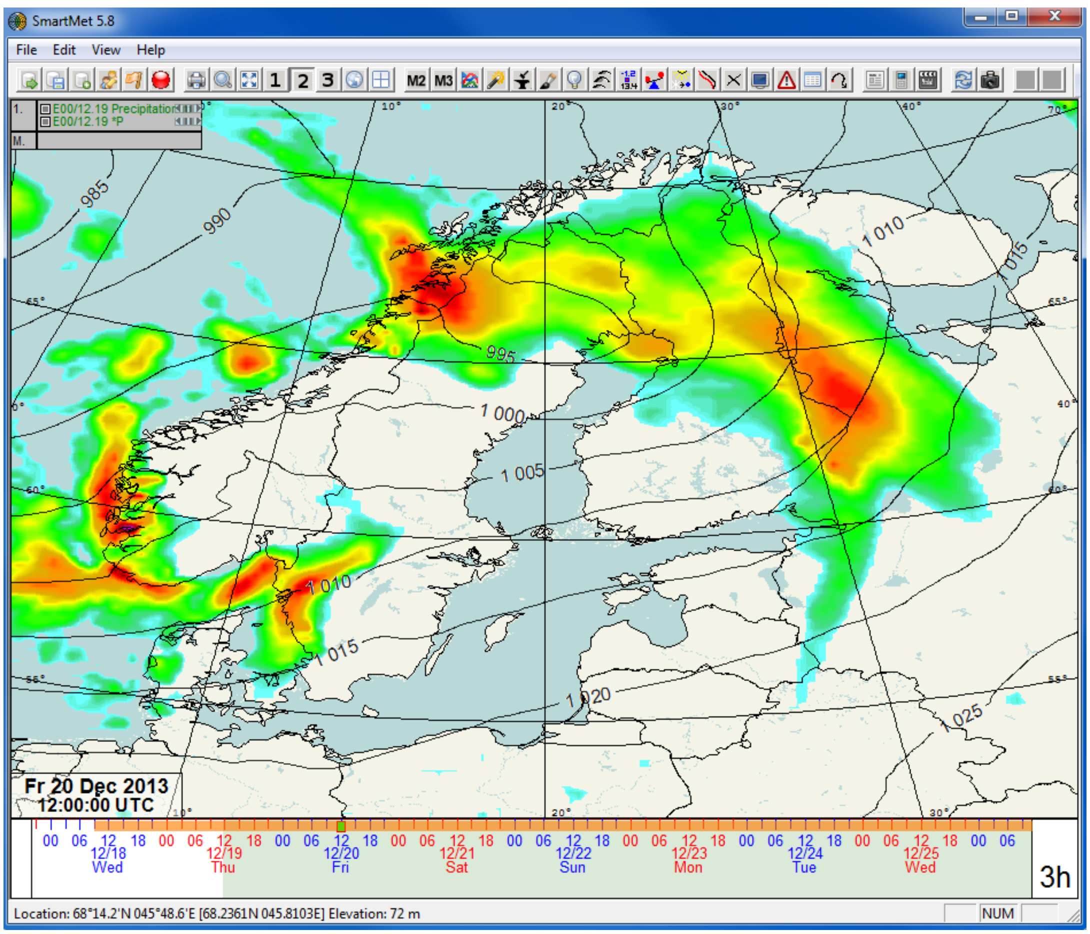
SmartMet – Tool for Visualizing and Editing Meteorological Data

User's Guide 5.1

Part 2 editing of data



1. The basic consept of editing

The editing of data means that forecasters may edit or modify model data in SmartMet with different tools. The idea is to “fix the errors” of model data so that the data is in line with the forecast that forecasters do. Once the data is edited it is sent to database so that products (i.e. diferrent kind of forecasts) may be produced from that corrected and better forecast data.

1.1 Loading data

Every time when editing data there obviously need to be some data that is edited so the first thing that forecaster should do beore editing is to select the data that is edited. Data is loaded from **data loading dialog** (Figure 2) which is opened by pressing the **load button** (Figure 1).

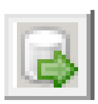


Figure 1 Data loading button.

