a symbol or group of symbols, a set of "tabs" at the top of the table that the user can click on to change the period charted (without opening a new page), and additional customization options such as chart type and moving averages. The main purpose of the chart table snapshot product is to provide a robust interface to the charting software for a given symbol.

This document will describe each of the customizable parameters included in the JavaScript code, the acceptable range for each parameter, and examples. The example JavaScript code below outputs a single stock table for the symbol RSLN. The code shown below contains brief comments; these comments will be expanded in the next section of this document.

```
<head>
<script language="javascript">
<!--

var CT_clientId      = "yourclientidhere"; // client ID
var CT_TableWidth    = 500;                // table width
var CT_ChartHeight   = 310;                // height of the chart
var CT_ChartLineThick = 1;                 // thickness in pixels of chart line
var CT_LeftBuffer    = 60;                 // number of pixels to allow on left for labels
var CT_RightBuffer   = 50;                 // number of pixels to allow on right for labels
var CT_BottomBuffer  = 30;                 // number of pixels to allow on bottom for labels
var CT_bottomTableHeight = 50;             // height for volume / legend table
var CT_TableColor    = "6699CC";          // color for table bkg
var CT_TableBkgColor = "FFFFFF";          // color behind table
var CT_ChartBkgColor = "6699CC";          // color for chart bkg
var CT_ChartGridBkg  = "6699CC";          // color for chart grid bkg
var CT_OffTabColor   = "666666";          // color for off tabs
var CT_ChartGridColor = "4A6F94";         // color for chart grid
var CT_ChartAxisColor = "385571";         // color for chart axis
var CT_FillMtnColor  = "04349C";          // fill color for mountain view
var CT_FillUpColor   = "049C04";          // fill color for price up
var CT_FillDnColor   = "9C0404";          // fill color vor price down
var CT_movingAvgColors = "FF8800,009900"; // colors for 50 and 200 day mov avgs
var CT_volumeColor   = "4A6F94";          // color for volume
var CT_ChartLineColor = "022061,AD0000"; // colors for chart lines
var CT_PrevClsColor  = "CC0000";          // color for previous close line
var CT_ShowPrevClose = "no";              // show previous closing price "yes"/"no"
var CT_LabelSide     = "left";            // show labels on "left" / "right" side of chart
var CT_LabelSymbols  = "no";              // show symbols at right of line end on chart
var CT_GridLocation  = "background";      // grid placed in "foreground"/"background"
var CT_showVolumeTable = "yes";           // show volume table "yes"/"no"
var CT_showLegendTable = "no";            // show legend table "yes"/"no"
var CT_smoothFonts   = "yes";             // smooth fonts on chart "yes"/"no"
var CT_smoothChart   = "yes";             // smooth lines on chart "yes"/"no"
var CT_StartType     = "mtn";             // start "mtn"/"prv"/"line"/"point"/"bar"/"candle"
var CT_StartPeriod   = "1y";              // start chart at - "1m"/"1q"/"1y"/"5y"
var CT_StartChartBy  = "";                // chart by price / % change to start - ""/"p"
```

```

var CT_StartMovAvg      = "no"; // show moving averages to start - "yes"/"no"
var CT_TabWidth        = 75;   // width of table tabs
var CT_TabHeight       = 14;   // height of table tabs
var CT_TabFontSize     = 2;    // font size
var CT_TabSmoothFonts  = "yes"; // smooth fonts on tabs "yes"/"no"
var CT_TabFontFace     = "Arial"; // font face
var CT_TabStdColor     = "FFFFFF"; // font color
var CT_TabLinkColor    = "EEEEEE"; // link color
var CT_TabRollColor    = "CCCCCC"; // link rollover color
var CT_upColor         = "067700"; // color to use for positive change value
var CT_dnColor         = "D60000"; // color to use for negative change value
var CT_chartLinks      = "yes"; // link clicks the chart to detailed quote
var CT_quoteBase       = "";    // base url for links
var CT_splitInfo       = "yes"; // show split history (when available)
var CT_chartByBox      = "yes"; // show price / % change choices
var CT_chartTypeBox    = "yes"; // show chart type choices
var CT_movAvgBox       = "yes"; // show moving averages choices
var CT_compareBox      = "yes"; // show "compare vs." form
var CT_showSymBox      = "yes"; // show "enter symbols" form

```

```

// -->
</script>

```

```

<script language="javascript"
src="http://www.tickertech.com/yourclientidhere/chart.mpl?ticker=RSLN&head=1">
</script>
</head>
<body>
<script language="javascript">
<!--
ShowHTTable();
// -->
</script>
</body>

```

This code was generated using the following URL: <http://demo.tickertech.com/charts3/>

This page contains a color-picking tool in the top left corner to help generate the hex codes corresponding to your custom color choices. Use this tool by clicking along the color bars, and then copying and pasting the hex code.

