

01 Grid Display

- Grid display
 - title
 - showPlotTreeHideAllToolWindows
 - defaults
 - timeDisplayLabels
 - gridPlotGroup
 - description
 - highlight
 - gridPlot
 - mapExtentId
 - timeSeriesSet
 - dataLayer
 - showArrowsOnLines
 - wmsLayer
 - trackLayer
 - showThumbnailsPanel
 - openEnsembleThumbnailsWindowWhenSelected
 - legendStyle
 - barLegend
 - uTimeSeriesSet / vTimeSeriesSet
 - uvAmplitudeParameterId (since 2015.02)
 - uvDirectionParameterId (since 2016.01)
 - uvDirectionParameterConvention (since 2016.02)
 - sigmaScaleReferenceTimeSeriesSet (since 2017.02)
 - zParameterId (since 2018.01)
 - Display true color imagery
 - Accumulation / Moving Average slider or Last Value Checkbox
 - verticalSliderRange
 - movieFrameDurationMillis
 - workflow
 - geoMap
 - locationFilterAttributId
 - classBreaks
 - description
 - missingValueColor
 - missingValueOpacityPercentage
 - unitVisible (available since build 18734)
 - rescaleAroundOrdinalValue (available since build 17892)
 - lowerColor
 - upperColor
 - lowerOpacityPercentage
 - upperOpacityPercentage
 - lowerSymbolSize
 - upperSymbolSize
 - lowerValue
 - color
 - break
 - geoMap (or background maps)
 - compare two different simulated forecasts spatially (Since 2020.02)
- Split Grids.xml file in multiple config files with gridPlotGroups (Since 2019.02)
- Extracting data from the Spatial Display
- Configuration (Examples)
 - 3D data display (example)
 - ClassBreak labels
 - Show last import time for external historical data
 - DisplayGroupPlotID
 - Logo at map
 - Showing T0 or last import time underneath plot

Grid display

The grid display is used in Delft-FEWS for viewing grid time series. These grid time series can be dynamically animated against a map background (comparable to the main map display).

Files following the gridDisplay.xsd need to be placed in the DisplayConfigFiles folder and can be referenced in the explorerTask using the displayConfigFileName. If desired gridPlotGroup-specifications can be placed in separate files following the gridPlotGroups.xsd while referencing the gridPlotGroupID in the gridDisplay (available since 2019.02). Since 2020.02, GridPlots with empty locationsets are automatically hidden. Users are informed with an info message.

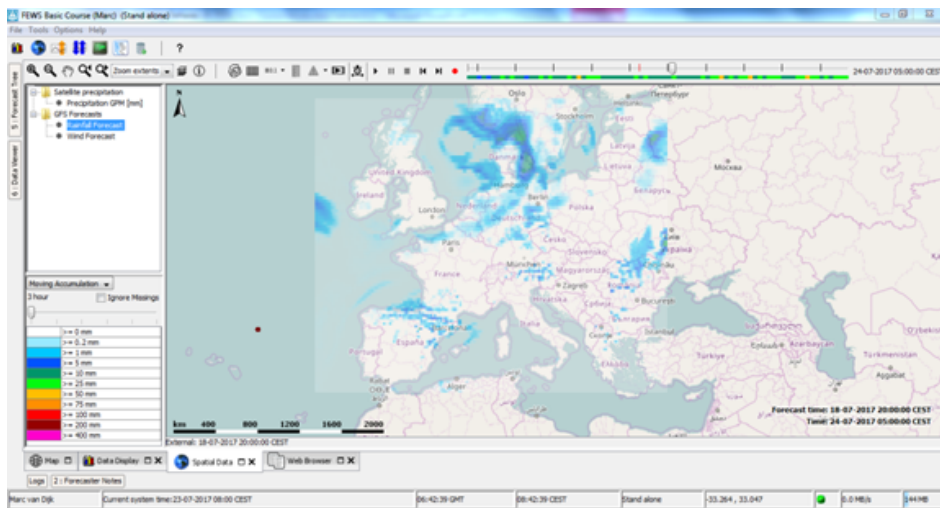


Figure 143 Example of a configuration of the Grid Display

Besides plotting a grid in the grid display, it is also possible to plot scalar data (Figure 10) and longitudinal profiles (figure 9).

