

# 15 Schematic Status Display

## Overview

The Schematic Status Display (SSD, formerly named Scada display) in Delft-FEWS is used for displaying and monitoring data. The schematic status display shows one or more configurable schematic views that represent data values in some way. For example, to show the most recent data value of a given time series, it is possible to show just the numerical value, or to show a rectangle that varies in height depending on the data value, or to show a polygon that changes in color when the data value crosses a certain threshold, etc. How data is represented and which data is shown can be configured in the schematic status display configuration file. The schematic status display is dynamically updated whenever new data becomes available in the system or whenever data is changed (e.g. in the time series editor). The SSD is comparable to the main map display, only the SSD does not and cannot refer to any geographical coordinate system. Furthermore the SSD can be used to show text and figures as well as objects that represent data values.



Please note that the SSD in Delft-FEWS is only used for displaying data, it does not implement all features that could be expected from a SCADA (Supervisory Control And Data Acquisition) system.



Before 2011\_02, multipliers were sometimes used to do unit conversion. As of 2011\_02 the standard unit conversion from Delft-FEWS will be applied to the data shown in the Schematic Status Display. In case of migration of systems from before 2011\_02, it can be necessary to verify whether the unit conversion is not applied twice.



Note when using of transformations in the Schematic Status Display, it is only supported to have outputVariables of the transformation output of timeSeriesType *temporary*. Though other transformation functions may function, it is only the UserSimple function that is supported.

## Contents

- [Overview](#)
- [Contents](#)
- [Explorer reference to the Schematic Status Display](#)
- [Generic Schematic Status Display Configuration Options](#)
  - [displayName](#)
  - [showTimeNavigatorToolbar](#)
  - [Time Navigator Toolbar Configuration Options](#)
    - [timeNavigatorTimeStep](#)
    - [timeNavigatorRelativePeriod](#)
    - [timeNavigatorRelativePeriodVariables](#)
    - [timeNavigatorTimeStep](#)
    - [movieFrameDurationMillis](#)
    - [activate](#)
    - [showTimeNavigatorActivationToggle](#)
  - [dateFormat](#)
  - [numberFormat](#)
- [variable](#)
  - [Variable Configuration Options](#)
    - [variableId](#)
    - [locationId](#)
    - [overrulingRelativeViewPeriod](#)
    - [timeSeriesSet](#)
- [transformation](#)
  - [Transformations within ScadaDisplay](#)
    - [Sample configuration for transformations within the ScadaDisplay](#)
- [scadaPanel](#)
  - [id](#)
  - [name](#)
  - [svgFile](#)
  - [overrulingTimeNavigatorTimeStep](#)
  - [nodeld](#)
  - [backgroundColor](#)
  - [permission](#)
  - [textComponentBehaviourDefinition](#)
  - [shapeComponentBehaviourDefinition](#)
  - [chartComponentBehaviourDefinition](#)
    - [display](#)
    - [axisLabels](#)
    - [axisTitle](#)
    - [imageScaleFactor](#)
  - [Component behaviour configuration options](#)
    - [svgObjectId](#)
    - [leftSingleClickAction](#)
    - [leftDoubleClickAction](#)
    - [toolTip](#)

- linkPropertiesToData
  - useThresholdWarningLevelColors
  - useThresholdWarningLevelSymbols
  - replaceTags
- Left Single Click Action and Left Double Click Action Configuration Options
  - switchToScadaPanel
  - scadaPanelId
  - openDisplay
  - spatialDisplay
  - timeSeriesDisplay
  - timeSeriesEditor
  - taskRunDialog
  - thresholdEventsDisplay
  - title
  - variable
  - showGraph
  - showTable
  - runWorkflow
  - OpenPdfFile
  - OpenURL
  - enumerateValues
  - enterValue
  - Link Properties To Data Configuration Options
- Link Height To Data Configuration Options
  - variable
  - dataLowerLimit
  - dataUpperLimit
  - heightLowerLimit
  - heightUpperLimit
  - anchorPoint
- Link Width To Data Configuration Options
  - variable
  - dataLowerLimit
  - dataUpperLimit
  - widthLowerLimit
  - widthUpperLimit
  - anchorPoint
- Link Rotation To Data Configuration Options
  - variable
  - dataLowerLimit
  - dataUpperLimit
  - rotationLowerLimit
  - rotationUpperLimit
  - anchorPointX
  - anchorPointY
- Use Threshold Warning Level Colors Configuration Options
  - variable
  - thresholdGroupId
  - thresholdReference
  - colorType
- Use Threshold Warning Level Symbols Configuration Options
- Tooltip Configuration Options
  - variable
  - toolTipText
- Replace Tags Configuration Options
- Navigation using Topology
  - mainPanel
  - scadaDisplayId
  - scadaPanelId
- Known Issues
- Tips And Tricks
  - SVG specification
  - Embedding image files into SVG files
  - Controlling the resizing behaviour of an svg document within the scada display
  - Determining the rotation anchor point for an SVG object in user space coordinates
  - Aligning text within svg text objects
  - Export maps from ArcGis as svg files
  - Reduce the size of svg files

## Explorer reference to the Schematic Status Display

The schematic status display shows one or more status panels, which can be selected in turn from the list on the left hand side. It is also possible to have multiple schematic status displays, each one with different panels. In that case there would be one configuration file for each different schematic status display, each one with a different filename. To be able to open a schematic status display from the user interface, there should be an explorer task for it in the Explorer.xml configuration file. The xml code for a schematic status display explorer task is for example:

```

<explorerTask name="Twentekanalen">
  <displayConfigFileName>StatusTwentekanalen</displayConfigFileName>
  <toolbarTask>true</toolbarTask>
  <menubarTask>true</menubarTask>
</explorerTask>

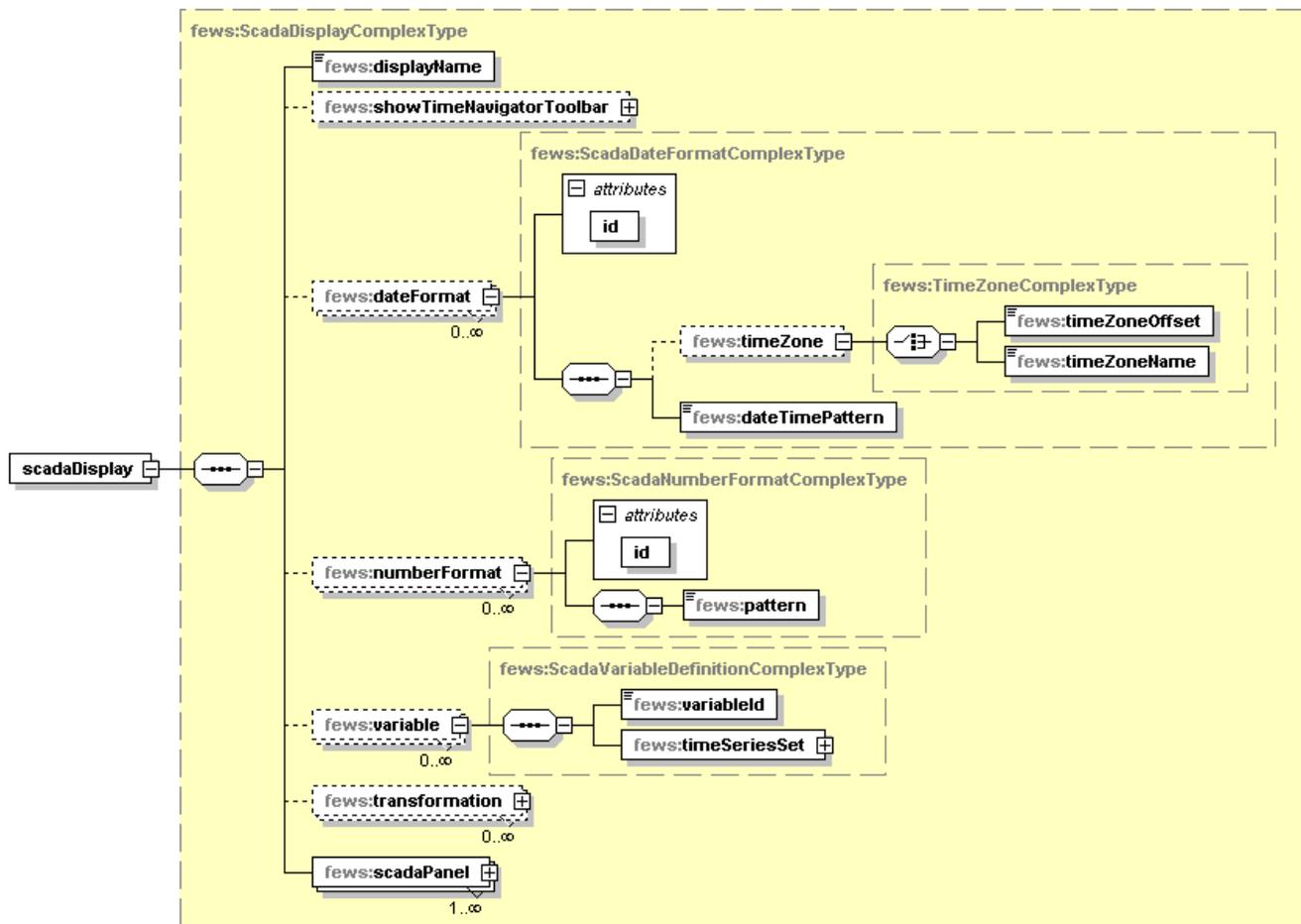
```

Example Configuration files:

<a href="#">ScadaTwentekanalen.xml</a>	Example of a scada display configuration file
<a href="#">Twentekanalen_10min.svg</a>	Example of an svg file, which is used in the ScadaTwentekanalen.xml example configuration file

## Generic Schematic Status Display Configuration Options

Below is an overview of the options that are available in the schematic status display xml schema. All configuration options are also documented in the annotations in the schematic status display xml schema. To get the most up to date information about the available configuration options and their documentation in the annotations, please consult the schematic status display xml schema, which is available [here](#).