For data to be edited you are able to choose only one model or blend modles together or blend from one model to another in respect of time. There are at maximum four models or data producers in the data loading dialog (Note that the amount of models depends on the availability of data in each country). The models may be selected from priorizing selections. This means that the model that is chosen to be priorized first will be loaded to the data that is edited with all timesteps that the model has. If the model’s time period is as long as or longer than the editable time period only that model is chosen to be the data that is edited. If the time interval is shorter than the editable time period the rest of the the editable time period is now covered with the model data that is priorized with number two. The priorization number three works respectively.

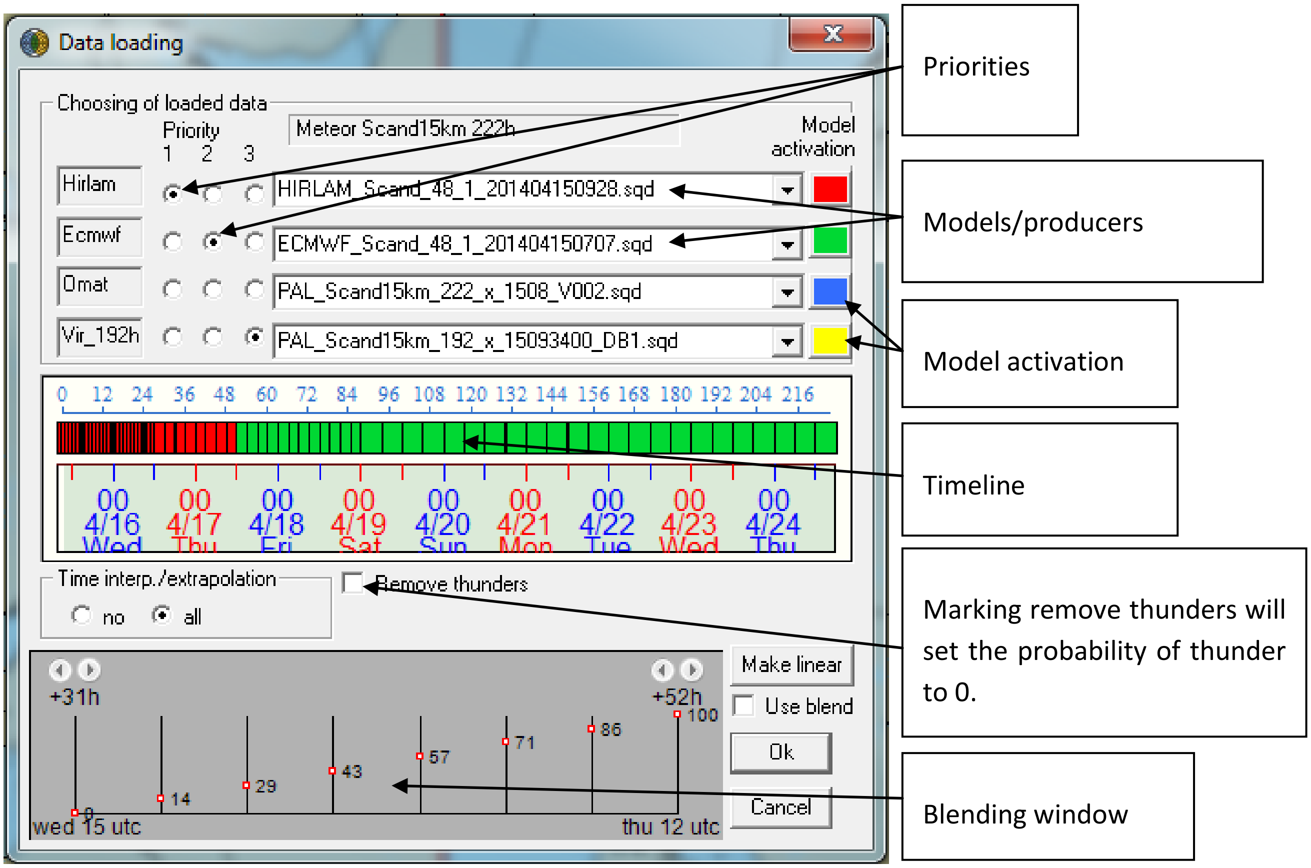


Figure 2 Data loading dialog. In this figure HIRLAM model would be loaded for the first approximately 50 hours and after that the data is ECMWF model.