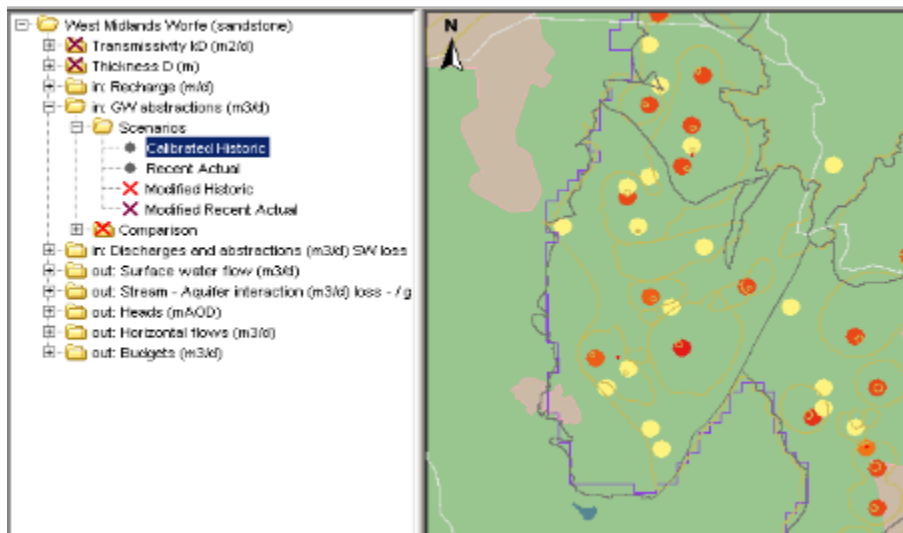


Figure 010 Example of a configuration of the scalar data in Grid Display

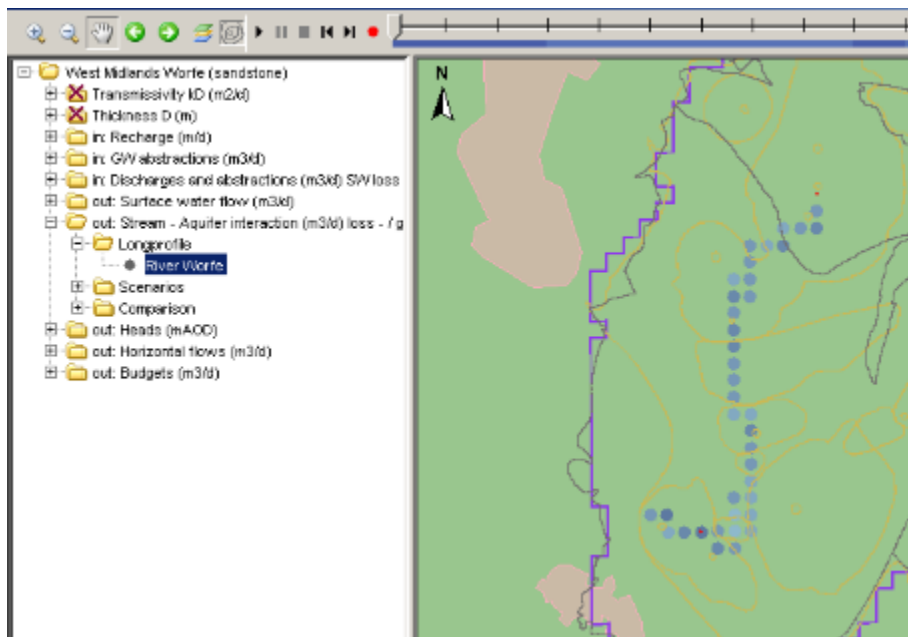


Figure 009 Example of a configuration of the longprofile in Grid Display

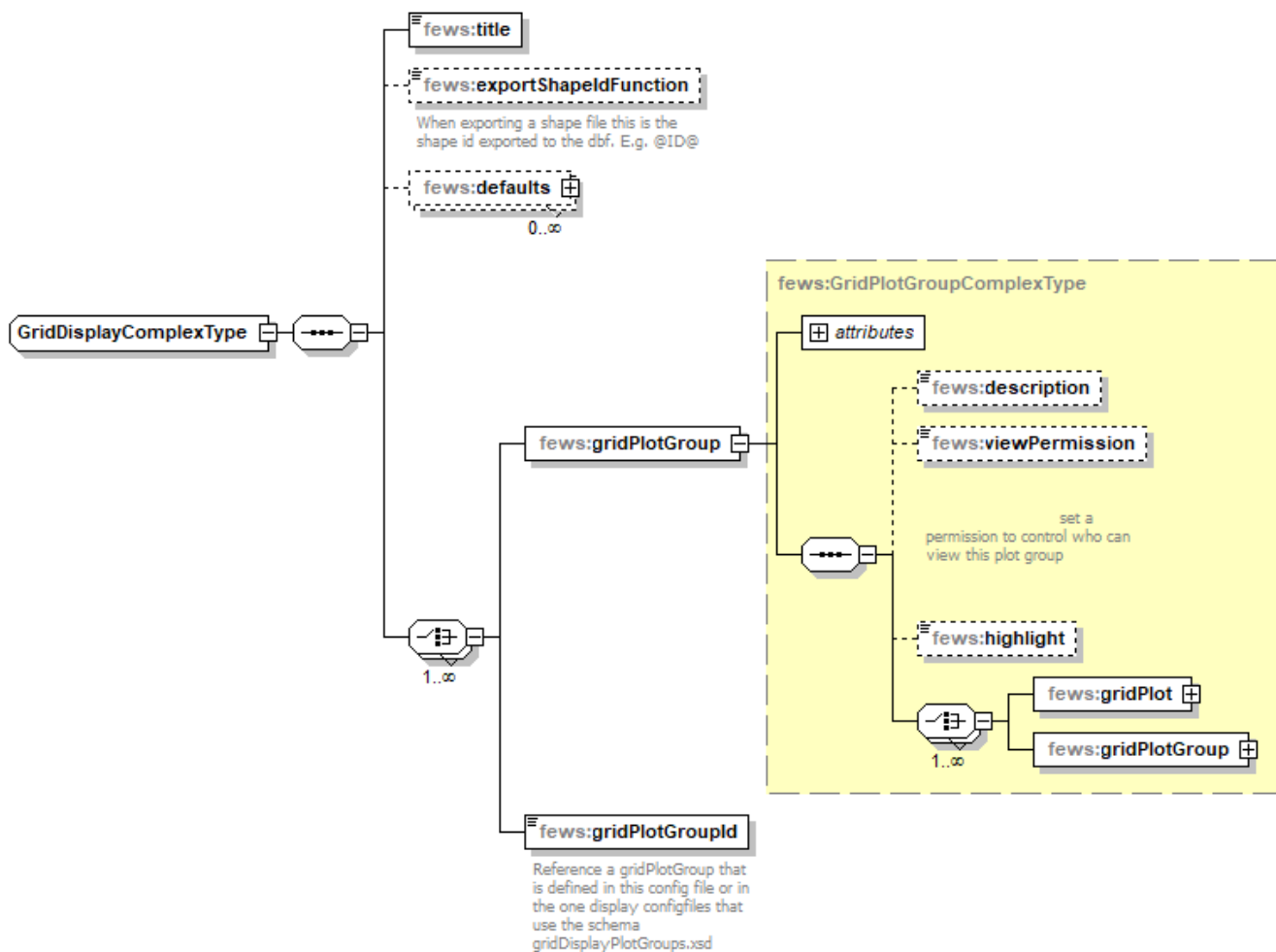


Figure 144 Root elements of the gridDisplay configuration

title

Name of the Grid Display. When opened this will appear in the title bar of the window.

showPlotTreeHideAllToolWindows

If true, indicates that the data viewer and tool windows must be closed when spatial display is configured.

defaults

Used to specify attributes that should apply to several gridPlots. Specifying which gridPlots the defaults apply to can be done through **parameterGroupId**, **plotGroupId** or **plotId**. When an attribute that is specified through defaults is needed, FEWS will search for the first defaults that contains the needed attribute and applies to the current gridPlot. If no parameterGroupId, no plotGroupId and no plotId is specified, the defaults is considered to apply to all gridPlots. Note that such a general defaults may override settings made in more specific defaults if it is found first by FEWS, when this occurs try specifying the most general defaults last.

timeDisplayLabels

When specified, the external forecast time and current time are plotted on the configured position on the map. In case of multiple forecasts, only the latest forecast time is displayed. By default (i.e. when option is not configured) no time labels are displayed on the map.

gridPlotGroup

Definition of a group in the grid display. Each group may have its own set of maps and time series to display. Defining groups creates a tree view in the left of the display (see example above). Multiple instances may exist.

Attributes;

- **id** : Id of the display group- this is used in the tree view.
- **name** : Optional name of the display group- used for reference purposes only

description

Optional description of the display group/grid plot. Used for reference purposes only

highlight

Optional property to highlight the Group name in bold in the selection filter.

gridPlot

Definition of a grid plot within the display group. Each grid plot forms a node in the tree view. When a gridPlot is selected, the appropriate maps will be displayed and the time series data retrieved from the database.

Attributes;

- **id** : Id of the grid plot- this is used in the tree view. If not unique, a warning will be logged.
- **name** : Optional name of the grid plot- used for reference purposes only
- **contourlinescolor**: required when the "contourlines button" in the griddisplay should be enabled

mapExtentId

Available since 2017.02. When the gridPlot is selected in the spatial display, the zoom extent will be automatically adapted to the specified extent. It is also possible to specify the mapExtentId for several gridPlots at once using the defaults.

timeSeriesSet

Definition of the time series set to be displayed in the selected grid plot. This can refer to one location with valuetype grid or longitudinal profile, or it can refer to a locationSet of scalars. Contourlines can only be displayed in combination with a regular grid.

When 2D or 3D data is imported using multiple domain partitions, this data can be displayed using LocationSets. All partitions have a LocationId and a LocationSet can be created for reference to all partitions. This LocationSet can be used consistently throughout the gridDisplay configuration, like when displaying:

- 2D data, e.g. water level
- 3D data, e.g. a combination of uTimeSeriesSet, VTimeSeriesSet and sigmaScaleReferenceTimeSeriesSet
- 3D data, e.g. a combination of timeSeriesSet (e.g. temperature) and sigmaScaleReferenceTimeSeriesSet

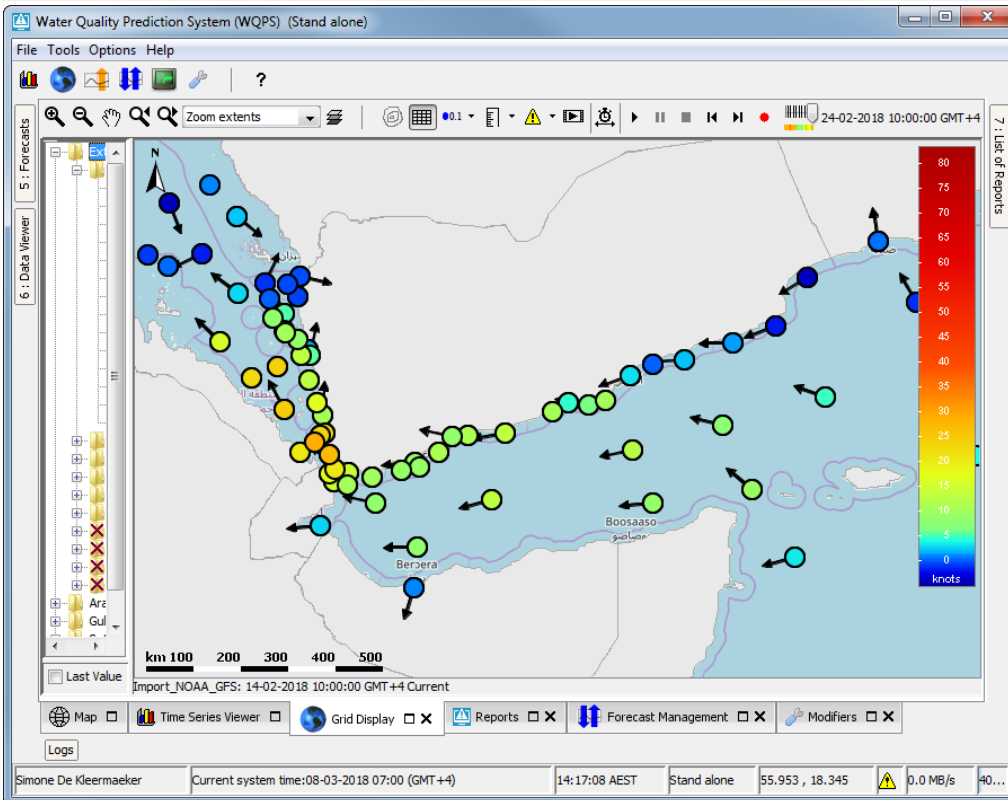
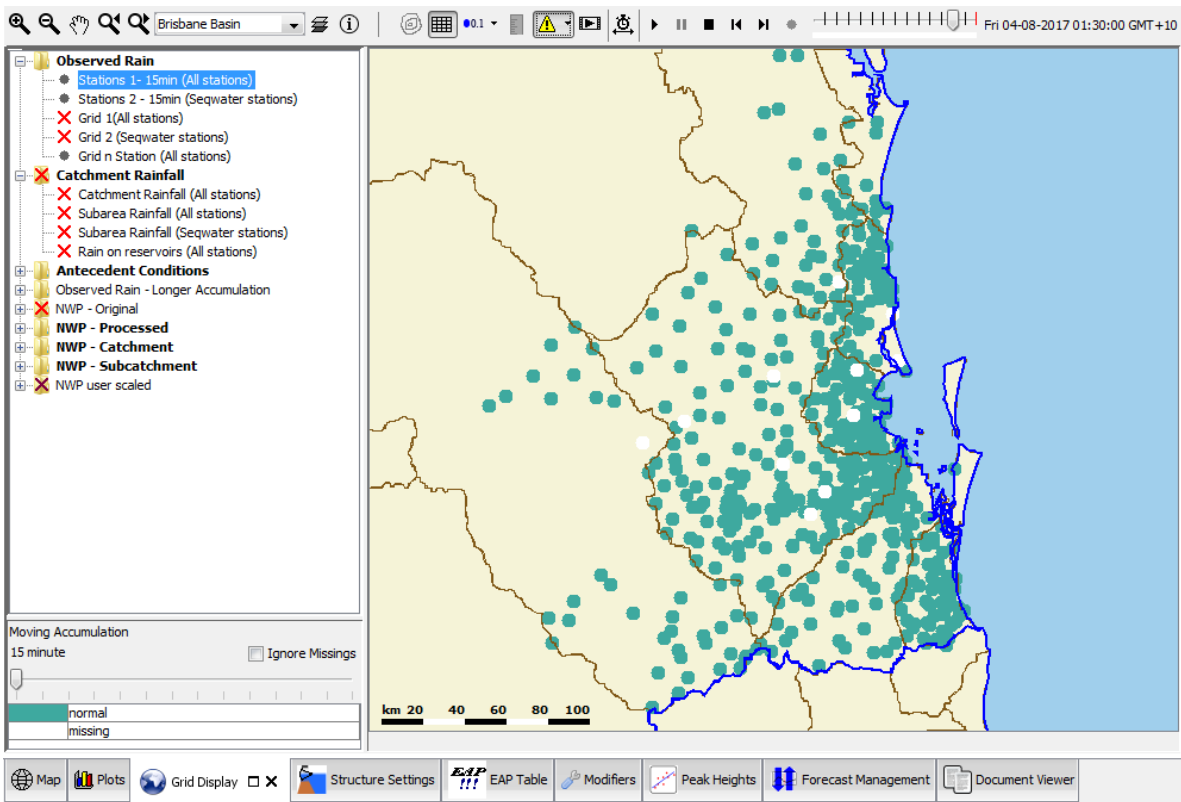
Of gridded ensemble datasets, only one ensemble member can be shown per gridplot. So this requires setting the ensembleMemberIndex of the specific TimeSeriesSet. Since 2019.02 it is possible to open a [thumbnail display](#) showing all ensemble members for the selected timestep. In this case, in the TimeSeriesSet the ensembleMemberIndexRange should be set.

dataLayer

Use this option to display multiple data grids. Each grid will be displayed in a separate data layer. The layer specified last, will be shown on top.