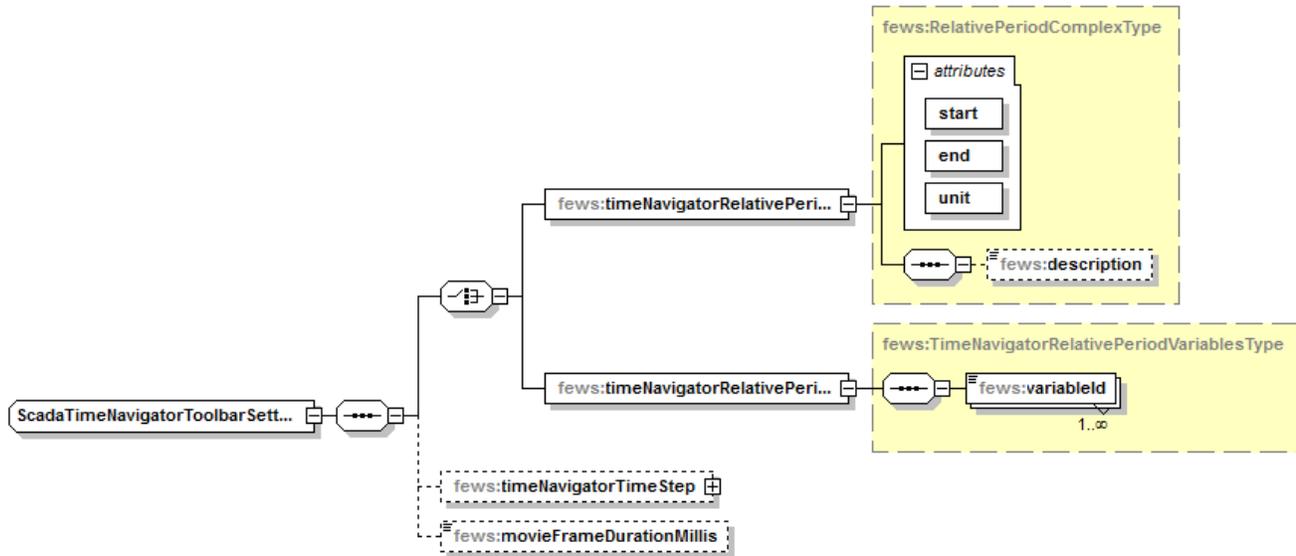
### displayName

Title of this display.

### showTimeNavigatorToolBar

Option to show a time navigator toolbar at the top of this schematic status display. The time navigator toolbar can be used to select the display time for this schematic status display. It is only possible to select a display time that is contained within the configured relative period and is a valid time according to the configured time step. This period is always relative to the current system time. If the current system time changes, then the display time is reset to the current system time. If this option is not specified, then the time navigator toolbar is not shown.

### Time Navigator Toolbar Configuration Options



Generated by XMLSpy

www.altova.com

Time navigator toolbar configuration elements

### timeNavigatorTimeStep

The default timestep by which the time navigator slider moves is by default the cardinal timestep which is configured in the explorer.xml configuration file, see [FEWS Explorer Configuration](#). This optional element can be used to use a different timestep for the time navigator than the cardinal timestep.

### timeNavigatorRelativePeriod

This is the period of the time navigator toolbar (slider) in this schematic status display. The time navigator toolbar can be used to select the display time for this schematic status display. It is only possible to select a display time that is contained within this period and is a valid time according to the cardinal time step (which is configured in the explorer.xml configuration file, see [FEWS Explorer Configuration](#)). This period is always relative to the current system time. If the current system time changes, then the display time is reset to the current system time. The start and end of the period are both included.

### timeNavigatorRelativePeriodVariables

Instead of the timeNavigatorRelativePeriod element, it is also possible to configure the relative view period using the timeNavigatorRelativePeriodVariables element. Using this element a list of variableId's have to be specified that refer to variables defined in the scada display configuration. For each specified variable the relative view period of the timeserieset is determined and will be joined together into a relative view period that includes all previous periods. The resulting calculated period will be used for the time navigator toolbar. To be able to determine the relative view period, the timeSeriesSet the variables reference should be set to: `<readWriteMode>read complete forecast</readWriteMode>`.

This period is always relative to the current system time, similar to the timeNavigatorRelativePeriod. A typical configuration might look as follows:

```
<timeNavigatorRelativePeriodVariables>
  <variableId>variableId1</variableId>
  <variableId>variableId2</variableId>
</timeNavigatorRelativePeriodVariables>
```

Suppose *variableId1* has the following relative view period: `<relativeViewPeriod start="-1" end="6" unit="day"/>`

And *variableId2* has the following relative view period: `<relativeViewPeriod start="-2" end="5" unit="day"/>`

The resulting relative view period used by the time navigator toolbar will be: `<relativeViewPeriod start="-2" end="6" unit="day"/>`

### timeNavigatorTimeStep

The timestep by which the time navigator slider is moved, by default the cardinal timestep which is configured in the explorer.xml configuration file. This optional element can be used to use a different timestep for the time navigator than the cardinal timestep.

### movieFrameDurationMillis

The duration of a frame when the time navigator is animating. This is the number of milliseconds a frame/time step is visible before the next time step becomes visible. If this option is not specified, then 200 milliseconds is used by default. When the CPU is too slow to display the specified frame rate, a frame will be displayed longer than specified.

## activate

The activate option is default true and can be set to false to show the time navigator in an inactive state. In this state the time navigator listens to the system time as if there was no time navigator configured at all.

## showTimeNavigatorActivationToggle

Using the showTimeNavigatorActivationToggle option (default false) will allow you to switch the time navigator between active and inactive with a toggle button on the navigator toolbar.

## backgroundColor

Optional configuration for the background color of the scadaPanels. This options also works with the defaultCustomColorKeys that can be used in combination with color schemes that will allow you to select a different color for each color schem. To use a custom color, the bracket notations is required to refer to a custom color. For example:

```
<backgroundColor>[MY_blue]</backgroundColor>
```

## dateFormat

Definitions of dateFormats that can be used for formatting dates and times in tags in texts of svg objects.

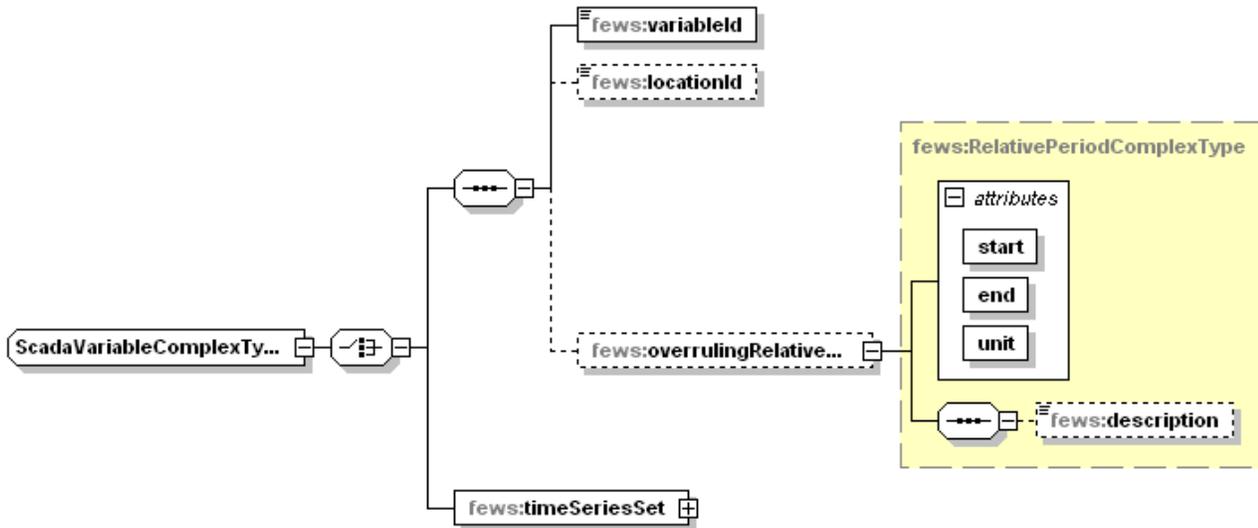
## numberFormat

Definitions of numberFormats that can be used for formatting numbers in tags in texts of svg objects.