On the right side of the model name you may see a colour for each model or data producer and when selecting priorizations the data loading timeline is coloured with corresponding colour. You are also able to select the models to each timestep separately. From the colour boxes, next to the model names, you are able to select the model by clicking the corresponding model activation button (i.e. corresponding colour box). Now you can paint the model for each timestep in the loading data dialog's timeline by clicking left mouse button and keeping it pressed down. All of the timesteps should have a colour other than gray before you press accept button because the gray colour means that the timestep is missing data. If data includes missing data timesteps a warning text is shown (Figure 3) and SmartMet will interpolate the missing timesteps after accepting the message.

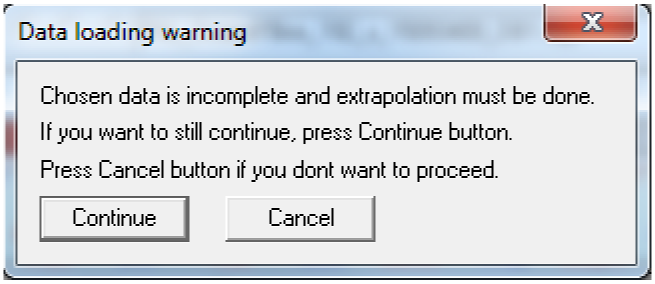


Figure 3 Warning message for missing data

1.1.1 Blending models

If you have chosen two or more models as a data to be edited you are able to blend the data so that the change from model to another won't make any discontinuations for the data. For the blending time period SmartMet will calculate weighted averages for all the parameters from the models to be blended. The blending is defined in data loading dialog's blending window (Figure 4). To blend the data you should first mark the **use blend** option. Once it is marked an orange bar appears on the timeline in the data loading dialog indicating the time period where the blending is made. The time period may be adjusted from the arrow buttons in the blending window; from the left hand side you adjust the starting point and from the right hand side the ending point. The blending is by default linear. This means that for all parameters the blending is calculated so that at the starting time of the blending the weight of the first priorized model is 100% and at the ending time the weight of the second priorized model is 100% and between them the weights chages linearily. The weights of the second priorized model are shown as percentage numbers in the blending window. From the red circles next to the numbers you are able to adjust the weights for each timestep by left clicking them if you want to make the blending nonlinear. From the **Make linear** button the blending may be restored back to linear.