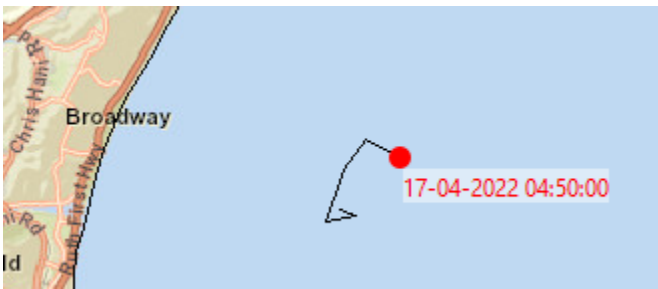
If a layer contains scalar data (possibly with a locationSet for multiple locations), this will be converted to a grid to display in that layer showing data at the specified (point) locations as follows:

- timeSeriesSet: displayed as dots (see figure)
- uTimeSeriesSet and vTimeSeriesSet: displayed as arrows
- uTimeSeriesSet and vTimeSeriesSet, as well as a timeSeriesSet for the wind velocity for the same location(s): displayed as dots and arrows combined, resembling wind vanes (see figure)



showArrowsOnLines

This element is available since 2018.01, in combination with a dataLayer containing a timeSeriesSet element. If this is set to true (false by default) and the locationSet in the timeSeriesSet contains (poly)-lines, the lines drawn in the spatial display will include directional arrows. If the time series has a positive value for a line, the arrow will point from the start of the line (first point found in the shape file) to the end of the line (second point found in the shape file). If the value is negative, the arrow is inverted. See the figures for examples.



```

<trackLayer>
  <dotFillColor>black</dotFillColor>
  <dotSize>1</dotSize>
  <onlyShowDotForCurrentTime>true</onlyShowDotForCurrentTime>
  <geoDatum>WGS_1984</geoDatum>
  <xTimeSeriesSet>
    <moduleInstanceId>ImportDrifters</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>Longitude</parameterId>
    <locationSetId>Drifters</locationSetId>
    <timeSeriesType>simulated forecasting</timeSeriesType>
    <timeStep unit="nonequidistant"/>
    <readWriteMode>read complete forecast</readWriteMode>
  </xTimeSeriesSet>
  <yTimeSeriesSet>
    <moduleInstanceId>ImportDrifters</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>Latitude</parameterId>
    <locationSetId>Drifters</locationSetId>
    <timeSeriesType>simulated forecasting</timeSeriesType>
    <timeStep unit="nonequidistant"/>
    <readWriteMode>read complete forecast</readWriteMode>
  </yTimeSeriesSet>
  <valueTimeSeriesSet>
    <moduleInstanceId>ImportDrifters</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>Age</parameterId>
    <locationSetId>Drifters</locationSetId>
    <timeSeriesType>simulated forecasting</timeSeriesType>
    <timeStep unit="nonequidistant"/>
    <readWriteMode>read complete forecast</readWriteMode>
  </valueTimeSeriesSet>
</trackLayer>

```

showThumbnailsPanel

If this is true, then the grid display shows the spatial thumbnails panel when this grid plot is selected. Otherwise the grid display will not show the spatial thumbnails panel when this grid plot is selected. Default is false. After this grid plot has been selected, the user can still switch on/off the spatial thumbnails panel manually, using the toggle button in the grid display.

Since 2017.02, if this element is never configured for any grid plot, the user settings will be used to decide whether the thumbnails panels is shown on startup, and the panel will remain shown / hidden when a different grid plot is selected. If this element is configured for any grid plot, the thumbnails panel will always be hidden on startup, and actively shown / hidden when selecting a different grid plot depending on the configuration.

Before 2017.02, the user settings were never used to reinstate the status of the thumbnails panel on startup, the panel was always hidden. When selecting grid plots, the panel was shown / hidden depending on the configuration, until the user manually switches the panel on or off. Once a manual change was made, the panel remained shown / hidden even when selecting different grid plots for the remainder of the session.

openEnsembleThumbnailsWindowWhenSelected

If this is true and the grid plot contains ensemble time series, then the [ensemble thumbnails window](#) will be opened directly after selecting the grid plot. Default is false.

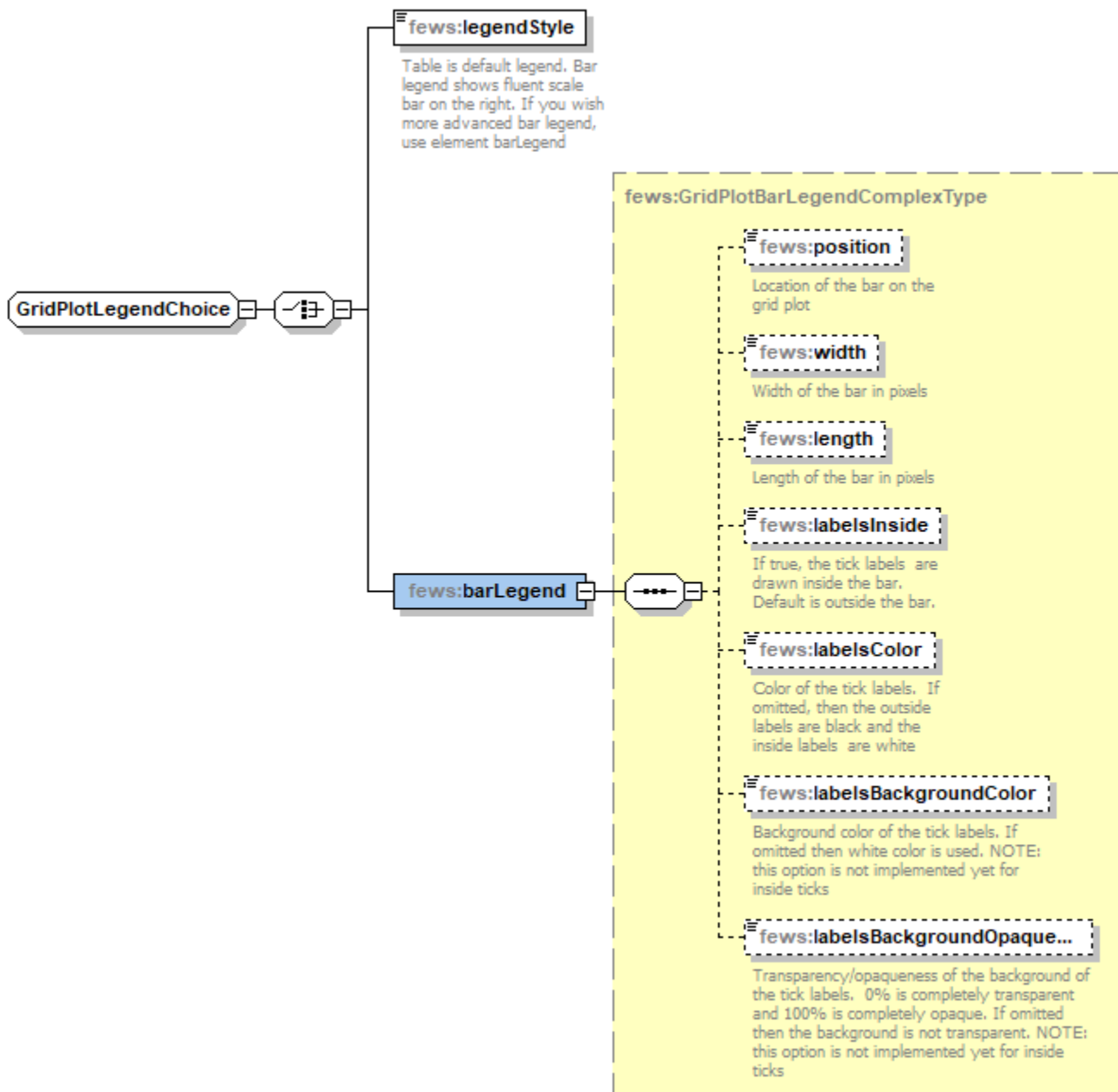
legendStyle

The legend can be shown as table legend or as bar legend.
Default legend style is table. Bar legend shows fluent scale bar on the right.