## variable

Definitions of variables that can be used as input and/or output for the components in the schematic status display. A variable is always a time series. Alternatively variable definitions can be embedded in the configuration below.

### Variable Configuration Options



Generated by XMLSpy

www.altova.com

Variable configuration elements

Choose between a reference to a variable or an embedded definition of a variable.

### variableId

Identifier of a variable to use.

### locationId

If the specified variable contains multiple locations, then specify the location to use here.

## overrulingRelativeViewPeriod

Optional time period for which data should be read. This time period overrules the viewPeriod in the timeSeriesSet of the referenced variable. This time period is relative to the selected display time in this scada display. The start and end of the period are both included. If the start and/or end of the period is not a valid time according to the timeStep of the variable, then the start and/or end is shifted to the previous valid time (e.g. for a period from 15:20 hours to 16:20 hours and a whole hour timeStep the period is shifted to be 15:00 hours to 16:00 hours).

## timeSeriesSet

A time series set that can be used as input for a component.

## transformation

Definitions of transformations that can be used to transform timeseries from the database in a way that matches the required behaviour in the display.

### Transformations within ScadaDisplay

Up to 2011\_01 it is necessary that all data to be displayed in the ScadaDisplay is beforehand available as time series. This includes simple sums and differences between other time series. From 2011\_02 onwards, it is possible to include in the ScadaDisplay configuration one or more transformations. These transformations will make it easier to use derived time series. The derived timeseries will be calculated on-the-fly as temporary timeseries. The transformations are processed in the order they appear in the configuration.

**NB.** For 2011\_02 only the UserSimpleFunction is supported and tested as a transformation that can be used.

### Sample configuration for transformations within the ScadaDisplay

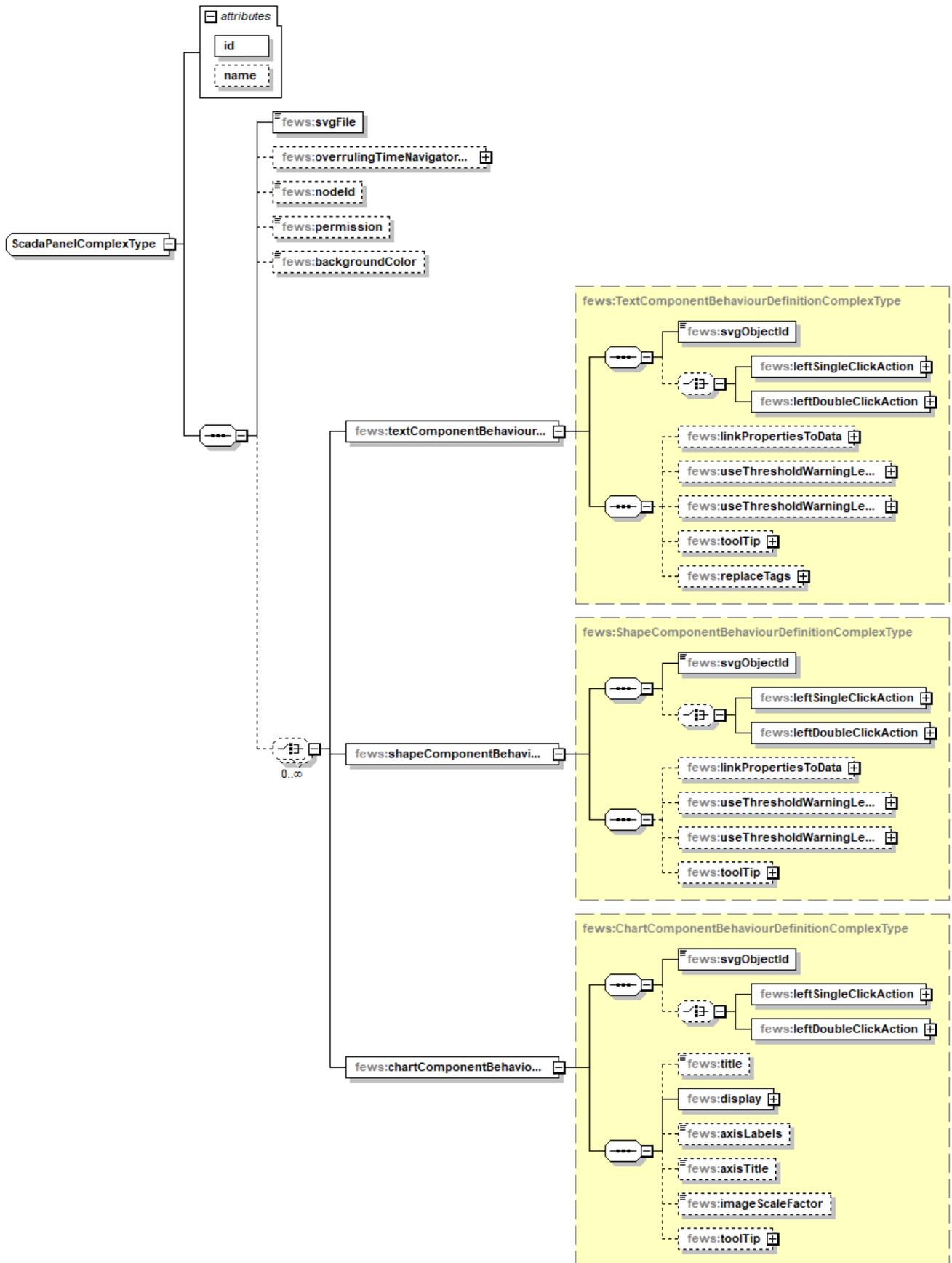
**NB.** It is required that the timeSeriesType of output variables are set to *temporary*.

This sample will allow the variable with variableId *Observation\_minus\_correction* to be displayed on a scadaPanel. This variable refers to a temporary timeseries that will be updated on-the-fly with the difference between the two other variables *Observation* and *Correction*.

## scadaPanel

The ScadaPanel element is the central element in the configuration. The configuration can hold one or more definitions of schematic status panels. In the user interface each schematic status panel will be available from the list in this schematic status display.

Most important within this configuration element is the specification of behaviour for text-objects, shape-objects or chart-objects.



**id**

Identifier of this schematic status panel.

**name**

The name of this schematic status panel as it is displayed in the user interface. If not specified, then id is used as name.

**svgFile**

The name of an svg (Scalable Vector Graphics) file in the ReportImageFiles directory. This schematic status panel shows all svg objects that are defined in the specified svg file. The svg objects in the svg file can be given special behaviour and/or properties using the configuration below. See <http://www.w3.org/TR/SVG11/> for the SVG 1.1 specification.

**overrulingTimeNavigatorTimeStep**

Since 2015.02. Optional. The timestep by which the time navigator slider is moved for this scada panel. This overrules the timeNavigatorTimeStep that is configured for the scada display as a whole. This timestep is only used when showTimeNavigatorToolbar is configured for the scada display.

**nodeld**

Optional. Identifier that refers to a node in the topology configuration file. If specified, then the referenced topology node will be selected when this scadaPanel is selected in the user interface. When the topology node is selected, then that may cause other things to be selected as well, like e.g. the displayGroup in the TimeSeriesDisplay that corresponds to that node.

**backgroundColor**

Optional configuration for the background color of a specific panel. If no color is configured, the background color that is configured for the scada display will apply.

**permission**

Optional. Hides the panel when this permission is not available for the logged on user in the Permissions.xml in theSystemConfigFiles.

**textComponentBehaviourDefinition**

Text objects can obtain special behaviour in this schematic status panel via the textComponentBehaviourDefinition. Each item refers to an svg object of type "text" that is defined in the given svg file. An svg object of type "text" can be a "text", "tspan", "tref", "textPath" or "altGlyph" element. Each item can be assigned behaviour and/or properties for that object. This way it is possible to e.g. replace tags in the text of a text object with certain values from a certain time series, or to define what should happen when the user clicks on a certain component.

Text component Behaviour configuration options available are:

- leftSingleClickAction
- leftDoubleClickAction
- linkPropertiesToData
- useThresholdWarningLevelSymbols
- useThresholdWarningLevelColors
- toolTip
- replaceTags

**shapeComponentBehaviourDefinition**

Shape objects can obtain special behaviour in this schematic status panel via the shapeComponentBehaviourDefinition. This refers to an svg object of type "shape" that is defined in the given svg file. An svg object of type "shape" can be a "path", "rect", "circle", "ellipse", "line", "polyline" or "polygon" element. Each item can be assigned behaviour and/or properties for that object. This way it is possible to e.g. give a color to an object based on a threshold crossing of a timeseries.