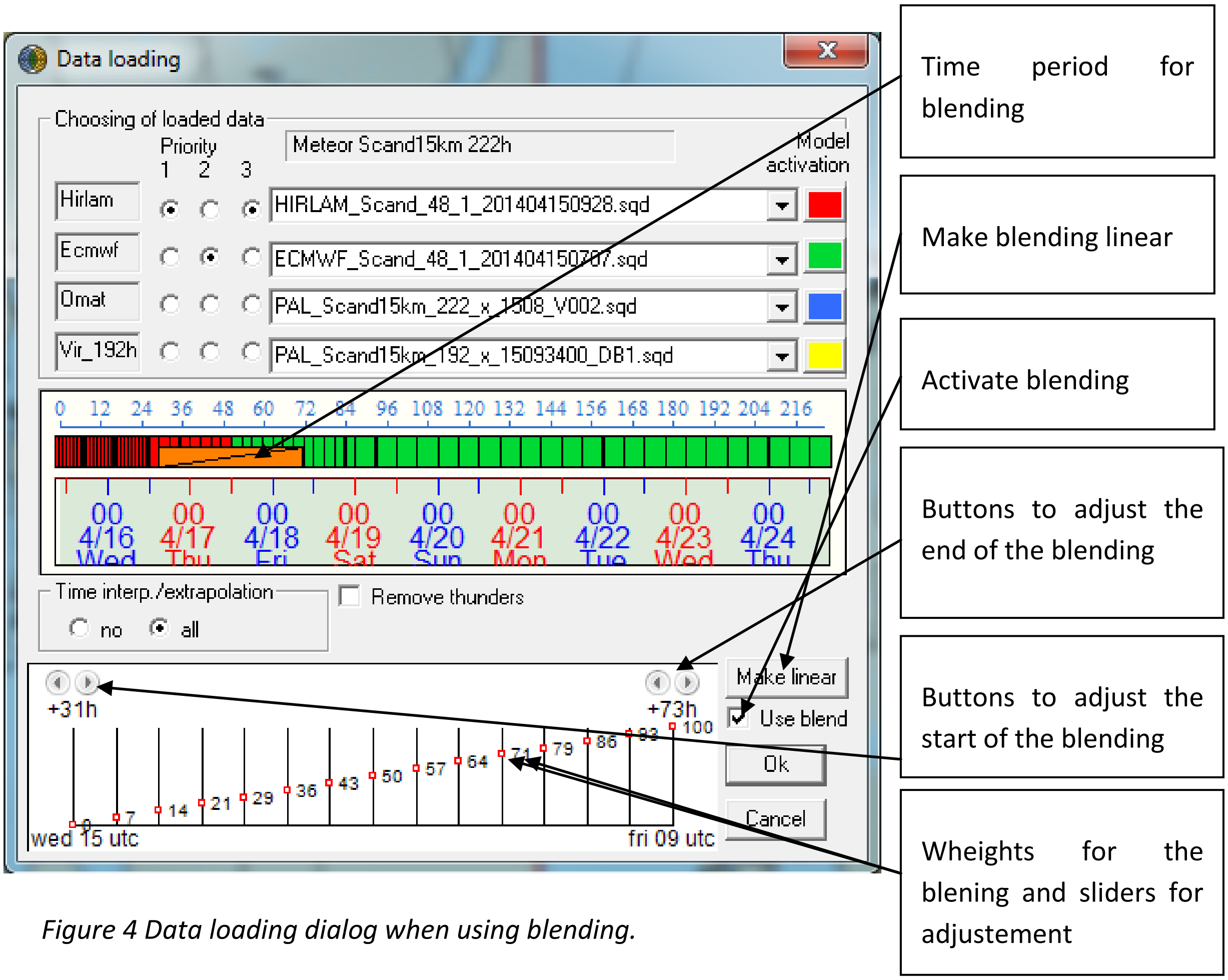


Figure 4 Data loading dialog when using blending.

1.1.2 Editable timesteps

The length of editable data varies from country to another depeding on the needs in each location. And so does the editable timesteps (i.e. the timesteps in the data that are edited). Quite common setup is that the nearest few to maybe 24 hours are edited with one hour interval and after that the data is edited with three hour interval or even six. The editable timesteps are visible in the data loading dialog (Figure 2) in the timeline (black lines between the forecast hour and the time line). Knowing what timesteps are editable is important because the data can't be edited in the timesteps that are uneditable and the tools that are explained further in this manual won't work in those timesteps. The easiest way of differentiating the the editable and uneditable timesteps when doing the editing is that there isn't a couloured outer border in editable timesteps and there is an yellow outer border in uneditable timesteps in the main map view of SmartMet (Figure 5). SmartMet will automatically interpolate the data between editable timesteps when changes are made.