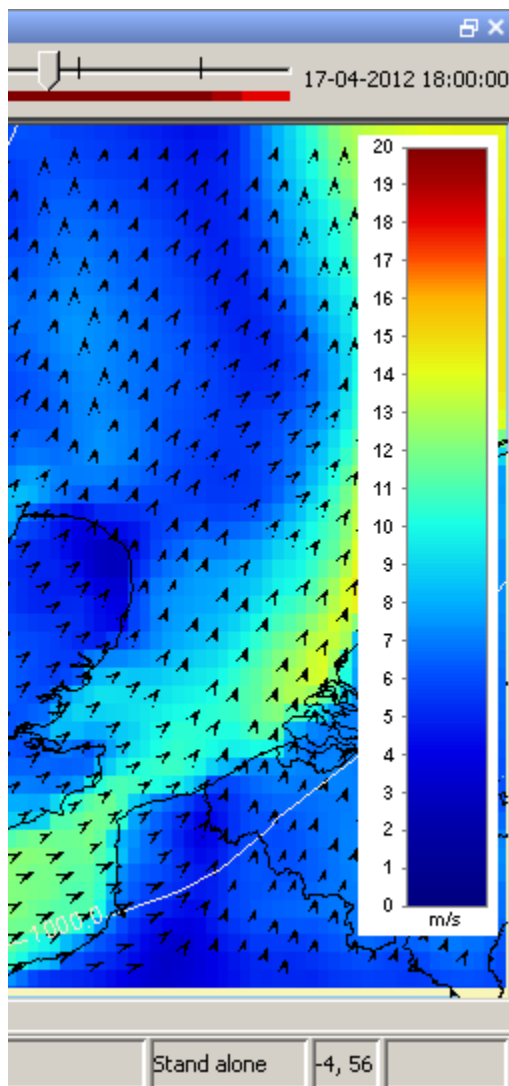
barLegend

Use this element to show a customized color bar legend.
The following legend properties can be changed:

- position: legend can be placed on the right, on the left, at the top or at the bottom. Default is on the right.
- width: width of the bar in pixels. Default is 40 pixels.
- length: length of the bar in pixels. The defaults are 400 pixels for legend on the right/left, and for the legend at the top/bottom the length equals the width of the window.
- labelsInside: if true, ticks and labels are displayed inside the legend bar. By default (false) labels and ticks are placed outside the color bar.
- labelsColor (since 2021.01): color of the tick label texts. If omitted, the outside labels are black and the inside labels are white.
- labelsBackgroundColor (since 2021.01): background color of the tick labels. If omitted, white color is used. Note that this option is not implemented for inside ticks and labels (i.e.: labelsInside=true).
- labelsBackgroundOpacityPercentage (since 2021.01): transparency/opaqueness of the background of the tick labels. 0% is completely transparent and 100% is completely opaque. If omitted then the background is not transparent. Note that this option is not implemented for inside ticks and labels (i.e.: labelsInside=true).
- logarithmic (since 2024.01, backported to 2023.01): if true the legend ticks and color breaks are spaced logarithmically.



Elements of the configuration of legend style



valueTimeSeriesSet / directionTimeSeriesSet

If a timeseries with direction data is available, this is where the TimeSeriesSet should be placed. Note: a TimeSeriesSet with velocity values is required in the ValueTimeSeriesSet.

Use classbreaks to scale the arrows, when values are too small the arrows can become invisible.

For example:



Created with:

Value/Direction with arrows

```
<dataLayer>
  <visibleInTimeSeriesDisplay>true</visibleInTimeSeriesDisplay>
  <visibleInSpatialDisplay>true</visibleInSpatialDisplay>
  <arrowDirection>from</arrowDirection>
  <multipleArrowsPerValue>false</multipleArrowsPerValue>
  <arrowMinimumPixelDistanceBetweenTwoCellCenters>2<
/arrowMinimumPixelDistanceBetweenTwoCellCenters>
  <valueTimeSeriesSet>
    <moduleInstanceId>Import_Wavenet</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>Wave.obs.total.height</parameterId>
    <locationSetId>UK_Coastal_Gauged_Wave</locationSetId>
    <timeSeriesType>external_historical</timeSeriesType>
    <timeStep unit="nonequidistant"/>
    <relativeViewPeriod unit="hour" start="-48" startOverrutable="
true" end="0" endOverrutable="false"/>

    <readWriteMode>read_only</readWriteMode>
  </valueTimeSeriesSet>
  <directionTimeSeriesSet>
    <moduleInstanceId>Import_Wavenet</moduleInstanceId>
    <valueType>scalar</valueType>
    <parameterId>Wave.obs.peak.dir</parameterId>
    <locationSetId>UK_Coastal_Gauged_Wave</locationSetId>
    <timeSeriesType>external_historical</timeSeriesType>
    <timeStep unit="nonequidistant"/>
    <relativeViewPeriod unit="hour" start="-48" startOverrutable="
true" end="0" endOverrutable="false"/>

    <readWriteMode>read_only</readWriteMode>
  </directionTimeSeriesSet>
</dataLayer>
<classBreaks>
  <lowerColor>blue</lowerColor>
  <upperColor>red</upperColor>
  <lowerSymbolSize>20</lowerSymbolSize>
  <upperSymbolSize>100</upperSymbolSize>
  <lowerValue>0</lowerValue>
  <lowerValue>5</lowerValue>
</classBreaks>
```

uTimeSeriesSet / vTimeSeriesSet

When data with velocities are available as a timeseries with the u-component and the v-component, these can be placed in the uTimeSeriesSet and vTimeSeriesSet elements. In the grid display automatically arrows will be drawn showing the direction, calculated on the v and u component. The velocity is presented in a color, defined in the classBreaks.

Double clicking in the grid display will open the u and v timeseries in the timeseries display. If one of the following is configured, they will be opened instead:

- Adding the uvAmplitudeParameterId element (since 2015.02), where an amplitude parameter can be configured, the timeseries display will combine the u and v timeseries into one amplitude timeseries ($\sqrt{u^2+v^2}$).
- Adding the uvDirectionParameterId element (since 2016.01), where a wind direction parameter can be configured, the timeseries display will combine the u and v timeseries into one wind direction timeseries. The wind direction is in degrees, where the north direction is 360 degrees, east is 90 degrees, south is 180 degrees and west is 270 degrees.

uvAmplitudeParameterId (since 2015.02)

Id of a parameter that corresponds to the amplitude of this u,v data (e.g. wind speed). If this amplitude parameter is configured and the user double clicks on a grid cell in the grid display, then the TimeSeriesDialog will only display the amplitude $\sqrt{u^2+v^2}$ of this u,v data. The displayed amplitude time series uses the configured amplitude parameter. If neither the uvAmplitudeParameterId nor the uvDirectionParameterId is configured, then the TimeSeriesDialog only displays the separate u and v time series.

uvDirectionParameterId(since 2016.01)

Id of a parameter that corresponds to the direction of this u,v data (e.g. wind direction). If this direction parameter is configured and the user double clicks on a grid cell in the grid display, then the TimeSeriesDialog will only display the direction. The displayed wind direction time series uses the configured direction parameter. If neither the uvAmplitudeParameterId nor the uvDirectionParameterId is configured, then the TimeSeriesDialog displays the separate u and v time series.

uvDirectionParameterConvention (since 2016.02)

Configure the direction of arrows drawn in the grid display. Options are:

- to, typically used when displaying flow data or
- from, typically used when displaying wind data - this is the default option, which is used when nothing is configured.

sigmaScaleReferenceTimeSeriesSet (since 2017.02)

Additional timeSeriesSet to link the sigma layers in a 3D model to indexed layers. For this you also need a locationSet linking a parentLocation to all sigma layers. Each sigma layer is linked to an index in a MapLayerFile. The parentLocation needs to be defined in Location.xml, linked to a grid definition in Grids.xml. See for a full config example [Sigmalayers - display D-FlowFM 3D results in GridDisplay](#).

zParameterId (since 2018.01)

When configured a vertical slide is displayed to interpolate between the layers. Configure only the most important layers. The sibling layers are used automatically

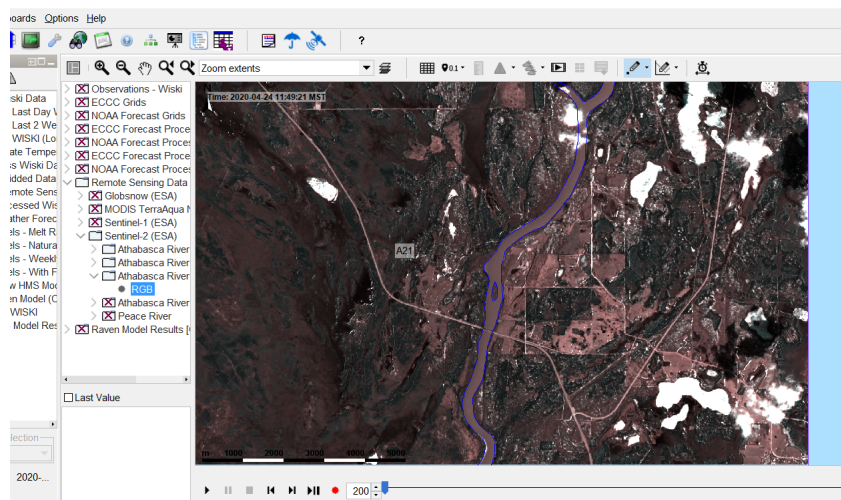
Display true color imagery

Available since: version 2019.02, patch #93400

By specifying a (3 bands) red, green and blue time series sets that provide values between 0..255 you can store and show true color satellite images as a grid time series. There is no need to configure classbreaks. You can change the brightness and contrast for each color channel separately. The incrementer changes the brightness and the multiplier the contrast of the color channel. When the incrementer + multiplier * value exceeds 255 it is set to 255, when the result is negative the color channel is set to 0. Since the 2020.02 it is possible to display a true color image and a normal grid with a legend at the same time.