Shape component Behaviour configuration options available are:

- leftSingleClickAction
- leftDoubleClickAction
- linkPropertiesToData
- useThresholdWarningLevelSymbols
- useThresholdWarningLevelColors
- toolTip
- replaceTags

**chartComponentBehaviourDefinition**

Since 2021.02, one can also include small thumbnail graphs referencing a display from the DisplayGroups in a schematic status panel. Such charts are implicitly linked to a *leftSingleClickAction* that opens the same display in a new time series dialog. By default, this the left click is disabled. But if the user wishes, *leftSingleClickAction* can be configured to enable the possibility to open a new time series dialog.

Charts are rendered as SVG `<image>` elements for embedding them in a schematic status panel. A practical problem with this is that empty `<image>` elements in a SVG file are considered in error and SVG editors (i.e. Inkscape) that typically are used to prepare the SVG files will not allow you to insert `<image>` elements without an actual image attached. Therefore any shape element (`<rect>` is recommend) that has width and height properties can be used as a placeholder, that will be replaced by an image as the SVG file is processed.

Shape component Behaviour configuration options available are:

- `leftSingleClickAction`
- `leftDoubleClickAction`
- `title`
- `display` (referencing a display from the `displaygroups.xml` configuration)
- `axisLabels`
- `axisTitle`
- `imageScaleFactor`
- `toolTip`

### display

Required reference to display from the DisplayGroups with option to overrule view period.

### axisLabels

Optional boolean to indicate of the axisLabels from the display-configuration should be included.

### axisTitle

Optional boolean to indicate of the axisTitle from the display-configuration should be included.

### imageScaleFactor

Optional scale factor to fine tune the resolution of the chart as the size differs from the original size in the TimeSeriesdisplay.

Below is an example of a `chartComponentBehaviourDefinition` element, including some of the possible configurations.

```
<chartComponentBehaviourDefinition>
  <svgObjectId>chart_Ijsselmeer</svgObjectId>
  <title>Waterstand IJsselmeer</title>
  <display>
    <displayId>Waterstand_IJsselmeer</displayId>
    <overrulingRelativeViewPeriod unit="hour" start="-48" end="72"/>
  </display>
  <axisLabels>true</axisLabels>
  <axisTitle>true</axisTitle>
  <imageScaleFactor>1.5</imageScaleFactor>
</chartComponentBehaviourDefinition>
```

## Component behaviour configuration options

### svgObjectId

The id of the object in the svg file for which this item defines special behaviour and/or properties.

### leftSingleClickAction

Action that is triggered when the user clicks once on this object with the left mouse button.

### leftDoubleClickAction

Action that is triggered when the user double clicks on this object with the left mouse button. The action cannot be combined with a `leftSingleClickAction`

### toolTip

Optional. If specified, then a toolTip with the specified text is displayed for this component.

### linkPropertiesToData

Optional. Contains options to link properties of this component to actual data values. For example the height of the component can be changed depending on the data values of a specified variable.

#### useThresholdWarningLevelColors

Optional. If specified, then the data for the specified variable within the specified relative view period is used to determine threshold crossings. For crossed thresholds, warningLevels are activated. The color of the most severe activated warningLevel is used as the fill and/or stroke color for the component, as specified.

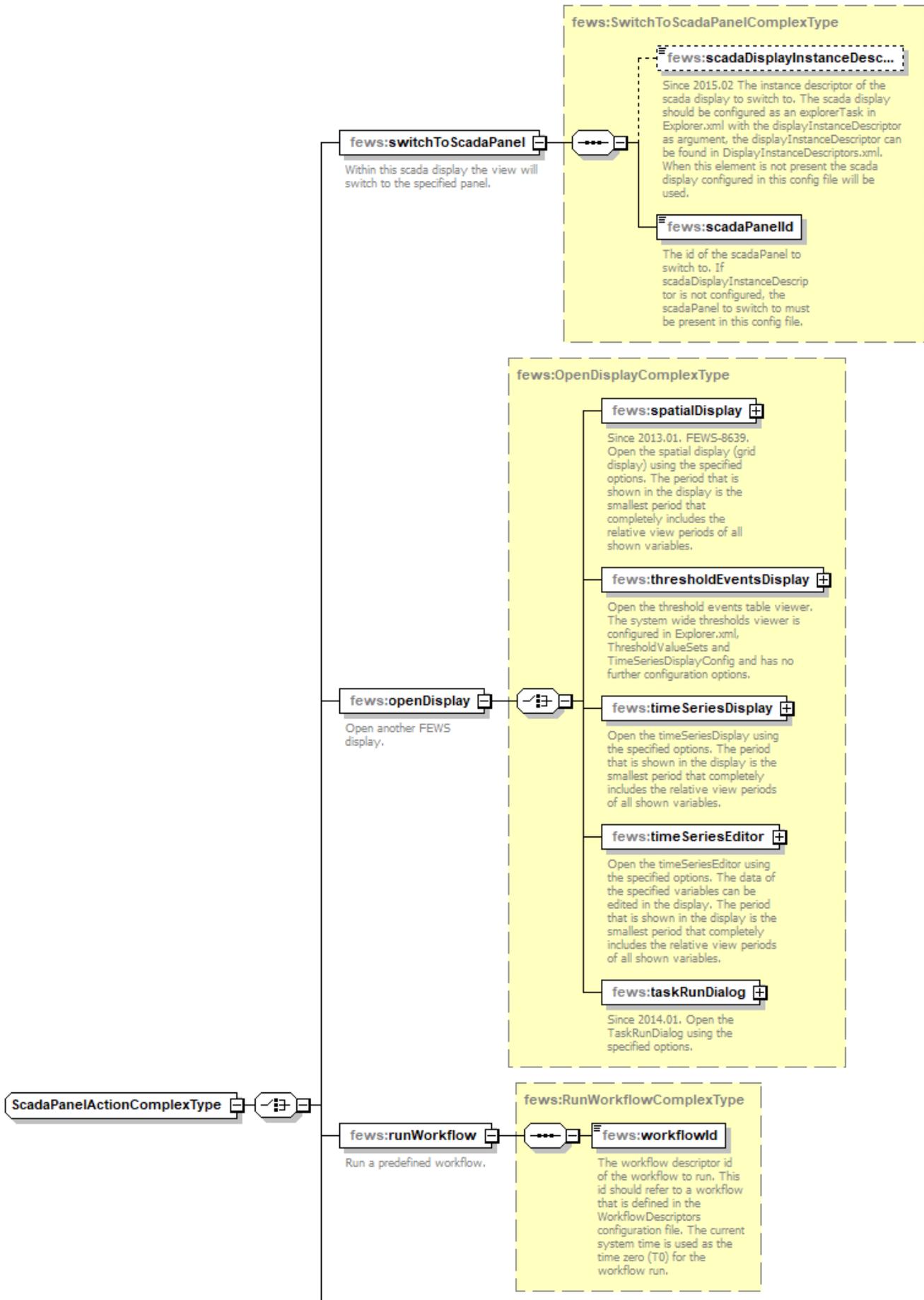
#### useThresholdWarningLevelSymbols

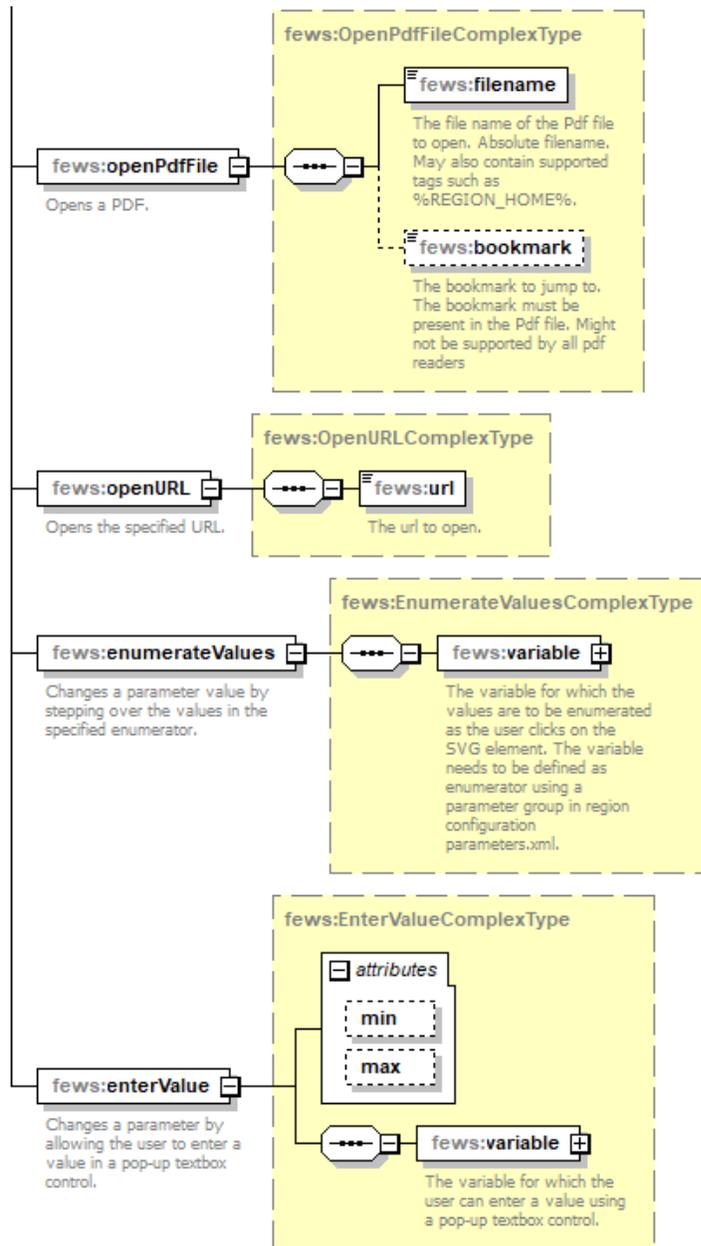
Optional. If specified, then the data for the specified variable within the specified relative view period is used to determine threshold crossings. For crossed thresholds, warningLevels are activated. The symbolId property of the most severe activated warningLevel is used to load a graphic from the <defs/> section of a SVG document. This is very similar to how the ThresholdWarningLevelColors works, but can be applied only to <use/> elements in an SVG file as these elements support embedding an element from the <defs/> section of the SVG document using a xlink:href attribute.