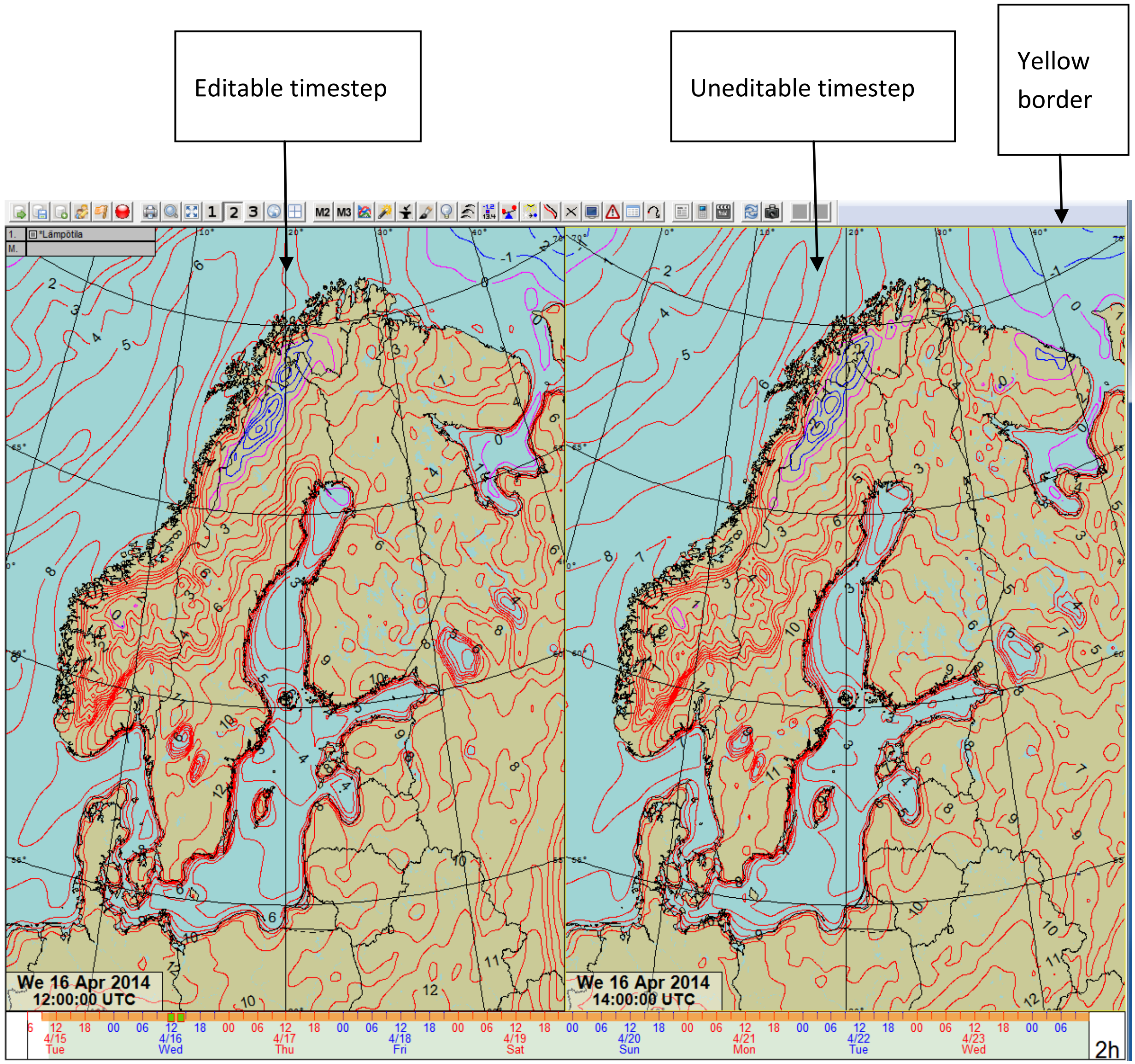


Figure 5 Editable an uneditable timesteps in SmartMet

1.1.3 Official data and work data

The last two producers in the data loading dialog is wok data and official data. Official data is the latest data that is sent to database. Work data is the latest work data that is saved to computers local hard drive (to see in detail the difference of saving data and sending data to database see next chapter). By default SmartMet saves automatically work data in every with one minute. So if SmartMet for some reason crashes (very unlikely!) you are able to get the data that you were editing by selecting the work data.

1.2 Saving data and sending data to database

Saving the data is made from the **save data button**. This means that the data you are editing is saved to the local hard drive on the computer but it isn't yet send to database. By default SmartMet also saves automatically a work file every minute. Sending to database is made from the **To database button**. When it is pressed the SmartMet asks whether you really want to export the data to the database. Accepting is made by pressing the OK button. A file that is exported to the database is also automatically saved to the work directory.

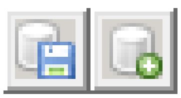


Figure 6 Save data and to database buttons.

1.2.1 Data validation

When the data is sent to data base SmartMet also makes validation for the data if the data validation is marked from the **data validation dialog** (Figure 7). The data validation dialog is opened by clicking from Edit menu settings and there the option Data validation. From Data validation you are able to adjust in addition wether the validations are made when sending data to database are the validations made automatically and is the added precipitation (made during the editing) large scale type precipitation. The used validations may be marked too. There are two different validations. From the first you are able to adjust the (surface) temperature thresholds for precipitation type (snow and water). Between the snow and water limits the precipitation type is sleet. The second validation is to check that the temperature should always be grater than the dew point (because if dew point is higher than temperature the relative humidity would be over 100%). If that option is marked and SmartMet finds data points where dew point is higher tan temperature the dew point in those points will be set to same as temperature. The data validation may also be executed from the data validation button in the main windows toolbar (Figure 7).