The bottom left corner of the page contains a form for customizing each parameter. At the bottom of the form is a text field for your client ID (the default setting will read "demo"). Change this field to the client ID you were assigned, and click the "generate chart table" button. Your customized chart table and the corresponding JavaScript code will now be displayed to the right side of the page.

## Customizable parameters

The remainder of this document will go through each of the customizable parameters shown in the JavaScript code above, and explain their function.

```
var CT_clientId      = "yourclientidhere";    // client ID
```

This parameter should contain the client ID you were assigned. This identifies your account, including the internet and intranet sites under your account currently activated to utilize the charting software.

```
var CT_TableWidth    = 500;                  // table width
```

This parameter controls the width of the table in pixels, and must be an integer value. Table width cannot be specified by %, but must be an integer value. Note that table width must be controlled by this parameter in conjunction with the tab width parameter (combined width of tabs may exceed specified table width value). Also note that the range of chart width is 20 pixels to 640 pixels. Should your size requirements fall outside of this range, contact your representative to discuss how we can meet your needs.

```
var CT_ChartHeight   = 310;                  // height of the chart
```

This parameter controls the height of the chart image in pixels, and must be an integer value. Note that the range of chart height is 20 pixels to 640 pixels. Should your size requirements fall outside of this range, contact your representative to discuss how we can meet your needs.

```
var CT_ChartLineThick = 1;                   // thickness in pixels of chart line
```

This parameter controls the thickness of the chart line in pixels. This parameter must be an integer value.

```
var CT_LeftBuffer    = 60;                   // number of pixels to allow on left labels
```

This integer value specifies the number of pixels to allow to the left of the grid for labels.

```
var CT_RightBuffer   = 50;                   // number of pixels to allow on right for labels
```

This integer value specifies the number of pixels to allow to the right of the grid for labels.

```
var CT_BottomBuffer  = 30;                   // number of pixels to allow on bottom for labels
```

This integer value specifies the number of pixels to allow to the bottom of the grid for labels.

```
var CT_bottomTableHeight = 50;              // height for volume / legend table
```

This integer value specifies the number of pixels to allow to for the height of a table below the grid, used for volume or symbol legend.

```
var CT_TableColor     = "6699CC";           // color for table bkg
```

This hex color value controls the main color of the table.

```
var CT_TableBkgColor  = "FFFFFF";           // color behind table
```

This hex color value controls the main background color for the table (this controls the background color used for the tabs and table edges).

```
var CT_ChartBkgColor = "6699CC"; // color for chart bkg
```

This hex color value controls the main background color of the chart image. This setting is usually set to correspond with the main table color, but may be set to any color value to meet your needs.

```
var CT_ChartGridBkg = "6699CC"; // color for chart grid bkg
```

This hex color value controls the background color of the grid area of the chart image.

```
var CT_OffTabColor = "666666"; // color for off tabs
```

This hex color value controls the color used for the tabs that are “off”. These are the tabs the user can click on to bring up a chart for a symbol other than the one currently shown. Note that the tab currently “on” will be shown in a color matching the main table.

```
var CT_ChartGridColor = "4A6F94"; // color for chart grid
```

This hex color value controls the color of the grid lines.

```
var CT_ChartAxisColor = "385571"; // color for chart axis
```

This hex color value controls the color of the axis lines.

```
var CT_FillMtnColor = "04349C"; // fill color for mountain view
```

This hex color value controls the color of the fill below the chart line under the mountain fill setting (as opposed to the fill to previous closing line setting).

```
var CT_FillUpColor = "049C04"; // fill color for price up
```

This hex color value controls the color of the fill areas above the closing price line under the fill to previous closing line setting.

```
var CT_FillDnColor = "9C0404"; // fill color for price down
```

This hex color value controls the color of the fill areas below the previous close line under the fill to previous closing line setting.

```
var CT_movingAvgColors = "FF8800,009900"; // colors for 50 and 200 day mov avgs
```

This pair of comma-delimited hex color values controls the color of the 50-day and 200-day moving average lines.

```
var CT_volumeColor = "4A6F94"; // color for volume
```

This hex color value controls the color used to draw the volume.

```
var CT_ChartLineColor = "022061,AD0000"; // colors for chart lines
```

This comma-delimited list of hex color values controls the line colors for each symbol being charted. For example, a value of “FF0000,0000FF” would set the first line to bright red and the second to bright blue.

```
var CT_PrevClsColor = "CC0000"; // color for previous close line
```