#### replaceTags

If specified, then the tags in the text of this component are replaced using data from the specified variable. See [#tags](#).

### **Left Single Click Action and Left Double Click Action Configuration Options**





Generated by XMLSpy

www.altova.com

Click action configuration elements

`switchToScadaPanel`

Within this schematic status display the view will switch to the specified panel, specified by `scadaPanelId`. To specify a panel in a different `scadaDisplay`, also specify the `scadaDisplayInstanceDescriptor`.

`scadaPanelId`

The id of the `scadaPanel` to switch to. The `scadaPanel` to switch to must be present in this config file.

`openDisplay`

Open another Delft-FEWS display.

`spatialDisplay`

Opens the Spatial Display with specified title and `gridPlotId`. The slider in the Spatial Display is set to the slider time that was active in the Schematic Status Display.

## timeSeriesDisplay

Open the timeSeriesDisplay using the specified options. The period that is shown in the display is the smallest period that completely includes the relative view periods of all shown variables. The option to directly open a display from the DisplayGroups is available from version 2012.01.

## timeSeriesEditor

Open the timeSeriesEditor using the specified options. The data of the specified variables can be edited in the display. The period that is shown in the display is the smallest period that completely includes the relative view periods of all shown variables.

## taskRunDialog

Opens the TaskRunDialog from the DisplayConfigurations with the specified config filename.

## thresholdEventsDisplay

Opens the system wide threshold events table viewer with the specified title and icon.

*(note, if the threshold events table is defined as explorer task in explorer.xml config, please specify the same title and icon here for consistent behaviour)*

## title

Title of the display window.

## variable

One or more variables to define the data that is shown in the display.

## showGraph

Optional. If true then a graph is shown in the opened display. If false, then the graph is not shown.

## showTable

Optional. If true then a table is shown in the opened display. If false, then the table is not shown.

## runWorkflow

Run a predefined workflow.

- workflowId (required): The workflow descriptor id of the workflow to run. This id should refer to a workflow that is defined in the WorkflowDescriptors configuration file. The current system time is used as the time zero (T0) for the workflow run.
- localRun (optional, default=false, available since 2022.0): if set to true, the workflow will be run locally instead on the server. The result of the local run will only be saved on the server if autoSaveToServerAfterLocalRun is set to true.
- autoSaveToServerAfterLocalRun (optional, default=false, available since 2022.01). It can only be set to true if localRun is also true. If autoSaveToServerAfterLocalRun is configured as true, the locally un workflow will be saved on the server.

```
<runWorkflow>
  <workflowId>ID</workflowId>
  <localRun>true</localRun>
  <autoSaveToServerAfterLocalRun>true</autoSaveToServerAfterLocalRun>
</runWorkflow>
```

## OpenPdfFile

Opens in a PDF viewer the specified file at the specified bookmark. Note: not all PDF viewer support focus at the specifiek bookmark.

## OpenURL

Opens the specified URL.

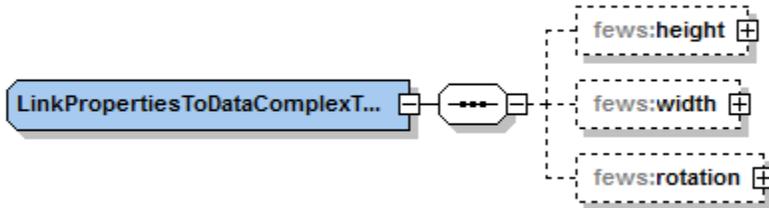
## enumerateValues

Steps through the possible values of a variable, of which the parameter has to be linked to an enumeration in the parameters configuration file. Every time the display element is clicked the next value (code) of the enumeration is assigned to all time steps in the defined timeSeriesSet and overrulingRelativeViewPeriod for the specified variable.

## enterValue

Pops up a dialog where the user can input a new value for the specified variable, the new value is assigned to all time steps in the defined timeSeriesSet and overrulingRelativeViewPeriod for the specified variable. Optional min and max attributes can be provided to limit the range (minimum, maximum) of values that can be entered.

### Link Properties To Data Configuration Options

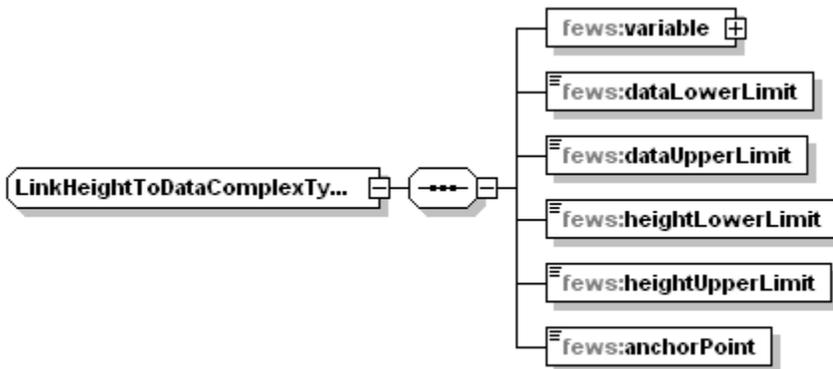


Link properties to data configuration elements

There are several options available:

- height
- width
- rotation

### Link Height To Data Configuration Options



Generated by XMLSpy

www.altova.com

Link height to data configuration elements

Optional. If specified, then for this component the height attribute is linked to the data values for the specified variable. This option can only be used for svg objects of type "rect". The data limits refer to the range of the timeseries. The height limits refer to the height of the object within the svg file for the unit (e. g. pixels or mm) defined in the document properties. If the data value is less than dataLowerLimit, then the height is set to heightLowerLimit. If the data value is greater than dataUpperLimit, then the height is set to heightUpperLimit. If the data value is between dataLowerLimit and dataUpperLimit, then the height will be linearly interpolated between heightLowerLimit and heightUpperLimit. If no data is available, then this component is made invisible.

Note: it is required that dataUpperLimit is greater than dataLowerLimit. However it is possible to define heightUpperLimit less than heightLowerLimit to control the direction of the change of the height.

variable

The data for this variable is used to determine the height for this component.

dataLowerLimit

If the data value is less than or equal to dataLowerLimit, then the height will be equal to heightLowerLimit.

dataUpperLimit

If the data value is greater than or equal to dataUpperLimit, then the height will be equal to heightUpperLimit.

heightLowerLimit

The height that corresponds to the dataLowerLimit value.

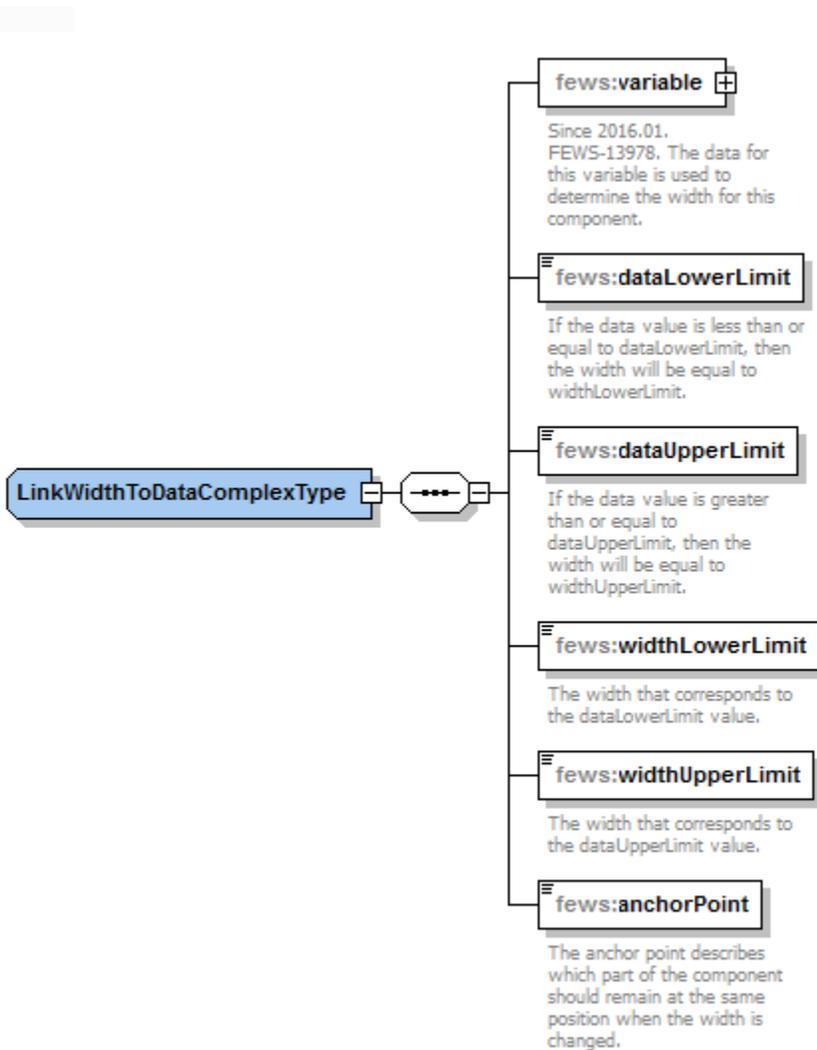
heightUpperLimit

The height that corresponds to the dataUpperLimit value.