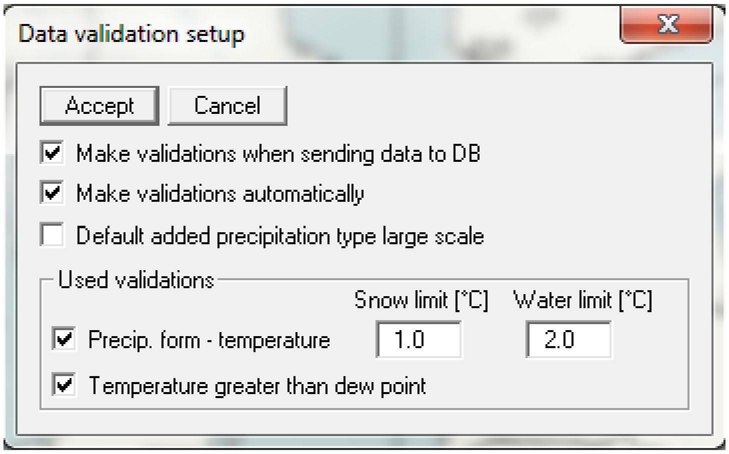


Figure 7 Data validation dialog and data validation button.

1.3 Help data

With SmartMet system it is also possible that more than one foecaster is doing the editing with different SmartMets simultaneously. This is done by using help editor mode. There always has to be one who is using the SmartMet as a “main” editor and the rest should use it as help editors (If everyone would use it simultaneously main editors their data would override eachothers). The idea is that help editors may edit for example only one parameter and the main editor then collects them and sends the whole data to database. The help editor mode is activated from **help editor** button (Figure 8).

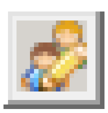


Figure 8 Help editor button

When help editor mode is activated and the **send to database** button is pressed the data won’t go in to database as it would go in “normal” editing mode. Instead of a help data file is created and now the “main” editor may select the wanted parameters from that file. This is done with the smarttool (see section 9). For example getting the temperature from help data file is expressed in smarttool language as follows.

T=T_HELP

2. BrushTool 

With the help of the BrushTool you can make changes to data. You can, for example, use the BrushTool to add clouds at some point or remove them at another. Generally, changes made with the BrushTool have to be added to all the necessary points of time. For this reason, the brush is more of a finishing tool and should not be used for drawing all the maps.

You can use the BrushTool by pressing the corresponding button in the toolbar. The button is pressed down for as long as the BrushTool is in use. During this time, the cursor is not an arrow but a cross-hairs cursor and the BrushTool dialog (Figure 9) remains open on the screen. When you want to stop using the BrushTool, you need to press the brush button in the toolbar again. The button is no longer pressed down and the cursor changes back to a standard arrow. You can also stop using the BrushTool by pressing the button in the upper right-hand corner of the BrushTool dialog.

The BrushTool dialog contains different options for the use of the brush. If **No constraints** is selected in the Constraint setting of the dialog (Figure 9), the effect of the change made by the brush to the data is of the size entered into the Brush value field. For continuous data, the value that equals the effect of the change made by the brush is entered into the field (Figure 9). For discontinuous data, the value is chosen from the list (Figure 10). The brush parameter and unit used can be seen on the right side of the field or list.

When painting with the brush you can decrease the values by pressing the left mouse button and dragging the mouse into the desired area. Values can be increased similarly with the right mouse button. When editing discontinuous data, the BrushTool adds the value chosen from the list to the desired position on the map when you click either the right or left mouse button. In Figure 11, the intensity of rain has been modified with the BrushTool.

By default, the brush has no constraints. If you want, you can set a limit, below or above which the modified value cannot go. If you want to add rain areas with the BrushTool, but do not want the amount of rainfall to be over 3.5mm, you can select the constraint **Not over** and enter the value 3.5 into the field. This way, the rainfall will not exceed 3.5. Note: If some value has already exceeded 3.5, it is changed to 3.5.

If the constraint is set to **Set value**, the brush changes the value of the data directly to the value of the constraint (not the brush value!). The brush value only tells how significant the change is, it is not an absolute value.

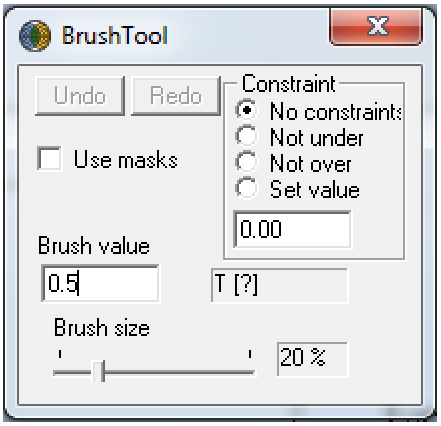


Figure 9 Settings of the BrushTool dialog.

