Conditional Formatting Rule Types

By Shayanna Gatchalian, Technical Services Engineer, 4D Inc.

Technical Note 23-10

Table of Contents

The Conditional Formatting Rule Types

This document serves as a condensed version of the SpreadJS documentation on conditional formatting, which provides the information needed to correctly create conditional formatting rules within the context of 4D. Each rule will be displayed as a table, detailing the properties that can be configured to create the rule object. A link to each rule's "add" method in the "CondtionalFormats" class will also be provided to show where these properties came from.

It should be noted that *all* rule objects must have the "ruleType" and "ranges" properties; any other properties used will depend on the chosen rule type. Moreover, when comparing the property names on the SpreadJS documentation website with the property names in the 4D View Pro object, there may be inconsistencies; the property names in this document are the ones used in the context of 4D.

If you would like to read the SpreadJS documentation yourself, you may find the following documentation links useful:

Rule Type Enumeration Values:
https://www.grapecity.com/spreadjs/api/enums/GC.Spread.Sheets.ConditionalFormatting.RuleType

Conditional Formats Class:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#class-conditionalformats

Average Rule

The average rule applies cell styling according to either the average value or standard deviation of the cell range.

Average Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addaveragerule

Average Rule

Property	**Value Type**	**Value**	**Description**
ruleType*	Real	8	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
type*	Real	AverageConditionType enumeration value • Above = 0 • above1StdDev = 4 • above2StdDev = 6 • above3StdDev = 8 • Below = 1 • below1StdDev = 5 • below2StdDev = 7 • below3StdDev = 9 • equalOrAbove = 2 • equalOrBelow = 3	Defines the condition in which to apply the conditional format to

*Required properties

Cell Value Rule

The cell value rule compares the cell's value against either a single specified value or two specified values. If the condition is true according to the "operator" property, the cell styling will be applied.

Cell Value Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addcellvaluerule

Cell Value Rule

Property	**Value Type**	**Value**	**Description**
ruleType*	Real	1	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
operator*	Real	ComparisonOperators enumeration value • between = 6 • equalsTo = 0 • greaterThan = 2 • greaterThanOrEqualsTo = 3 • lessThan = 4 • lessThanOrEqualsTo = 5 • notBetween = 7 • notEqualsTo = 1	Defines which comparison operator to use
value1*	Real or Text	Number or text value	Defines the value you would like to compare the cell's value to
value2	Real or Text	Number or text value	Defines the second value you would like to compare the cell's value to (only needed for "between" and "notBetween" operators)

*Required properties

Column State Rule

The column state rule formats the entire column depending on its state. This rule type (along with the row state rule) is unique in that the “states” property under the active spreadsheet must also be configured.

Column State Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addcolumnstaterule

Column State Rule

Property	Value Type	Value	Description
ruleType*	Real	15	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
state*	Real	RowColumnStates enumeration value • active = 16 • dirty = 64 • edit = 8 • hover = 1 • inserted = 128 • invalid = 2 • invalidFormula = 512 • selected = 32	Defines the state condition in which to apply cell styling to

*Required properties

For each unique state type used (under the “state” property of the column rule object), a state object is inserted into a “states” collection, which is then attached to the “states” property of the current sheet. **When a state applies to both columns and rows, the “column” and “row” properties are both set to true.**

State Object

Property	Value Type	Value	Description
state*	Real	RowColumnStates enumeration value • active = 16 • dirty = 64 • edit = 8 • hover = 1 • inserted = 128 • invalid = 2 • invalidFormula = 512 • selected = 32	Defines the state condition for either the column or row **(must be the same value defined in the “state” property of the column state rule)**
column*	Boolean	True/False	True = state condition is applied to column False = state condition is not applied to column
row*	Boolean	True/False	True = state condition is applied to row False = state condition is not applied to row

Data Bar Rule

The data bar rule draws a bar within in the cell according to its value, in comparison with the other cell values of the rule's cell range.

Data Bar Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#adddatabarrule

Data Bar Rule

Property	Value Type	Value	Description
ruleType*	Real	12	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
color*	Text	Color constant, HEX value, RGB object	Sets the fill color of the positive data bar
maxType*	Real	ScaleValueType enumeration value • automin = 5 • formula = 6 • highestValue = 2 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling (for max value)
maxValue*	Real	Max scale value	Defines the max scale value of the cell range
minType*	Real	ScaleValueType enumeration value • automax = 7 • formula = 6 • lowestValue = 1 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling (for minimum value)
minValue*	Real	Minimum scale value	Defines the minimum scale value of the cell range
axisColor	Text	Color constant, HEX value, RGB object	Sets the color of the axis
axisPosition	Real	DataBarAxisPosition enumeration value • automatic = 0 • cellMidPoint = 1 • none = 2	Sets the position of the axis