This hex color value controls the color of the previous closing price line (if activated using the parameter below).

```
var CT_ShowPrevClose = "yes"; // show previous closing price "yes"/"no"
```

This parameter controls whether or not the previous closing price line is shown. A setting of “yes” will show the previous close, while a setting of “no” will omit it.

```
var CT_LabelSide = "left"; // show labels on "left" / "right" side of chart
```

This parameter can be set to “left” or “right” to control which side of the chart the y-axis labels appear on.

```
var CT_LabelSymbols = "no"; // show symbols at right of line end on chart
```

This parameter controls the labeling of symbols to the right of the chart line end. Set this parameter to “yes” or “no” to turn symbol labeling on/off. Note that symbol labeling is done to the right side of the chart, and a setting of “yes” conflicts with the right-side label setting (CT\_LabelSide parameter above).

```
var CT_GridLocation = "background"; // grid placed in "foreground" / "background"
```

This parameter can be set to “background” or “foreground” and controls the position of the grid in relation to the chart lines and color fills.

```
var CT_showVolumeTable = "yes"; // show volume table "yes"/"no"
var CT_showLegendTable = "no"; // show legend table "yes"/"no"
```

These parameters control the usage of the bottom table. To turn on a volume table, the two parameters would be set to “yes” and “no” (as shown), while to turn on a legend table the two parameters would be set to “no” and “yes”. To turn off the bottom table entirely, specify “no” and “no”, and set the parameter for bottom table height to 0.

```
var CT_smoothFonts = "yes"; // smooth fonts on chart "yes"/"no"
```

This parameter controls whether or not fonts used in the chart for labels are smoothed. Smoothing uses color variations to give otherwise rough looking edges the appearance of being smooth. The effect can be more visually pleasing, but has the tradeoff of producing images with a greater number of colors. Images with more colors cannot be compressed as much as images with fewer colors. Thus, the file size of the chart image will be greater with this parameter set to “yes” than it will if this parameter is set to “no”. Charts with smoothing turned on for both fonts and chart line (next parameter) are approximately 40% larger in size (and consequently load more slowly). Acceptable settings for this parameter are “yes” and “no”. Use this parameter in combination with the CT\_smoothChart and CT\_TabSmoothFonts parameters below to help control the trade-off between a visually pleasing image and a larger file size.

```
var CT_smoothChart = "yes"; // smooth lines on chart "yes"/"no"
```

This parameter controls whether or not chart lines are smoothed, and can be set to “yes” or “no”.

```
var CT_StartType = "mtn"; // start "mtn"/"prv"/"line"/"point"/"bar"/"candle"
```

This parameter controls the chart type to start with, and can be set to the following values:

- “mtn”: Line graph with mountain fill (fill below line)
- “prv”: Line graph with fill to previous closing line
- “line”: Line graph with no fill
- “point”: Point graph
- “bar”: Bar graph
- “candle”: Candle stick graph

```
var CT_StartPeriod      = "1y"; // start chart at - "1m"/"1q"/"1y"/"5y"
```

This parameter controls the period of the chart to start with, and can be set to the following values:

- “1m” : 1 month chart
- “1q” : 1 quarter chart
- “1y” : 1 year chart
- “5y” : 5 year chart

Should your particular needs require additional periods, contact your representative to discuss custom options.

```
var CT_StartChartBy     = ""; // chart by price / % change to start - ""/"p"
```

This parameter controls whether the chart is shown by price or by % change from starting value. A setting of "" (empty string) results in a chart by price, while a setting of “p” results in a chart by % change.

```
var CT_StartMovAvg      = "no"; // show moving averages to start - "yes"/"no"
```

This parameter controls whether or not moving averages are shown, and can be set to “yes” or “no” to turn moving averages on/off.

```
var CT_TabWidth         = 75; // width of table tabs
```

This integer value specifies the number of pixels for the width of each “tab” shown at the top of the table. The currently allowed range of values is 15 to 300 pixels. Width may not be specified by %. Should your size requirements fall outside of this range, contact your representative to discuss how we can meet your needs.

```
var CT_TabHeight        = 14; // height of table tabs
```

This integer value specifies the number of pixels for the height of each “tab” shown at the top of the table. The currently allowed range of values is 5 to 100 pixels. Height may not be specified by %. Should your size requirements fall outside of this range, contact your representative to discuss how we can meet your needs.

```
var CT_TabFontSize     = 2; // font size
```

This parameter controls the font size. The font point size will be 9pt +/- the indicated integer. Acceptable font range is 4pt to 100pt.

```
var CT_TabSmoothFonts  = "yes"; // smooth fonts on tabs "yes"/"no"
```

This parameter controls whether or not smoothing is done on the text in the tabs.

```
var CT_TabFontFace     = "Arial"; // font face
```

This parameter controls the font face used for the tabs, table text, and labels in the chart. Dozens of fonts are currently supported, and other custom fonts can be installed for a fee. The list of currently supported fonts can be found at <http://www.tickertech.com/fonts/>. Please contact your representative for details if you are interested in a font not listed.