True Color Imagery example 1

```
<gridPlot id="RGB">
  <dataLayer>
    <redTimeSeriesSet>
      <moduleInstanceId>InterpolateSentinel2</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>Intensity_B4</parameterId>
      <locationId>A21</locationId>
      <timeSeriesType>external historical</timeSeriesType>
      <timeStep unit="nonequidistant"/>
      <relativeViewPeriod unit="week" end="52" start="-52"/>
      <readWriteMode>read only</readWriteMode>
      <multiplier>1000</multiplier> <!--red contrast-->
      <incrementer>20</incrementer> <!--red brightness-->
    </redTimeSeriesSet>
    <greenTimeSeriesSet>
      <moduleInstanceId>InterpolateSentinel2</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>Intensity_B3</parameterId>
      <locationId>A21</locationId>
      <timeSeriesType>external historical</timeSeriesType>
      <timeStep unit="nonequidistant"/>
      <relativeViewPeriod unit="week" end="52" start="-52"/>
      <readWriteMode>read only</readWriteMode>
      <multiplier>1000</multiplier> <!--green contrast-->
      <incrementer>20</incrementer> <!--green brightness-->
    </greenTimeSeriesSet>
    <blueTimeSeriesSet>
      <moduleInstanceId>InterpolateSentinel2</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>Intensity_B2</parameterId>
      <locationId>A21</locationId>
      <timeSeriesType>external historical</timeSeriesType>
      <timeStep unit="nonequidistant"/>
      <relativeViewPeriod unit="week" end="52" start="-52"/>
      <readWriteMode>read only</readWriteMode>
      <multiplier>1000</multiplier> <!--blue contrast-->
      <incrementer>20</incrementer> <!--blue brightness-->
    </blueTimeSeriesSet>
  </dataLayer>
</gridPlot>
```



The below XML config example shows how you can make a plot where you first show the clouds and on top of that the precipitation, combined with isolines of the air pressure. The cloud coverage is a parameter ranging from 0-100%. In the example, we do only want to show clouds when there is more than 50% of cloud coverage. To prevent that the clouds become too much black or gray, we scale the range of coverage between RGB 226 to 255. To scale this to relevant RGB, the values are increased with 127.5 and multiplied by 1.275. Finally, by using the redGreenBlueVisibilityThreshold only RGB values of more than 226 are plotted.



True Color Imagery example 2: rain on top of clouds, combined with isolines

```
<gridPlot id="Harmonie_full">
  <dataFeedId>WIWB (Knmi.Harmonie)</dataFeedId>
  <dataLayer>
    <redGreenBlueOpaquenessPercentage>60</redGreenBlueOpaquenessPercentage>
    <redGreenBlueVisibilityThreshold>226</redGreenBlueVisibilityThreshold>
    <redTimeSeriesSet>
      <moduleInstanceId>ImportWIWB</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>TCC.voorsp</parameterId>
      <locationId>KNMI-HARMONIE</locationId>
      <timeSeriesType>external forecasting</timeSeriesType>
      <timeStep unit="hour"/>
      <relativeViewPeriod unit="day" start="-10" end="5"/>
      <readWriteMode>read only</readWriteMode>
      <multiplier>1.275</multiplier>
      <incrementer>127.5</incrementer>
    </redTimeSeriesSet>
    <greenTimeSeriesSet>
      <moduleInstanceId>ImportWIWB</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>TCC.voorsp</parameterId>
      <locationId>KNMI-HARMONIE</locationId>
      <timeSeriesType>external forecasting</timeSeriesType>
      <timeStep unit="hour"/>
      <relativeViewPeriod unit="day" start="-10" end="5"/>
      <readWriteMode>read only</readWriteMode>
      <multiplier>1.275</multiplier>
      <incrementer>127.5</incrementer>
    </greenTimeSeriesSet>
    <blueTimeSeriesSet>
      <moduleInstanceId>ImportWIWB</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>TCC.voorsp</parameterId>
      <locationId>KNMI-HARMONIE</locationId>
      <timeSeriesType>external forecasting</timeSeriesType>
      <timeStep unit="hour"/>
      <relativeViewPeriod unit="day" start="-10" end="5"/>
      <readWriteMode>read only</readWriteMode>
      <multiplier>1.275</multiplier>
      <incrementer>127.5</incrementer>
    </blueTimeSeriesSet>
  </dataLayer>
</gridPlot>
```

```

</dataLayer>
<dataLayer>
  <timeSeriesSet>
    <moduleInstanceId>ImportWIWB</moduleInstanceId>
    <valueType>grid</valueType>
    <parameterId>P.voorsp</parameterId>
    <locationId>KNMI-HARMONIE</locationId>
    <timeSeriesType>external forecasting</timeSeriesType>
    <timeStep unit="hour"/>
    <relativeViewPeriod unit="day" start="-10" end="5"/>
    <readWriteMode>read only</readWriteMode>
  </timeSeriesSet>
</dataLayer>
<dataLayer>
  <contourValues>
    <value>910</value>
    <value>915</value>
    <value>920</value>
    <value>925</value>
    <value>930</value>
    <value>935</value>
    <value>940</value>
    <value>945</value>
    <value>950</value>
    <value>955</value>
    <value>960</value>
    <value>965</value>
    <value>970</value>
    <value>975</value>
    <value>980</value>
    <value>985</value>
    <value>990</value>
    <value>995</value>
    <value>1000</value>
    <value>1005</value>
    <value>1010</value>
    <value>1015</value>
    <value>1020</value>
    <value>1025</value>
    <value>1030</value>
    <value>1035</value>
    <value>1040</value>
    <value>1045</value>
    <value>1050</value>
    <value>1055</value>
    <value>1060</value>
    <value>1065</value>
    <value>1070</value>
    <value>1075</value>
    <value>1080</value>
  </contourValues>
  <timeSeriesSet>
    <moduleInstanceId>ImportWIWB</moduleInstanceId>
    <valueType>grid</valueType>
    <parameterId>PPPP.voorsp</parameterId>
    <locationId>KNMI-HARMONIE</locationId>
    <timeSeriesType>external forecasting</timeSeriesType>
    <timeStep unit="hour"/>
    <relativeViewPeriod unit="day" start="-10" end="5"/>
    <readWriteMode>read only</readWriteMode>
  </timeSeriesSet>
</dataLayer>
<showThumbnailsPanel>false</showThumbnailsPanel>
<accumulationTimeSpan multiplier="6" unit="hour"/>
<accumulationTimeSpan multiplier="12" unit="hour"/>
<accumulationTimeSpan multiplier="24" unit="hour"/>
<accumulationTimeSpan multiplier="48" unit="hour"/>
<accumulationTimeSpan multiplier="120" unit="hour"/>
<contourLinesColor>black</contourLinesColor>
</gridPlot>

```

Accumulation / Moving Average slider or Last Value Checkbox

Optionally, an accumulation or moving average slider can be configured. If this slider is present the [last value checkbox](#), is shown by default, which can optionally be hidden. Only one of the following elements may be configured, although the chosen element may appear several times when configuring a slider:

- `<movingAverageTimeSpan>`: When configured a slider is displayed with the configured time spans. If a time span is selected in the slider, then the data in the grid display is aggregated (moving average) using that time span. The start of a timeSpan is exclusive and the end of a timeSpan is inclusive. Configure a very large time span (eg 100000 days) to allow the user to see the total average of the loaded time series.
- `<movingAccumulationTimeSpan>`: When configured an accumulation slider is displayed with the configured time spans. If a time span is selected in the slider, then the data in the grid display is aggregated (moving accumulation) using that time span. The start of a timeSpan is exclusive and the end of a timeSpan is inclusive. Configure a very large time span (eg 100000 days) to allow the user to see the total sum of the loaded time series.
- `<accumulationTimeSpan>`: Since 2014.02. When configured an accumulation slider is displayed with the configured time spans. When a time span is selected with the slider, the interval of the ticks in the time slider in the grid display is recalculated to the selected time step. Configure a very large time span (eg 100000 days) to allow the user to see the total sum of the loaded time series.
- `<accumulationTimeStep>`: Since 2016.02. When configured an accumulation slider is displayed with the configured time steps. When a time step is selected with the slider, the interval of the ticks in the time slider in the grid display is recalculated to the selected time step. Configure a very large time step (eg 100 years) to allow the user to see the total sum of the loaded time series.
- `<hideLastValueCheckBox>`: Since 2017.02. Default false. When set to true, the "show last value" will be hidden. Note that if an accumulation slider is configured, the checkbox will be hidden by default.

Note: When using this functionality in the Grid Display it is important to know that when the moving option is deactivated, the accumulation periods are relative to the Display time of the Grid Display. When you want to step with daily time steps at 00:00 hours through a rainfall series, first set the display time to 00:00 hours. Then accumulate the data to daily steps. The accumulated timeseries will calculate daily interval series from 00:00 to 00:00 next day.

Config example for configuring a moving accumulation time span slider:

Configuration of Moving Accumulation Slider

```
<gridPlot id="MeteoSat">
  <!-- omitted the time series configuration for clarity -->
  <movingAccumulationTimeSpan unit="hour" multiplier="6"/>
  <movingAccumulationTimeSpan unit="hour" multiplier="12"/>
  <movingAccumulationTimeSpan unit="day" multiplier="1"/>
  <movingAccumulationTimeSpan unit="day" multiplier="2"/>
  <movingAccumulationTimeSpan unit="day" multiplier="3"/>
</gridPlot>
```

verticalSliderRange

Available since: 2017.02

When a `sigmaScaleReferenceTimeSeriesSet` is configured, a vertical slider becomes automatically visible in `GridDisplay` to slide through the water column, displaying the 2D model results at an arbitrary water depth. The values displayed are dynamically interpolated between the sigma layers of the model, but only for the visible time step for performance reasons. The range of the slider is automatically adjusted to all available water depths for whole period, however you can limit this range using the `<verticalSliderRange>` option. By default 100 ticks with 10 intervals each are visible at the vertical slider. Since FEWS 2021.01 you can control the tick distance with the `interval` attribute in the `verticalSliderRange`. The direction is controlled with `verticalPositiveDirection` in the parameters.xml of the `sigmaScaleReferenceTimeSeriesSet` parameter. See for a config example [Sigmalayers - display D-FlowFM 3D results in GridDisplay](#).