The chosen parameter is the active parameter of the active row. The size of the brush can be adjusted with the slider between 0-100% of the size of one field of the grid. In addition, you can choose whether masks affect the brush (slows down drawing).

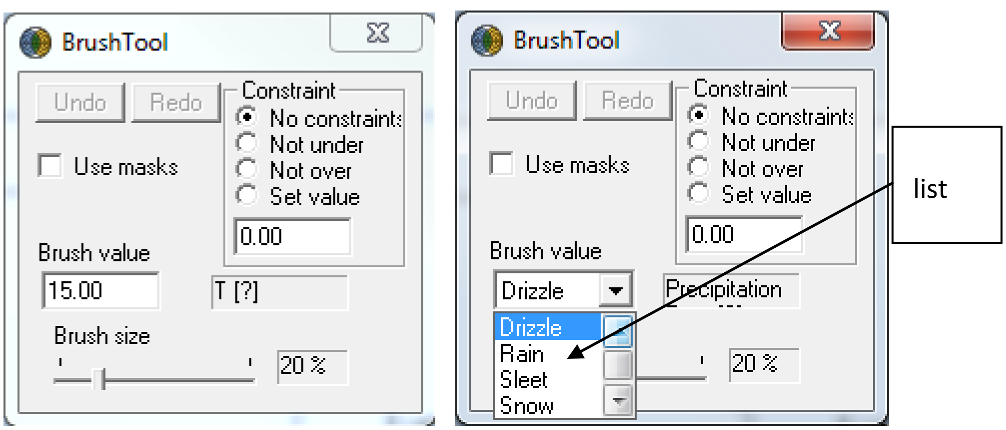


Figure 10 In the BrushTool dialog, a value for continuous data is entered into the field and a value for discontinuous data is chosen from the list.