Please Note: When not using <http://demo.tickertech.com/charts3/> to perform customizations, take special care when specifying a font name that has multiple words (such as “calisto mt”). Be sure to replace every space (‘ ’) with a ‘+’ character.

```
var CT_TabStdColor     = "FFFFFF"; // font color
```

This hex color value controls the color used to draw the chart labels, tabs, and table text.

```
var CT_TabLinkColor    = "EEEEEE"; // link color
```

This hex color value controls the color used to display hyperlinks, in both the tabs and links within the table.

```
var CT_TabRollColor    = "CCCCCC"; // link rollover color
```

This hex color value controls the color used for rollovers on hyperlink items (i.e. when the user passes their mouse over a linkable item, the color of that item will change to the color specified here). Note that the rollover color in the table text is controlled by a style sheet, and may not be supported in Netscape browsers.

```
var CT_upColor         = "067700"; // color to use for positive change value
```

This hex color value controls the color used to draw a positive price change in the table text. This also controls the color of the up arrow.

```
var CT_dnColor        = "D60000"; // color to use for negative change value
```

This hex color value controls the color used to draw a negative price change in the table text. This also controls the color of the down arrow.

```
var CT_chartLinks     = "yes";      // link clicks the chart to detailed quote
```

This parameter controls whether or not links are enabled, by setting to “yes” or “no”. If you have not supplemented your stock table with detailed quotes or other packages, you can set this parameter to “no” to disable links.

```
var CT_quoteBase      = "";        // base url for links
```

This parameter can be set to a base url for links (if links are enabled). Leave this parameter set to “” unless you have consulted with your representative.

```
var CT_splitInfo      = "yes";     // show split history (when available)
```

This parameter controls whether or not split history is shown below the chart image (if it exists for the symbol in question), and can be set to “yes” or “no”.

```
var CT_chartByBox     = "yes";     // show price / % change choices
```

This parameter controls whether the option to change the chart to by-%-change or by-price is displayed, and can be set to “yes” or “no”.

```
var CT_chartTypeBox   = "yes";     // show chart type choices
```

This parameter controls whether the option to change the chart to different views (mountain, line, point, etc) is displayed, and can be set to “yes” or “no”.

```
var CT_movAvgBox      = "yes";     // show moving averages choices
```

This parameter controls whether the option to turn moving averages on and off is displayed, and can be set to “yes” or “no”.

```
var CT_compareBox     = "yes";     // show "compare vs." form
```

This parameter controls whether the option to compare multiple stocks and indices is displayed, and can be set to “yes” or “no”.

```
var CT_showSymBox      = "yes";          // show "enter symbols" form
```

This parameter controls whether an “enter symbols” form is displayed, and can be set to “yes” or “no”. This form can be used to bring up charts for additional symbols, and can also act as a portal to basic & detailed quotes, news, company profiles, and other packages which you may have purchased to supplement your chart table. Please contact your representative for details on other available packages.

```
<script language="javascript"
src="http://www.tickertech.com/yourclientidhere/chart.mpl?ticker=RSLN&head=1">
</script>
```

This portion of JavaScript code calls an application on the tickertech.com servers, which generates a function to output the HTML code for the stock table based on your customization settings. The customization parameters and this code should be placed within the `<head>` and `</head>` tags. Note that your client ID must replace the text “yourclientidhere”. Also note that the ticker symbol “RSLN” must be replaced with the ticker symbol(s) of your choice.

```
<body>
<script language="javascript">
<!--
ShowHTTable();
// -->
</script>
</body>
```

This portion of JavaScript code calls the function to output the HTML code into the page. Place this portion of code within the `<body>` and `</body>` tags where you want the content to appear.

## Examples

Your account includes two example pages that you can access from the tickertech.com web site. In the URLs below, replace “**yourclientidhere**” with the client ID you were issued:

[http://www.tickertech.com/\*\*yourclientidhere\*\*/chart1.html](http://www.tickertech.com/<b>yourclientidhere</b>/chart1.html)

[http://www.tickertech.com/\*\*yourclientidhere\*\*/chart2.html](http://www.tickertech.com/<b>yourclientidhere</b>/chart2.html)