Since 2022.02 ticks can be configured with irregular spacings. The following config is for 5 irregularly spaced tick marks:

```
<verticalSliderRange start="0" end="-1000" ticks="0, -1, -10, -22, -1000"></verticalSliderRange>
```



Note

The time slider ticks at the top of the gridDisplay are always colored according to the top layer, even when a different depth is selected in the vertical slider.

movieFrameDurationMillis

Option to set the speed (in milliseconds per frame) for an animation. Default value is 200 milliseconds. Available since 2019.02.

workflow

Adds a run button to the toolbar. When the user hits this button the configured workflow is started. The point or area selected at the map is added to the task run properties which can be used in the workflow. The selected area can be exported by the general adapter as a shape file. A popup is displayed before the run is started to allow the user to enter additional properties. When the sketch type is configured a popup is displayed for the wrong sketch type before the workflow is started

```
<workflow>
  <workflowId>Import</workflowId>
  <sketchType>point</sketchType>
  <properties>
    <string key="test" value="test"/>
    <string key="choice" value="A|B|C"/>
  </properties>
</workflow>
```

geoMap

Definition of the maps used as a background to the dynamic grid displayed. The layout and zoom extent are also defined in this element. More information on the configuration of the geoMap element can be found [here](#).

locationFilterAttributeId

Specify the location attributes the user can use to filter the locations/polygons/lines/grids that are visible in the plot. When specified the filter by location attribute button will be enabled. If a attribute name besides the id in the locationsSets.xml is specified this will be used in the attribute filter tree.

example

```
<locationFilterAttributeId>URBS_CATCHMENT</locationFilterAttributeId>
<locationFilterAttributeId>REGION</locationFilterAttributeId>
```

classBreaks

Definition of colors to use in displaying the dynamic grid. These are also shown in the legend on the left of the grid display (see example elsewhere on this page). Optional when default class breaks are available for this plot group and parameter group.

Classbreaks can also be configured in the [TimeSeriesDisplayConfig.xml](#) file . If this functionality is used, a reference to the classBreakId is required. When using the Deft-FEWS Web-OC, all classbreaks need to be configured this way otherwise no legend is shown in the WMS layer of the spatial display.

```
<classBreaksId>rainfall</classBreaksId>
```

description

Optional description. Since FEWS 2017.01 it is allowed to have multiple class breaks for a plot. The user can switch between the class breaks with a drop down list. This description is used in the drop down list.

`missingValueColor`

Color that is used to display missing values (NaN values) for point locations and tracks. Default is fully transparent.



Note

The `missingValueColor` configured in this `classBreaks` element is also used to display missing values (NaN values) for point locations and tracks when the display is showing threshold warning level flags.

`missingValueOpacityPercentage`

Opacity percentage that is used to display missing values (NaN values) for point locations and tracks. This overrides the opacity of the configured `missingValueColor`. If `missingValueOpacityPercentage` is not configured, then opacity of `missingValueColor` is used

`unitVisible` (available since build 18734)

When this is true the display unit for the class break values will be displayed in the legend. Default is false. The display unit can be configured in parameter group.

`rescaleAroundOrdinalValue` (available since build 17892)

Definition of the optional ordinal value that will always keep the same colour when the class break colours are rescaled in the grid display. After rescaling, the highest `lowerValue` will be changed to the maximum grid value visible in the current zoom extent and the lowest `lowerValue` will be changed to the minimum grid value visible in the current zoom extent. The `lowerValues` and the colours in between will be rearranged to fit between the minimum and maximum. Thus the colours for given values change.

If no ordinal value is specified, then the colours are just rearranged. However, if e.g. ordinal value = 0 is specified and 0 values have a white colour, then the rescaling will take this into account so that 0 values always stay coloured white. This can be used for instance when displaying temperatures, where red colours are used for positive values and blue colours are used for negative values and zero values are coloured white.

`lowerColor`

Colour definition for the colour in the legend associated with the lowest value in the range.

`upperColor`

Colour definition for the colour in the legend associated with the highest value in the range.

`lowerOpacityPercentage`

Optional definition of the opacity of the colour in the legend associated with the lowest value in the range.

`upperOpacityPercentage`

Optional definition of the opacity of the colour in the legend associated with the highest value in the range.

`lowerSymbolSize`

Optional definition of the size of symbols associated with the lowest value in the range.

`upperSymbolSize`

Optional definition of the size of symbols associated with the highest value in the range.

`lowerValue`

Definition of the value at which the colour in the grid displayed changes. The legend will be a gradual change in colours from the lowest colour to the highest colour, with the number of increments determined by the number of `lowerValue` items entered. Multiple entries may exist.

`color`

Deprecated, use `break`.

`break`

The options described above can be used for definitions of `lowerValues` that have colors that change gradually between a `lowerColor` and `upperColor`. The `break` option can be used instead for specifying a discrete `lowerValue` with an absolute color, `symbolSize` and `opacityPercentage`. In this case, set `colorSmoothingEnabled` to false. Multiple entries may exist. Labels cannot contain the ">" and "<" sign. To present these, use the `>` and `<` respectively.

`Break` also has the attribute `hideValueLabel`. Default is false, if it is set to true, labels of values that fall into this classbreak will not be displayed. See example below. This is available since 2020.01.

```

<classBreaks>
  <break color="white" label="0 mg/L" opaquenessPercentage="30" lowerValue="0"
colorSmoothingEnabled="true" hideValueLabels="true"/>
  <color color="070268" label="10 mg/L" lowerValue="10" colorSmoothingEnabled="true"
/>
</classBreaks>

```



geoMap (or background maps)

The background maps to be displayed are defined in the geoMap element. More information on the configuration of the geoMap element can be found [here](#).

compare two different simulated forecasts spatially (Since 2020.02)

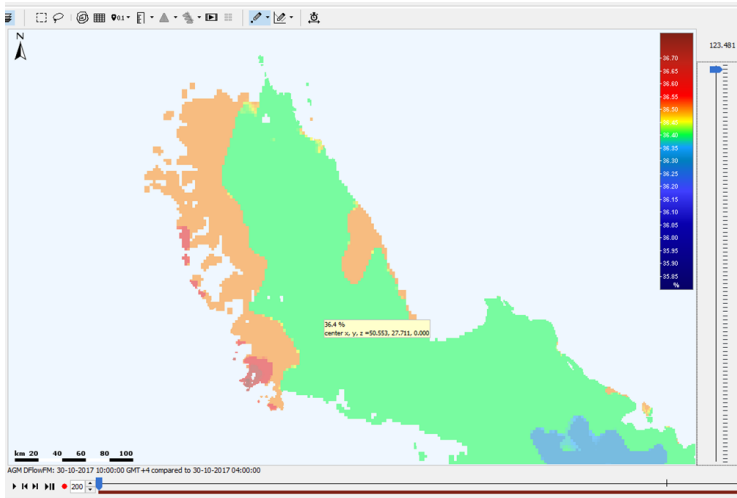
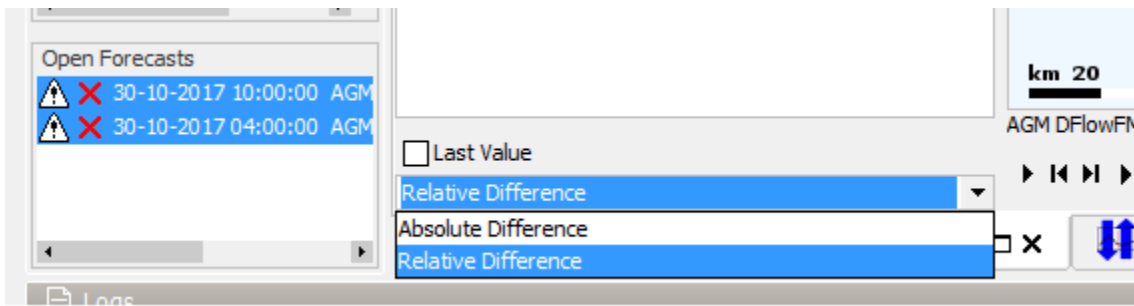
It is possible to compare results from two simulated forecasts. Open the two forecasts you want to compare in the forecast manager. When the open button is not visible make sure the forecastListSize is larger than 0.

xml

```

<panelSizes>
  <loggingPanelSize>0</loggingPanelSize>
  <listsPanelSize>22</listsPanelSize>
  <filterListSize>38</filterListSize>
  <locationListSize>45</locationListSize>
  <parameterListSize>19</parameterListSize>
  <forecastListSize>20</forecastListSize>
</panelSizes>

```



The classbreaks (legend) for the absolute and relative difference are automatically calculated based on the default classbreaks for the plot. However you can configure these two classbreaks yourself the `absoluteDifferenceClassBreaksId` and `relativeDifferenceClassBreaksId`. You can configure both or one of them.

xml

```
<defaults>
  <plotId>Meteo - Air Temperature</plotId>
  <absoluteDifferenceClassBreaksId>abs_id</absoluteDifferenceClassBreaksId>
  <relativeDifferenceClassBreaksId>relative_id</relativeDifferenceClassBreaksId>
```

Split Grids.xml file in multiple config files with gridPlotGroups (Since 2019.02)

As with many regional configuration files, the Grids.xml file can be split in multiple configuration files to make the configuration more manageable. There should be one main Grids.xml file with all defaults for parametergroups, geoMaps, etc.. The splitting supports additional files with gridPlotGroups. There is no naming convention for the additional gridPlotGroups XML files. The only restriction is that the config files use the new gridPlotGroups.xsd XML schema and are stored in the DisplayConfigFiles folder or sub-folder.

The example below shows an additional gridPlotGroups config file with only gridPlotGroups and a main Grids.xml file. The Grids.XML only requires a reference to the gridPlotGroups (`gridPlotGroupId>DFlow-FM</gridPlotGroupId>`) that are stored in the additional file; this is required to define the folder structure of the Spatial Display filters.