borderColor	Text	Color constant, HEX value, RGB object	Sets the border color of the positive data bar
dataBarDirection	Real	BarDirection enumeration value - leftToRight = 0 - rightToLeft = 1	Sets the direction of the data bar within the cell

gradient	Boolean	True/false	True = gradient
			False = no gradient

negativeBorderColor	Text	Color constant, HEX value, RGB object	Sets the border color of the negative data bar
negativeFillColor	Text	Color constant, HEX value, RGB object	Sets the fill color of the negative data bar
showBarOnly	Boolean	True/false	True = does not display number values False = shows number value
showBorder	Boolean	True/false	True = shows border False = does not show border
useNegativeBorderColor	Boolean	True/false	True = will use fill color defined by “negativeBorderColor” property False = will use fill color defined by “borderColor” property
userNegativeFillColor	Boolean	True/false	True = will use fill color defined by “negativeFillColor” property False = will use fill color defined by “color” property

*Required properties

Date Occurring Rule

The date occurring rule applies cell styling to the cell range if the date falls within the condition’s date range in relation to the current date.

Date Occurring Rule Documentation: https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#adddateoccurringrule

Date Occurring Rule

Property	Value Type	Value	Description
ruleType*	Real	4	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
type*	Real	DateOccuringType enumeration value • last7Days = 3 • lastMonth = 5 • **lastQuarter = 12** • lastWeek = 8 • **lastYear = 15** • nextMonth = 6 • **nextQuarter = 10** • nextWeek = 9 • **nextYear = 13** • thisMonth = 4 • **thisQuarter = 11** • thisWeek = 7 • **thisYear = 14** • today = 0 • tomorrow = 2 • yesterday = 1	Defines the date range in which to apply the cell styling to **(bolded values are not available through the View Pro UI but can be applied programmatically)**

*Required properties

Duplicate Rule

The duplicate rule applies cell styling to any duplicate cells within the cell range.

Duplicate Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addduplicaterule

Duplicate Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	7	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details

*Required properties

Formula Rule

The formula rule applies cell styling if the formula defined in the “formula” parameter returns true.

Formula Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addformularule

Formula Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	3	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
formula*	Text	The formula	Defines the formula that determines if the cell range will be styled when true

*Required properties

Icon Set Rule

The icon set rule applies a specific icon depending on the cell value in comparison to the rest of the cell range. Icon sets are limited to what is provided by SpreadJS.

Icon Set Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addiconsetrule

There are two ways to implement the icon set rule. You can either simply use one icon set on a cell range with the “iconSetType” parameter or use a custom icon set with individual icons from different sets. The two tables below will help you set the properties depending on which method you choose.

Easy Method – One Icon Set

Property	Value Type	Value	Description
ruleType*	Real	13	Defines the enumeration value to use for this rule
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
iconSetType	Real	iconSetType enumeration value • threeArrowsColored = 0 • threeArrowsGray = 1 • threeTriangles = 2 • threeStars = 3 • threeFlags = 4 • threeTrafficLightsUnrimmed = 5 • threeTrafficLightsRimmed = 6 • threeSigns = 7 • threeSymbolsCircled = 8 • threeSymbolsUncircled = 9 • fourArrowsColored = 10 • fourArrowsGray = 11 • fourRedToBlack = 12 • fourRatings = 13 • fourTrafficLights = 14 • fivearrowsColored = 15 • fiveArrowsGray = 16 • fiveRatings = 17 • fiveQuarters = 18 • fiveBoxes =19	
reverseIconOrder	Boolean	True/false	True = uses reverse order of icon set False = uses default order of icon set
showIconOnly	Boolean	True/false	True = only shows the icon False = will show both the icon and cell value

*Required properties

Advanced Method – Custom Icon Set

Property	Value Type	Value	Description
ruleType*	Real	13	Defines the enumeration value to use for this rule
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
iconCriteria*	Collection	Collection of iconCriterion objects	Refer to the iconCriterion object table for object properties
icons*	Collection	Collection of icon objects	Refer to the icon object table for object properties
reverseIconOrder	Boolean	True/false	True = uses reverse order of icon set False = uses default order of icon set
showIconOnly	Boolean	True/false	True = only shows the icon False = will show both the icon and cell value

*Required properties

IconCriterion Object

Property	Value Type	Value	Description
iconValue*	Real	One of the cell values in the cell range	Refers to the value of the cell that will serve as the end of the range for the particular icon used
iconValueType*	Real	iconValueType enumeration value • formula = 7 • number = 1 • percent = 4 • percentile = 5	Defines the scaling used when applying the icon set
isGreaterThanEqualTo*	Boolean	True/false	True = will apply icon if **greater than/equal to value** False = will apply icon if **strictly greater than value**

*Required properties

Icon Object

Property	Value Type	Value	Description
iconIndex*	Real	Refer to “Icon Index Reference” table	The index of the icon you would like to use from the