anchorPoint

The anchor point describes which part of the component should remain at the same position when the height is changed. Can be "bottom", "top" or "center".

## Link Width To Data Configuration Options



### Link width to data configuration elements

Since 2016.01. Optional. If specified, then for this component the width attribute is linked to the data values for the specified variable. This option can only be used for svg objects of type "rect". The data limits refer to the range of the timeseries. The width limits refer to the width of the object within the svg file for the unit (e.g. pixels or mm) defined in the document properties. If the data value is less than dataLowerLimit, then the width is set to widthLowerLimit. If the data value is greater than dataUpperLimit, then the width is set to widthUpperLimit. If the data value is between dataLowerLimit and dataUpperLimit, then the width will be linearly interpolated between widthLowerLimit and widthUpperLimit. If no data is available, then this component is made invisible.

Note: it is required that dataUpperLimit is greater than dataLowerLimit. However it is possible to define widthUpperLimit less than widthLowerLimit to control the direction of the change of the width.

#### variable

The data for this variable is used to determine the height for this component.

#### dataLowerLimit

If the data value is less than or equal to dataLowerLimit, then the height will be equal to heightLowerLimit.

#### dataUpperLimit

If the data value is greater than or equal to dataUpperLimit, then the height will be equal to heightUpperLimit.

#### widthLowerLimit

The width that corresponds to the dataLowerLimit value.

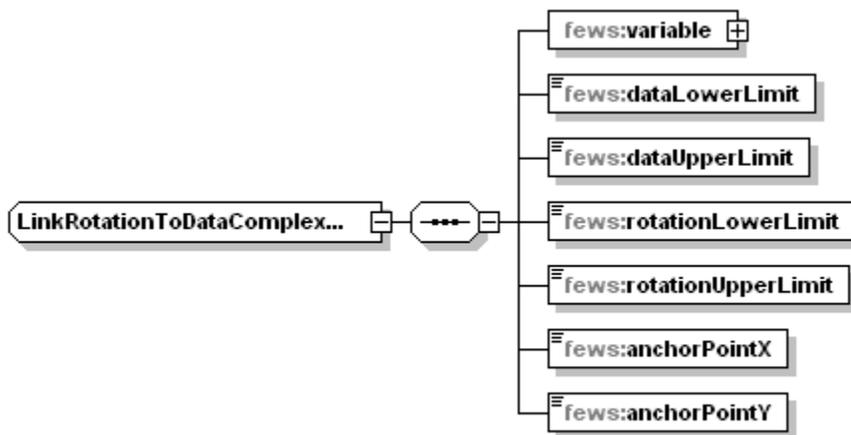
#### widthUpperLimit

The width that corresponds to the dataUpperLimit value.

#### anchorPoint

The anchor point describes which part of the component should remain at the same position when the height is changed. Can be "left", "right" or "center".

### Link Rotation To Data Configuration Options



Generated by XMLSpy

www.altova.com

Link rotation to data configuration elements

Optional. If specified, then for this component the rotation is linked to the data values for the specified variable. The rotation that is derived from the data values is always relative to the rotation angle that is specified for this component in the svg file. This option can only be used for svg objects of type "path", "rect", "circle", "ellipse", "line", "polyline", "polygon" or "text". If the data value is less than dataLowerLimit, then the rotation is set to rotationLowerLimit. If the data value is greater than dataUpperLimit, then the rotation is set to rotationUpperLimit. If the data value is between dataLowerLimit and dataUpperLimit, then the rotation will be linearly interpolated between rotationLowerLimit and rotationUpperLimit. If no data is available, then this component is made invisible. If the data value is flagged as "varying direction" (e.g. varying wind direction), then the rotation will increase linearly in time (animation). In this case the rotation will increase from rotationLowerLimit to rotationUpperLimit and then start from rotationLowerLimit again.

Note: it is required that dataUpperLimit is greater than dataLowerLimit. However it is possible to define rotationUpperLimit less than rotationLowerLimit to control the direction of the rotation. If rotationUpperLimit is greater than rotationLowerLimit, then increasing data values result in clockwise rotation.

#### variable

The data for this variable is used to determine the rotation for this component.

#### dataLowerLimit

If the data value is less than or equal to dataLowerLimit, then the rotation will be equal to rotationLowerLimit.

#### dataUpperLimit

If the data value is greater than or equal to dataUpperLimit, then the rotation will be equal to rotationUpperLimit.

## rotationLowerLimit

The rotation (in degrees) that corresponds to the dataLowerLimit value. This rotation is always relative to the rotation angle that is specified for this component in the svg file.

## rotationUpperLimit

The rotation (in degrees) that corresponds to the dataUpperLimit value. This rotation is always relative to the rotation angle that is specified for this component in the svg file.

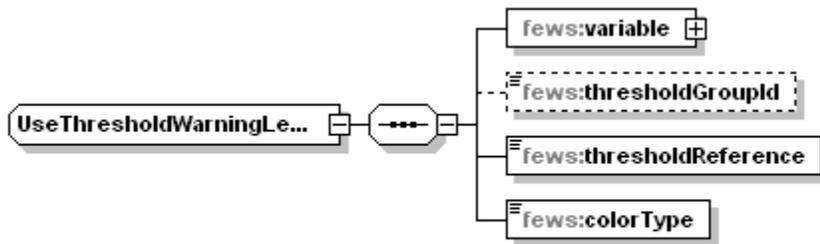
## anchorPointX

The x coordinate of the anchor point. The rotation will be around the anchor point. This x coordinate has to be specified in the user space coordinate system of the svg object for this component in the svg file. The user space coordinate system is determined by all transforms that are specified in the parent svg objects of the svg object for this component. All transforms that are specified in the svg object itself are not part of the user space coordinate system and thus should be taken into account in the coordinates that are specified here. E.g. to rotate a "rect" svg object with attributes width="200" height="200" x="500" y="300" transform="translate(50 0)" around its center, use anchorPoint coordinates (x, y) = (650, 400). See also [Determining the rotation anchor point for an SVG object in user space coordinates](#).

## anchorPointY

The y coordinate of the anchor point. The rotation will be around the anchor point. This y coordinate has to be specified in the user space coordinate system of the svg object for this component in the svg file. The user space coordinate system is determined by all transforms that are specified in the parent svg objects of the svg object for this component. All transforms that are specified in the svg object itself are not part of the user space coordinate system and thus should be taken into account in the coordinates that are specified here. E.g. to rotate a "rect" svg object with attributes width="200" height="200" x="500" y="300" transform="translate(50 0)" around its center, use anchorPoint coordinates (x, y) = (650, 400). See also [Determining the rotation anchor point for an SVG object in user space coordinates](#).

## Use Threshold Warning Level Colors Configuration Options



Generated by XMLSpy

www.altova.com

Use threshold warning level colors configuration elements

### variable

The data for this variable is used to determine threshold crossings. For crossed thresholds, warningLevels are activated. The color of the most severe activated warningLevel is used as the fill and/or stroke color for the component, as specified below.

### thresholdGroupId

Optional. If specified, then only thresholds in the specified thresholdGroup are used in the determination of threshold crossings and warningLevels for the specified variable. If not specified, then thresholds in all thresholdGroups are used.

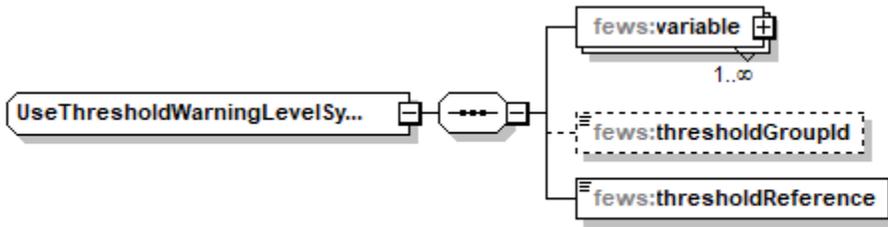
### thresholdReference

Specify which data is used for determining threshold crossings. Either choose the first or last reliable or doubtful value within the relative view period, or choose all reliable or doubtful values within the relative view period. Can be "first\_value", "last\_value" or "relative\_view\_period".