Config file using gridDisplayGroups.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<gridPlotGroups xmlns="http://www.wldelft.nl/fews" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.wldelft.nl/fews https://fewsdocs.deltares.nl/schemas/version1.0/gridPlotGroups.xsd">
  <gridPlotGroup id="DFlow-FM" name="DFlow-FM">
    <gridPlot id="Observed" name="Observed2">
      <timeSeriesSet>
        <moduleInstanceId>Import_Telemetry</moduleInstanceId>
        <valueType>grid</valueType>
        <parameterId>P.obs</parameterId>
        <locationId>Delft3DFM_PROI1</locationId>
        <timeSeriesType>external historical</timeSeriesType>
        <timeStep unit="hour"/>
        <relativeViewPeriod unit="hour" start="-96" end="48"/>
        <readWriteMode>read only</readWriteMode>
      </timeSeriesSet>
    </gridPlot>
    <gridPlot id="Forecast" name="Forecast">
      <timeSeriesSet>
        <moduleInstanceId>Import_Telemetry</moduleInstanceId>
        <valueType>grid</valueType>
        <parameterId>P.obs</parameterId>
        <locationId>Delft3DFM_PROI1</locationId>
        <timeSeriesType>external historical</timeSeriesType>
        <timeStep unit="hour"/>
        <relativeViewPeriod unit="hour" start="-96" end="48"/>
        <readWriteMode>read only</readWriteMode>
      </timeSeriesSet>
    </gridPlot>
  </gridPlotGroup>
</gridPlotGroups>
```

Extract of SpatialDisplay.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<gridDisplay xmlns="http://www.wldelft.nl/fews" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
  schemaLocation="http://www.wldelft.nl/fews https://fewsdocs.deltares.nl/schemas/version1.0/gridDisplay.xsd">
  .....
  <gridPlotGroupId>DFlow-FM</gridPlotGroupId>
</gridDisplay>
```

Extracting data from the Spatial Display

The user can create plots on the fly based on the data presented in the Spatial display. Depending on the data this can be:

- scalar time series
- longitudinal profile
- 2D longitudinal profile
- data for an area from a grid time series
- data for multiple plots at once
- save and loading sketches

This all explained in the user Delft-FEWS User Guide: [05 Spatial Display](#)

Configuration (Examples)

The following example shows how to configure a Meteosat image as grayScaleImage in the Grid display.

Extract of SpatialDisplay.xml

```
<gridPlotGroup id="Meteosat Images">
  <gridPlot id="MeteoSat">
    <timeSeriesSet>
      <moduleInstanceId>ImportMeteosat</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>image</parameterId>
      <locationId>meteosat</locationId>
      <timeSeriesType>external historical</timeSeriesType>
      <timeStep unit="minute" multiplier="15"/>
      <relativeViewPeriod unit="hour" start="-12" end="36"/>
      <readWriteMode>read only</readWriteMode>
    </timeSeriesSet>
    <classBreaks>
      <lowerColor>black</lowerColor>
      <upperColor>white</upperColor>
      <lowerValue>0</lowerValue>
      <lowerValue>8</lowerValue>
      <lowerValue>16</lowerValue>
      <lowerValue>24</lowerValue>
      <lowerValue>32</lowerValue>
      <lowerValue>40</lowerValue>
      <lowerValue>48</lowerValue>
      <lowerValue>56</lowerValue>
      <lowerValue>64</lowerValue>
      <lowerValue>72</lowerValue>
      <lowerValue>80</lowerValue>
      <lowerValue>88</lowerValue>
      <lowerValue>96</lowerValue>
      <lowerValue>104</lowerValue>
      <lowerValue>112</lowerValue>
      <lowerValue>120</lowerValue>
      <lowerValue>128</lowerValue>
      <lowerValue>136</lowerValue>
      <lowerValue>144</lowerValue>
      <lowerValue>152</lowerValue>
      <lowerValue>160</lowerValue>
      <lowerValue>168</lowerValue>
      <lowerValue>176</lowerValue>
      <lowerValue>184</lowerValue>
      <lowerValue>192</lowerValue>
      <lowerValue>200</lowerValue>
      <lowerValue>208</lowerValue>
      <lowerValue>216</lowerValue>
      <lowerValue>224</lowerValue>
      <lowerValue>232</lowerValue>
      <lowerColor>orange</lowerColor>
      <upperColor>red</upperColor>
      <lowerValue>240</lowerValue>
      <lowerValue>248</lowerValue>
      <lowerValue>255</lowerValue>
    </classBreaks>
  </gridPlot>
</gridPlotGroup>
```

3D data display (example)

The following example shows how to configure 3D data originating from several computational domains in the Grid display.

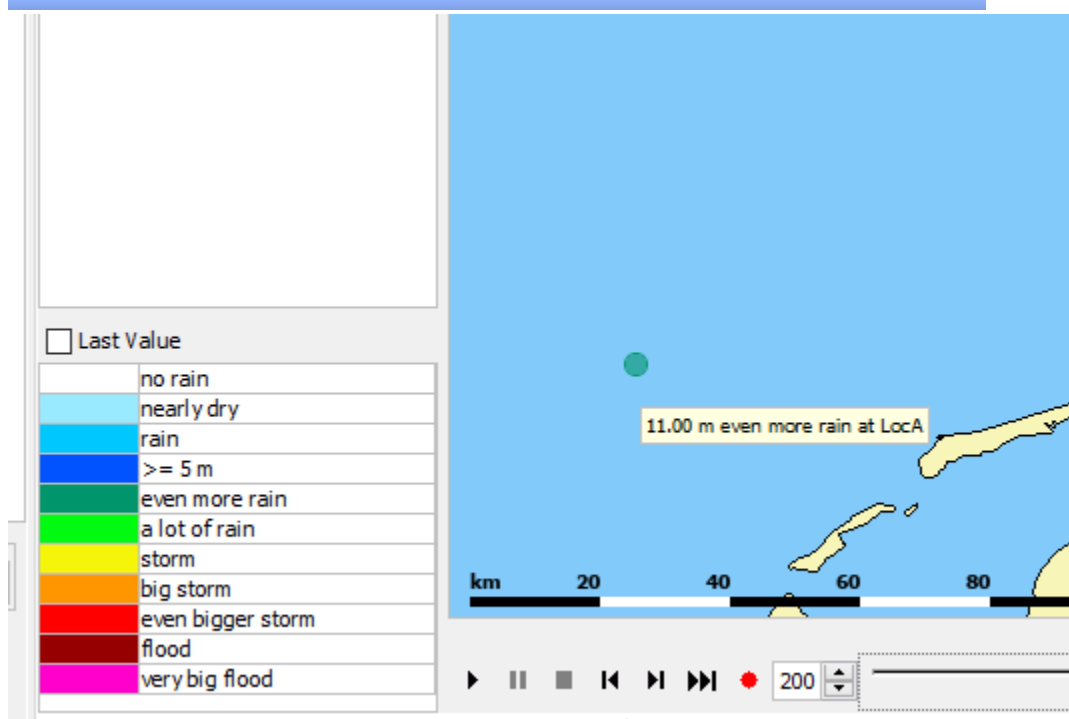
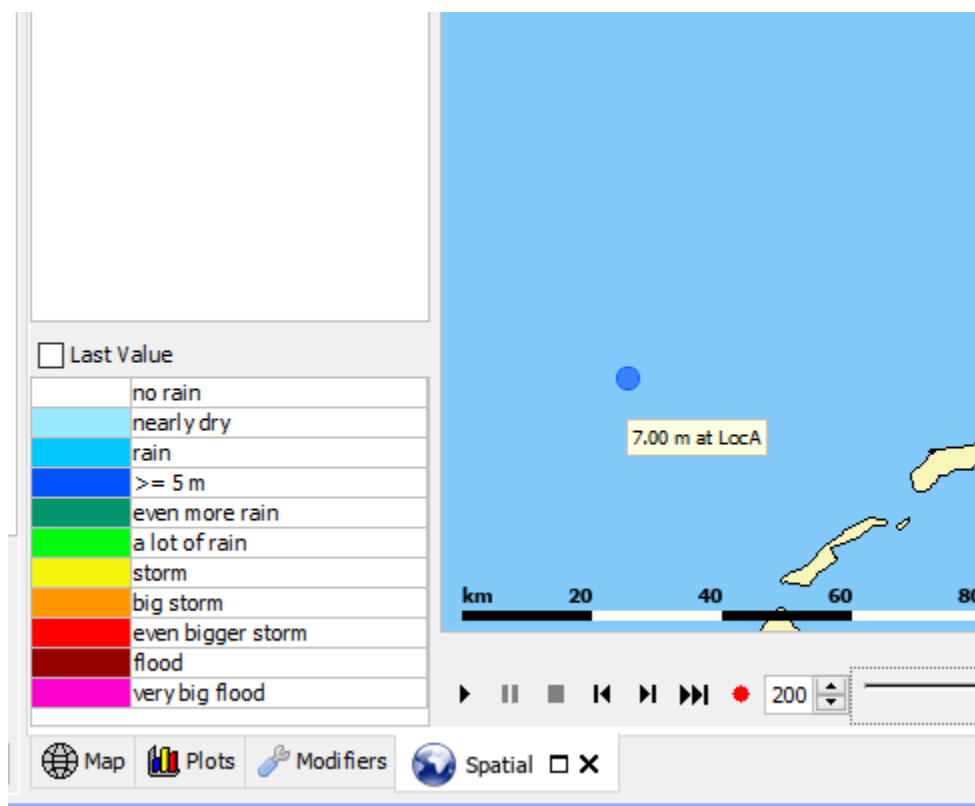
3D data example for SpatialDisplay.xml

```
<gridPlot id="3D_data" name="3D data display">
  <dataLayer>
    <arrowColor>white</arrowColor>
    <arrowSymbol>flow</arrowSymbol>
    <multipleArrowsPerValue>false</multipleArrowsPerValue>
    <uTimeSeriesSet>
      <moduleInstanceId>DFlowFM_FC</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>C.simulated.u</parameterId>
      <locationSetId>DFlowFM_0_####</locationSetId>
      <timeSeriesType>simulated forecasting</timeSeriesType>
      <timeStep unit="nonequidistant"/>
      <readWriteMode>read complete forecast</readWriteMode>
    </uTimeSeriesSet>
    <vTimeSeriesSet>
      <moduleInstanceId>DFlowFM_FC</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>C.simulated.v</parameterId>
      <locationSetId>DFlowFM_0_####</locationSetId>
      <timeSeriesType>simulated forecasting</timeSeriesType>
      <timeStep unit="nonequidistant"/>
      <readWriteMode>read complete forecast</readWriteMode>
    </vTimeSeriesSet>
    <uvAmplitudeParameterId>C.simulated.speed</uvAmplitudeParameterId>
    <uvDirectionParameterId>C.simulated.dir</uvDirectionParameterId>
    <sigmaScaleReferenceTimeSeriesSet>
      <moduleInstanceId>DFlowFM_FC</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>WD.simulated</parameterId>
      <locationSetId>DFlowFM.merged</locationSetId>
      <timeSeriesType>simulated forecasting</timeSeriesType>
      <timeStep unit="nonequidistant"/>
      <readWriteMode>read complete forecast</readWriteMode>
    </sigmaScaleReferenceTimeSeriesSet>
  </dataLayer>

  <barLegend>
    <position>right</position>
    <width>50</width>
    <length>400</length>
    <labelsInside>true</labelsInside>
  </barLegend>
  <classBreaksId>Class.Currents</classBreaksId>
  <contourLinesColor>antique white</contourLinesColor>
</gridPlot>
```