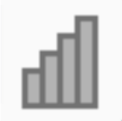
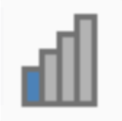
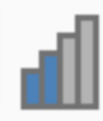
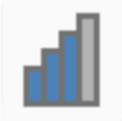
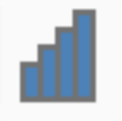
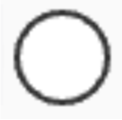
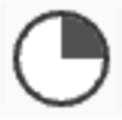
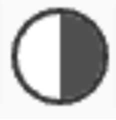
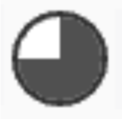
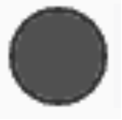
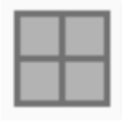
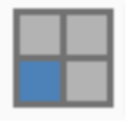
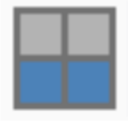
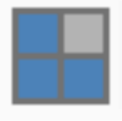
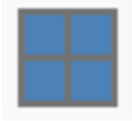
			“iconSetType” set (refer to Icon Index table)
iconSetType*	Real	iconSetType enumeration value • threeArrowsColored = 0 • threeArrowsGray = 1 • threeTriangles = 2 • threeStars = 3 • threeFlags = 4 • threeTrafficLightsUnrimmed = 5 • threeTrafficLightsRimmed = 6 • threeSigns = 7 • threeSymbolsCircled = 8 • threeSymbolsUncircled = 9 • fourArrowsColored = 10 • fourArrowsGray = 11 • fourRedToBlack = 12 • fourRatings = 13 • fourTrafficLights = 14 • fivearrowsColored = 15 • fiveArrowsGray = 16 • fiveRatings = 17 • fiveQuarters = 18 • fiveBoxes =19	Defines which icon set to be used on the cell range

*Required properties

Icon Index Reference

Icon Set Name	Enumeration Value	Icon Index		
threeArrowsColored	0	⬇	➡	⬆
		0	1	2
threeArrowsGray	1	⬇	➡	⬆
		0	1	2
threeTriangles	2	▼	▬	▲
		0	1	2
threeStars	3	☆	⯪	★
		0	1	2
threeFlags	4	⚑	⚑	⚑
		0	1	2
threeTrafficLightsUnrimmed	5	🔴	🟡	🟢
		0	1	2
threeTrafficLightsRimmed	6	🔴	🟡	🟢
		0	1	2
threeSigns	7	◆	▲	●
		0	1	2

threeSymbolsCircled	8	⊗	❗	✅		
		0	1	2		
fourSymbolsUncircled	9	✖	!	✓		
		0	1	2		
fourArrowsColored	10	⬇	⬈	⬊	⬆	
		0	1	2	3	
fourArrowsGray	11	⬇	⬈	⬊	⬆	
		0	1	2	3	
fourRedToBlack	12	⚫	⚪	●	🔴	
		0	1	2	3	
fourRatings	13	📶	📶	📶	📶	
		0	1	2	3	
fourTrafficLights	14	⚫	🟢	🟡	🔴	
		0	1	2	3	4
fiveArrowsColored	15	⬇	⬊	➡	⬈	⬆
		0	1	2	3	4
fiveArrowsGray	16	⬇	⬊	➡	⬈	⬆
		0	1	2	3	4

fiveRatings	17					
		0	1	2	3	4
fiveQuarters	18					
		0	1	2	3	4
fiveBoxes	19					
		0	1	2	3	4

Row State Rule

The row state rule formats the entire row depending on its state. This rule type (along with the column state rule) is unique in that the "states" property under the active spreadsheet must also be configured.

Row State Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addrowstaterule

Row State Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	14	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
state*	Real	RowColumnStates enumeration value • active = 16 • dirty = 64 • edit = 8 • hover = 1 • inserted = 128 • invalid = 2 • invalidFormula = 512 • selected = 32	Defines the state condition in which to apply cell styling to

*Required properties

For each unique state type used (under the "state" property of the column rule object), a state object is inserted into a "states" collection, which is then attached to the "states" property of the current sheet. **When a state applies to both columns and rows, the "column" and "row" properties are both set to true.**

State Object

Property	Value Type	Value	Description
state*	Real	RowColumnStates enumeration value • active = 16 • dirty = 64 • edit = 8 • hover = 1 • inserted = 128	Defines the state condition for either the column or row **(must be the same value defined in the "state" property of the row state rule)**

		• invalid = 2 • invalidFormula = 512 • selected = 32	
column*	Boolean	True/False	True = state condition is applied to column False = state condition is not applied to column
row*	Boolean	True/False	True = state condition is applied to row False = state condition is not applied to row

Specific Text Rule

The specific text rule compares the cell's value against a specified text value. If the condition is true according to the "operator" property, the cell styling will be applied.