### colorType

Specify which color type (fill and/or stroke) should be changed to use warningLevel colors. Color types that are not specified here are not changed. Can be "fill", "stroke" or "fill\_and\_stroke".

## Use Threshold Warning Level Symbols Configuration Options



Generated by XMLSpy

www.altova.com

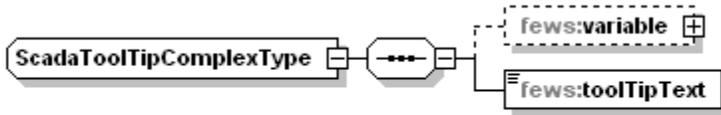
The fields in this xml configuration are the same as for the threshold warning level colors described above, except that the `fews:colortype` field is not applicable here.

In the `warninglevels.xml` configuration, both a color and a symbol can be specified (optionally), the `symbolld` can be applied only to `<use/>` elements in a SVG document where the setup is as follows:

```
<svg ...>
  <defs>
    <path id="..." d="..." />
    <polyon id="..." points="..." />
  </defs>
  ...
  <use xlink:href="#id" .... />
</svg>
```

The `xlink:href` attribute will lookup any graphic in the `defs` section with the given `id` and embed this in the `<use/>` element, which can contain transforms, fill and stroke definitions that can be manipulated in other ways at the same time. So by putting the "fill" attributes in the `<use/>` element, it should be possible to apply both a color and symbol from a `ThresholdWarningLevel` at the same time.

### Tooltip Configuration Options



Generated by XMLSpy

www.altova.com

Tooltip configuration elements

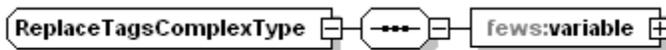
#### variable

The data from this variable is used to replace the tags in the specified `toolTipText`. If for a given tag the required data is not available, then that tag is replaced by a dash symbol "-". This variable is only required if the specified `toolTipText` contains tags.

#### toolTipText

Text that is displayed in the tooltip for this component. See the next section on tags.

### Replace Tags Configuration Options



Generated by XMLSpy

www.altova.com

If specified, then the tags in the text of this component are replaced using data from the specified variable. Tags should be separated by "%" signs. Text can be e.g. "Last value = %LASTVALUE(numberFormatId)%", which would be replaced by e.g. "Last value = 10.0". The following tags can be used in the text (numberFormatId/dateFormatId should be replaced by the id of a numberFormat/dateFormat that is defined at the start of this configuration file):

<code>%CURRENTTIME(dateFormatId)%</code>	the local time of the client.
<code>%STARTTIME(dateFormatId)%</code>	the start date and time of the relative view period of the time series. Requires the relative view period to be configured.
<code>%ENDTIME(dateFormatId)%</code>	the end date and time of the relative view period of the time series. Requires the relative view period to be configured.
<code>%LASTVALUE(numberFormatId)%</code>	the most recent reliable or doubtful value in the time series. In case of enumeration parameter, omit the numberformatId.

%LASTVALUETIME (dateFormatId)%	the date and time of the most recent reliable or doubtful value in the time series.
%LASTVALUECOMMENT()%	the comment of the most recent reliable or doubtful value present in given time series array
%FIRSTVALUE (numberFormatId)%	the first reliable or doubtful value in the time series. In case of enumeration parameter, omit the numberformatId
%FIRSTVALUETIME (dateFormatId)%	the date and time of the first reliable or doubtful value present in given time series array
%FIRSTVALUECOMMENT()%	the comment of the first reliable or doubtful value present in given time series array
%MINVALUE(numberFormatId)%	the minimum reliable or doubtful value in the time series. In case of enumeration parameter, omit the numberformatId.
%MAXVALUE(numberFormatId)%	the maximum reliable or doubtful value in the time series. In case of enumeration parameter, omit the numberformatId.
%EXTERNALFORECASTINGSTARTTIME(dateFormatId)%	the start of the external forecast.
%MAXTIME(dateFormatId)%	the date/time of maximum value found in the time series (occurrence closest to T0).
%MINTIME(dateFormatId)%	the date/time of minimum value found in the time series (occurrence closest to T0).
%TIMEZERO(dateFormatId)%	in case of forecast data the forecast time, otherwise timezero.
%MAXWARNINGLEVEL (thresholdId)%	name of the highest warning level threshold that has been crossed (requires an available thresholdValueSet)
%LOCATIONATTRIBUTE (textAttribute)%  %LOCATIONATTRIBUTE (numberAttribute;numberFormat)%	attribute to put in text. the numberFormat is intended for number attributes.
%LOCATIONATTRIBUTE (numberAttribute;numberFormat; variableId)%	<i>This option can only be used within the SSD configuration (so not the SVG file itself), and only within the title element of the openDisplay element of leftSingleClickAction and rightSingleClickAction. It is supported since 2017.02</i>  VariableId is intended for the case that multiple variables have been configured.  If no numberFormat is used the tag should be %LOCATIONATTRIBUTE(numberAttribute;;variableId)%.  VariableId is optional if only one variable is configured. At least a locationId or locationSet has to be configured for the variable.
%PARAMETERNAME (variableId)%	Since 2017.02. Used to set title in openDisplay leftSingleClickAction and rightSingleClickAction. Variable is optional if only one variable was configured.



The *Relative View Period* specified in the variables of the Schematic Status Display is always relative to the sliderTime. While moving the slider in the timeNavigator, the sliderTime is used as TimeZero. The choice of Relative View Period is therefore essential for the best behaviour of the Schematic Status Display. For instance, for displaying the time series value at the sliderTime using the LASTVALUE tag, the configured variable should have a Relative View Period with length 0. But for instance when using the click action that opens the TimeSeriesDialog, a separate second variable should be used with a non-zero Relative View Period.

variable

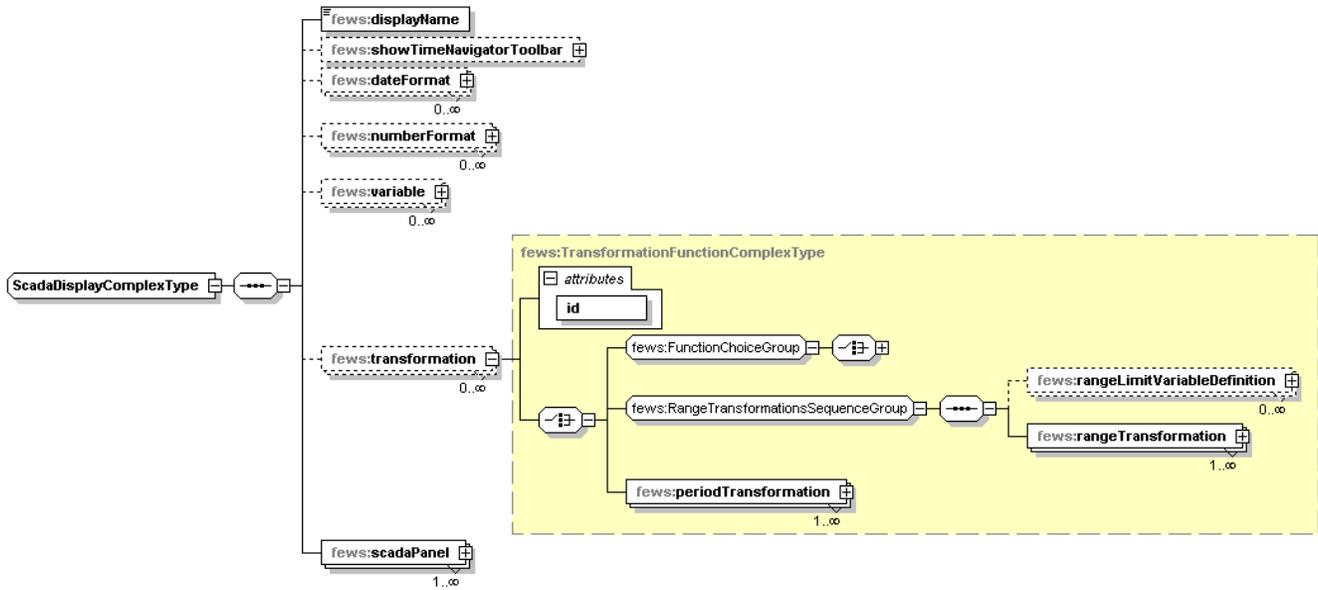
The data from this variable is used to replace the tags in the text in the svg object that this component refers to. If for a given tag the required data is not available, then that tag is replaced by a dash symbol "-".

```

<variable>
  <variableId>Observation</variableId>
  <timeSeriesSet>
    <moduleInstanceId>Afgeleide_Twentekanalen</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>H.meting</parameterId>
    <locationSetId>Hydro_LMW_TK_H</locationSetId>
    <timeSeriesType>external historical</timeSeriesType>
    <timeStep unit="hour"/>
    <relativeViewPeriod unit="hour" start="-1" end="0"/>
    <readWriteMode>read only</readWriteMode>
  </timeSeriesSet>
</variable>
<variable>
  <variableId>Correction</variableId>
  <timeSeriesSet>
    <moduleInstanceId>Afgeleide_Twentekanalen</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>H.meting</parameterId>
    <locationSetId>Hydro_LMW_TK_H</locationSetId>
    <timeSeriesType>external historical</timeSeriesType>
    <timeStep unit="hour"/>
    <relativeViewPeriod unit="hour" start="-1" end="0"/>
    <readWriteMode>read only</readWriteMode>
  </timeSeriesSet>
</variable>
<variable>
  <variableId>Observation_minus_correction</variableId>
  <timeSeriesSet>
    <moduleInstanceId>Afgeleide_Twentekanalen</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>H.meting</parameterId>
    <locationSetId>Hydro_LMW_TK_H</locationSetId>
    <timeSeriesType>temporary</timeSeriesType>
    <timeStep unit="hour"/>
    <relativeViewPeriod unit="hour" start="-1" end="0"/>
    <readWriteMode>add originals</readWriteMode>
    <synchLevel>9</synchLevel>
  </timeSeriesSet>
</variable>
<transformation id="TransformationObservationMinusCorrection">
  <user>
    <simple>
      <expression>Observation - Correction</expression>
      <outputVariable>
        <variableId>Observation_minus_correction</variableId>
      </outputVariable>
    </simple>
  </user>
</transformation>