ClassBreak labels

If label is configured in a classbreak, the legend will show the label in the table instead of the values. When a mouse is hovering over a point, the tooltip will show the configured value if there is no label, and the label plus the value if there is a label (since 2018.02.). Notice in the example below that one of the breaks has no label.



```

<gridPlotGroup id="PlotGroup1">
  <gridPlot id="PlotGroup1 Plot1">
    <dataLayer>
      <timeSeriesSet>
        <moduleInstanceId>Import</moduleInstanceId>
        <valueType>scalar</valueType>
        <parameterId>H.m</parameterId>
        <locationId>LocA</locationId>
        <timeSeriesType>external historical</timeSeriesType>
        <timeStep unit="hour"/>
        <relativeViewPeriod unit="hour" start="-24" end="0"/>
        <readWriteMode>read only</readWriteMode>
      </timeSeriesSet>
    </dataLayer>
    <classBreaks>
      <unitVisible>true</unitVisible>
      <break color="FFFFFF" lowerValue="0" label="no rain" opaquenessPercentage="35"/>
      <break color="99E9FF" lowerValue="0.2" label="nearly dry" opaquenessPercentage="35"/>
      <break color="00C8FF" lowerValue="1" label="rain" opaquenessPercentage="60"/>
      <break color="0053FF" lowerValue="5" opaquenessPercentage="60"/>
      <break color="00956A" lowerValue="10" label="even more rain" opaquenessPercentage="60"/>
      <break color="00FA10" lowerValue="25" label="a lot of rain" opaquenessPercentage="60"/>
      <break color="F5F50A" lowerValue="50" label="storm" opaquenessPercentage="60"/>
      <break color="FF9600" lowerValue="75" label="big storm" opaquenessPercentage="60"/>
      <break color="FF0000" lowerValue="100" label="even bigger storm" opaquenessPercentage="80"/>
      <break color="960000" lowerValue="200" label="flood" opaquenessPercentage="80"/>
      <break color="FF00CC" lowerValue="400" label="very big flood" opaquenessPercentage="80"/>
    </classBreaks>
  </gridPlot>
</gridPlotGroup>

```

Show last import time for external historical data

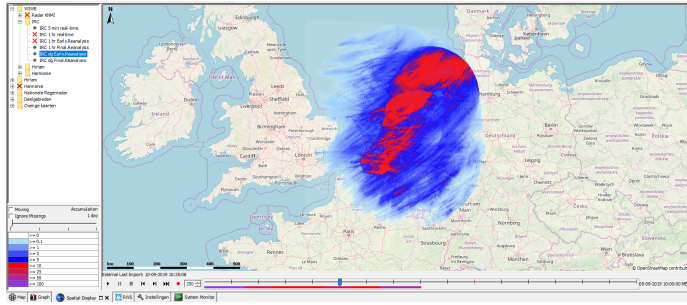
This option is supported with online version only. The option dataFeedId needs to be configured per gridPlot. The value of the dataFeedId must be the same value that can be seen in System Monitor table, tab Import Status. The id should be the one that corresponds with the module instance id of the time series.

```

<gridPlotGroup id="Radar KNMI">
  <gridPlot id="TF0005_R" name="Uncorrected 5 min">
    <dataFeedId>ATIS</dataFeedId>
    <timeSeriesSet>
      <moduleInstanceId>ImportWIWB</moduleInstanceId>
      <valueType>grid</valueType>
      <parameterId>P.radar</parameterId>
      <qualifierId>realtime</qualifierId>
      <locationId>KNMI-RADAR1km</locationId>
      <timeSeriesType>external historical</timeSeriesType>
      <timeStep unit="minute" multiplier="5"/>
      <relativeViewPeriod unit="day" start="-10" end="5"/>
      <readWriteMode>read only</readWriteMode>
    </timeSeriesSet>
    <accumulationTimeSpan multiplier="1" unit="hour"/>
    <accumulationTimeSpan multiplier="6" unit="hour"/>
    <accumulationTimeSpan multiplier="12" unit="hour"/>
    <classBreaks>
      <break lowerValue="0" color="white" label="0 mm/hr" opaquenessPercentage="25"/>
      <break lowerValue="0.008" color="light blue" label="0.1" opaquenessPercentage="75"/>
      <break lowerValue="0.083" color="8CAAFF" label="1" opaquenessPercentage="80"/>
      <break lowerValue="0.167" color="636DFF" label="2" opaquenessPercentage="85"/>
    </classBreaks>
    <contourLinesColor>gray</contourLinesColor>
  </gridPlot>
</gridPlotGroup>Run workflow from spatial display

```

If dataFeedId is configured, this is how the time will appear:



DisplayGroupPlotID

When selecting a location in the spatial Display the displayGroup for this location will be opened in the TimeSeriesDisplay. Available for 2019.01 and later.

You have to edit 2 files to get this working. The SpatialDisplay.xml and the displayGroups.xml.

Adjustments to SpatialDisplay.xml

```
<gridPlotGroup id="Coastal" name="Coastal">
  <gridPlotGroup id="coastal_obs" name="Observations">
    <gridPlot id="coastal_obs_discrete_level_rf" name="Water level (rise or fall)"
  >
    <displayGroupPlotId>TidalGauges_Centre_CMRENS</displayGroupPlotId>
    <dataLayer>
      <visibleInTimeSeriesDisplay>true</visibleInTimeSeriesDisplay>
```

So the **<displayGroupPlotId>** in the code example above should have a valid reference to a plotId in the displayGroups.xml, in this case: "TidalGauges_Centre_CMRENS"

Adjustments to displayGroups.xml

```
<plot id="TidalGauges_Centre_CMRENS">
  <subplot>
    <!--total water level-->
    <line>
      <!--obs-->
      <color>blue</color>
      <lineStyle>solid;thick</lineStyle>
      <timeSeriesSet>
        <moduleInstanceId>Import_Telemetry</moduleInstanceId>
        <valueType>scalar</valueType>
        <parameterId>H.obs</parameterId>
        <locationSetId>UK_ASTROSUM</locationSetId>
        <timeSeriesType>external historical</timeSeriesType>
        <timeStep unit="minute" multiplier="15"/>
        <relativeViewPeriod unit="hour" start="-48" end="120"/>
        <readWriteMode>read only</readWriteMode>
      </timeSeriesSet>
    </line>
```

Logo at map

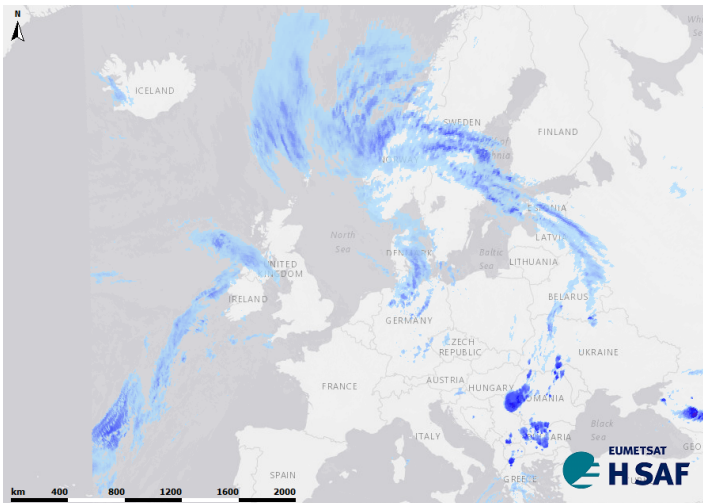
Some satellite products require to be shown with a logo, where the imageFile with the logo should be located in the MaplayerFiles directory.

The logo file can be place at the next positions: topLeft, bottomLeft, topRight or bottomRight.

```

<gridPlot id="Precipitation">
  <timeSeriesSet>
    <moduleInstanceId>Import_NWP</moduleInstanceId>
    <valueType>grid</valueType>
    <parameterId>P.fc</parameterId>
    <qualifierId>aladin_hr</qualifierId>
    <locationId>NWP_ALADIN_HR</locationId>
    <timeSeriesType>external forecasting</timeSeriesType>
    <timeStep unit="hour"/>
    <relativeViewPeriod unit="day" start="-5" end="10"/>
    <readWriteMode>read only</readWriteMode>
  </timeSeriesSet>
  <logo>
    <imageFile>logo.png</imageFile>
    <position>topRight</position>
  </logo>
  <classBreaksId>Precipitation_1h</classBreaksId>
  <contourLinesColor>gray</contourLinesColor>
</gridPlot>

```



Showing T0 or last import time underneath plot

work in progress