Specific Text Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addspecifictextrule

Specific Text Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	2	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
operator*	Real	TextComparisonOperators enumeration value • beginsWith = 2 • contains = 0 • doesNotContain = 1 • endsWith = 3	Defines which operator will be used when comparing the cell value to the "text" property
text*	Text	Text value	The string value that will be compared against the cell value

*Required properties

Three Scale Rule

The three-scale rule changes the background color of each cell in a range based on a 3-color gradient scale.

Three Scale Rule Documentation: https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#add3scalerule

Three Scale Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	11	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
minType*	Real	ScaleValueType enumeration value • formula = 6 • lowestValue = 1 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling used at the minimum end of the range
minValue*	Real	Minimum scale value	Specifies the minimum value for the cell range **(not used with "lowest value" minType)**
minColor*	Text	Color constant, HEX value, RGB object	Sets the background color for minimum value of the range
midType*	Real	ScaleValueType enumeration value • formula = 6 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling used at the middle of the range
midValue*	Real	Mid-scale value	Specifies the middle value for the cell range
midColor*	Text	Color constant, HEX value, RGB object	Sets the background color for middle value of the range
maxType*	Real	ScaleValueType enumeration value • formula = 6 • highestValue = 1 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling used at the maximum end of the range

maxValue*	Real	Maximum scale value	Specifies the max value for the cell range **(not used with “highest value” maxType)**
maxColor*	Text	Color constant, HEX value, RGB object	Sets the background color for maximum value of the range

*Required properties

Top 10 Rule

The top 10 rule applies cell styling to the top or bottom N values, according to the “type” and “rank” properties.

Top 10 Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#addtop10rule

Top 10 Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	5	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
type*	Real	Top10ConditionType enumeration value • bottom = 1 • top = 0	Specifies whether the rule should apply to either top or bottom-ranking cells
rank*	Real	Number of cells	Specifies the number of top/bottom-ranking cells to apply cell styling to

*Required properties

Two Scale Rule

The two scale rule changes the background color of each cell in a range based on a 3-color gradient scale.

Two Scale Rule Documentation: https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#add2scalerule

Three Scale Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	10	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details
minType*	Real	ScaleValueType enumeration value • formula = 6 • lowestValue = 1 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling used at the minimum end of the range
minValue*	Real	Minimum scale value	Specifies the minimum value for the cell range **(not used with "lowest value" minType)**
minColor*	Text	Color constant, HEX value, RGB object	Sets the background color for minimum value of the range
maxType*	Real	ScaleValueType enumeration value • formula = 6 • highestValue = 1 • number = 0 • percent = 3 • percentile = 4	Defines the type of scaling used at the maximum end of the range
maxValue*	Real	Maximum scale value	Specifies the max value for the cell range **(not used with "highest value" maxType)**
maxColor*	Text	Color constant, HEX value, RGB object	Sets the background color for maximum value of the range

*Required properties

Unique Rule

The unique rule only applies cell styling to cells with unique values in the cell range.

Unique Rule Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.ConditionalFormatting.ConditionalFormats#adduniquerule

Unique Rule Properties

Property	Value Type	Value	Description
ruleType*	Real	6	Defines the enumeration value to use for this rule
style*	Object	The style object	Refer to the "Style Object" section for more details
ranges*	Object	The ranges object	Refer to the "Ranges Object" section for more details

*Required properties

Style Object

While the style object is not a required property for the rule object in general, most rule types require it; since style objects are used more often than not, this section is dedicated explaining the style object structure.

View Pro already has built-in commands to get and set the cell style of a range, but only the style object is needed when implementing conditional formatting. To reiterate once more, the View Pro object is based off the SpreadJS structure; so, although the View Pro documentation includes an extensive list of the style object's properties and even subproperties, you can access more features when using the SpreadJS documentation as a reference (e.g., "showEllipsis" and "mask" properties).

View Pro Style Object Documentation:
https://developer.4d.com/docs/ViewPro/configuring#style-objects

SpreadJS Style Class Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.Style#class-style

Ranges Object

The ranges collection specifies which cells to apply the conditional formatting rule to. This collection is comprised of range objects, where each one specifies a “rectangle” of contiguous cells, enabling the entire collection to span over different groups of cells. Each range object is comprised of the following properties.

Range Object Documentation:
https://www.grapecity.com/spreadjs/api/classes/GC.Spread.Sheets.Range

Range Object Properties

Property	Value Type	Value	Description
col*	Real	A positive whole number	Column index of the top-left cell in the contiguous range
colCount*	Real	A positive whole number	Number of columns (goes from left to right from starting column index)
row*	Real	A positive whole number	Row index of the top-left cell in the contiguous range
rowCount*	Real	A positive whole number	Number of rows (goes from top to bottom from starting row index)

*Required properties