```

...



## Navigation using Topology

To navigate different SSD displays and panels within a SSD display, (starting with FEWS release 2019.02) specific elements can be added to the topology.xml configuration, as in the following example.

```

<nodes id="SSD_NL" name="SSD NL">
  <node id="Overzichtsscherm_WMCN" name="Overzichtsscherm Nederland">
    <mainPanel>schematic status display</mainPanel>
    <scadaDisplayId>ScadaOverzichtsscherm_WMCN</scadaDisplayId>
    <scadaPanelId>Overzichtsscherm_WMCN</scadaPanelId>
  </node>
  <node id="Verwachtingenscherm_WMCN" name="Verwachtingen Nederland">
    <mainPanel>schematic status display</mainPanel>
    <scadaDisplayId>ScadaOverzichtsscherm_WMCN</scadaDisplayId>
    <scadaPanelId>Verwachtingenscherm_WMCN</scadaPanelId>
  </node>
</nodes>

```

### mainPanel

This element will bring the SSD display specified in the <scadaDisplayId> element to the front if it has already been opened. It will not open a new window if no SSD display with the given ID is open..

### scadaDisplayId

This element will first check if the SSD display with the given ID is already open, if so, this display will be brought to front to give it the focus. If the SSD display is not already open, explorer will try to re-use another SSD display and load the SSD display with given ID to replace the current content. If no SSD display is open, nothing happens, at least one SSD display must be opened as an explorer task for it to be able to respond to selection changes in the Topology panel.

### scadaPanelId

This element can be combined with the scadaDisplayId element to navigate to a specific SSD panel in the given display. Explorer will first look for an open display which contains the relevant SSD display and selected the SSD panel in this display as if the user selected this from the list of available SSD panels in this display.

*tip: When using the topology nodes to navigate through SSD displays and panels, it is recommended that for each SSD panel, a noideId field is added to the SSD display configuration to link back from each SSD panel to the relevant topology node. If you do so properly you can click either on the topology node or on a item in the list of panels to select a SSD panel and both selections will appear synchronized. If you don't, the selected topology node may not be in synch. with the selection in the SSD display which may be a bit confusing to the user.*

## Known Issues

When using Delft-FEWS the configuration can be present as files on the file system or can be contained in a local data store. The svg files that are used for the schematic status display work in both cases. However, if the svg files refer to separate image files, then the schematic status display can only display these images if the image files are present as files on the file system. If these separate image files are contained in a local data store, then they cannot be displayed in the schematic status display. Therefore, when using separate image files, make sure that the configuration is present as files on the file system. If this is not possible, then it is possible to choose from two different workarounds:

- The first possible workaround is to not use separate image files. For a schematic image it is possible to create svg elements that resemble the contents of the image. If these svg elements are added to the svg file for the schematic status display, then there is no need to use the image file anymore.
- The second possible workaround is to use embedded images instead of separate image files. The section [Embedding image files into SVG files](#) describes how to do this.

## Tips And Tricks

### SVG specification

The schematic status display uses SVG files. For details on the format and possibilities of SVG files, please refer to <http://www.w3.org/TR/SVG11/> for the SVG 1.1 specification.

### Embedding image files into SVG files

It is possible to embed image files into an SVG file. If an image file is embedded into an SVG file, then the original image file is no longer needed, because the image data is then available from the SVG file itself. To embed an image file into an SVG file using Inkscape, do the following. Open the SVG file in Inkscape. Add one or more images to the file in the normal way. Then select menu "Extensions", then select "Images", then select "Embed Images". Then click "Apply". Then save the SVG file in the normal way. Now the images are embedded into the SVG file. If all the images in an SVG file are embedded, then Delft-FEWS only needs the SVG file itself (and not the original image files) for displaying in the scada display.

### Controlling the resizing behaviour of an svg document within the scada display

In an svg file in the root element use the following attributes to control its resizing behaviour: width, height, viewBox, preserveAspectRatio.

- If only width and height present, then the svg document gets an absolute size, appears in the top-left corner of the display and is never resized (not even when the display window is resized). This means it can be cut-off when the display window is too small.
- If only viewBox and preserveAspectRatio are present, then the viewBox determines the rectangular region of the svg document that is drawn in the display window (the coordinates for the viewBox edges are the same as the coordinate system used within the svg file, usually the coordinates are in pixels). The preserveAspectRatio determines how the drawn region is sized and aligned within the display window. In this case the svg document is automatically resized when the display window is resized.

Examples:

Resizing attributes in the svg root element	Resizing behaviour
<pre>viewBox="0 0 400 400" preserveAspectRatio="xMidYMid meet"</pre>	The svg document is scaled to fit the display window and the aspect ratio is preserved.
<pre>viewBox="0 0 400 400" preserveAspectRatio="none"</pre>	The svg document is scaled and stretched to fill the window (aspect ratio is not preserved).
<pre>width="1200" height="700"</pre>	Only the region with coordinates $0 \leq x \leq 1200$ and $0 \leq y \leq 700$ pixels is shown. The svg document is not resized when the display window is resized.

Background information:

The width and height attributes in the root svg element of an svg file determine the size of the viewport, in other words the size of the svg document when it is viewed. The coordinates of the objects in the svg file are specified in user space, which is different from the viewport space. The viewBox attribute in the root svg element defines the rectangle in user space that should be mapped to the edges of the viewport. The preserveAspectRatio attribute in the root svg element determines how this mapping takes place. This mapping uses one of three possible methods: "meet", "slice" or "none". See <http://www.w3.org/TR/SVG11/coords.html#ViewBoxAttribute> and <http://www.w3.org/TR/SVG11/coords.html#PreserveAspectRatioAttribute> for more detailed information.

### Determining the rotation anchor point for an SVG object in user space coordinates

To determine the rotation anchor point for an SVG object in user space coordinates using Inkscape, do the following. Open the SVG file in Inkscape. Select the object. Select menu "Edit", then select "XML Editor". Then in the window that opens, in the box on the right, look for the important attributes (e.g. "x", "y", "width", "height", "transform" or "d") and use their values to calculate the required anchor point. E.g. to rotate a "rect" svg object with attributes width="200" height="200" x="500" y="300" transform="translate(50 0)" around its center, use anchor point coordinates (x, y) = (650, 400).

### Aligning text within svg text objects

By default text in an svg text object is left-aligned and the x and y coordinates of the object denote the upper-left corner of the object. To right-align text in an svg text object, add the following attribute to the text element:

```
text-anchor="end"
```

The entire element would then e.g. be:

```
<text id="Delft" x="757.48602" y="308.03317" style="fill:#000099" text-anchor="end">%LASTVALUE(numberFormat1)%  
meter</text>
```

When an object is right-aligned, then the x and y coordinates of the object denote the upper-right corner of the object. The attribute text-anchor can also have the values "start" or "middle". To create multiple pieces of text with different alignments, use separate text objects.

When using right-alignment, the decimal separator for number values can be aligned by using the following number format:

```
{0,number,#.00}
```

Here "#" means one or more decimals before the decimal separator and ".00" means that always two decimal places are shown (number is either rounded or padded with zero's).

### **Export maps from ArcGis as svg files**

In ArcGis it is possible to export map as svg file. Go to "File > Export map" and select \*.svg as export file type.

### **Reduce the size of svg files**

Within inkscape the size of an svg file can be reduced by saving it as compressed svg file (\*.svgz) or as plain svg file. Also cleaning up the file by using the "vacuum defs" option in the file menu makes the file significantly smaller.