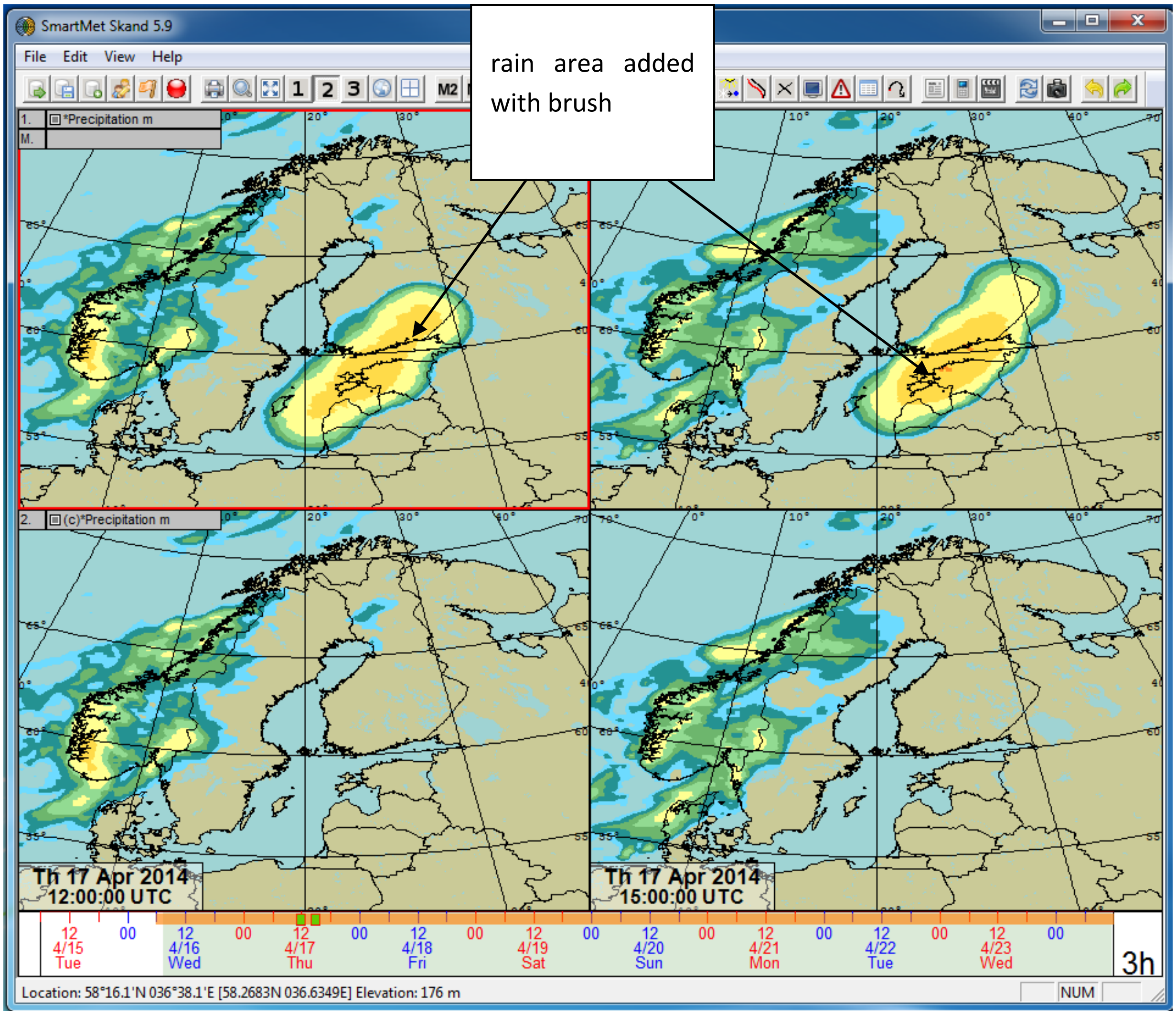


Figure 11 In the upper row the rain intensity was modified with a brush. The row below contains an image with reference data.

3. Undo/redo

With the help of the **Undo/redo** function you can discard unwanted changes made to the data (and redo them, if you undo them by mistake). Each tool that modifies data stores an image of each status and this image can be restored, if needed.

The number of consecutive Undo/redo functions can be set in the **Edit** – (Options) Settings dialog. Here it should be noted that the undo/redo function takes up a lot of memory. Each possible undo/redo step takes up the entire space of the edited data. If the data to be edited is small, it is possible to set more undo/redo steps, but if the data is large, the number of steps should only be 2 or 3. When using really large data (or using the editor just for viewing), the number of steps should be set to 0, which disables the undo/redo function.

4. Data selection

You can choose certain modifiable grid points from the editable data. Selected grid points are marked with small circles around them. Timeserial modifications (See section 7.1.2) are applied only to the selected grid points. The data filter tool can be set to have modifications be applied only to the selected points. You can select grid points by clicking the map view, when a parameter of editable data is visible. When you click the view, the new selection replaces the old one. By **CTRL**-clicking you can add new selections to the old ones. This way, you can select all the grid points of a certain area. By **SHIFT**-clicking you can remove points from the selection.

4.1 Selection tool 

With the selection tool (fig. 50), you can set the selection mode of the grid points. You can select grid points one at a time (default mode) or within a set circle. The radius of the circle is set at the **Relative radius of search area** slider.

The selection tool also has an **undo/redo** function, with the help of which you can redo grid point selections that have been removed by mistake. The number of possible undo/redo functions is the same as that of the data itself. The undo/redo function of grid point selections can be activated with the Undo sel. and Redo sel. buttons.

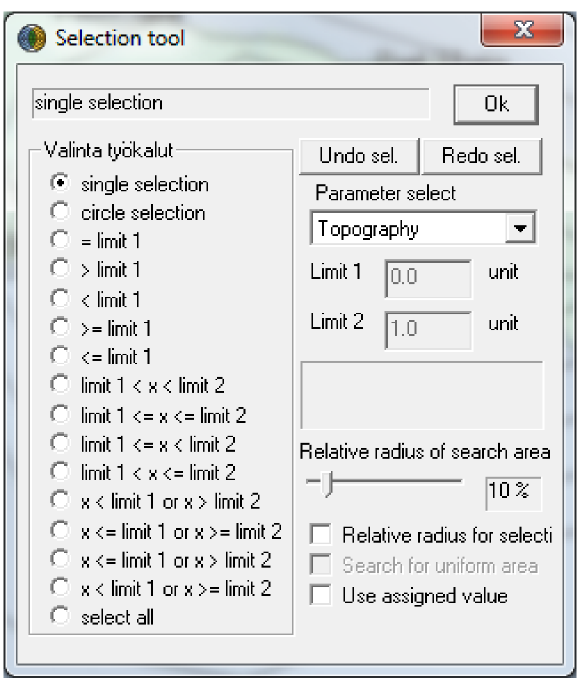


Figure 12 Selection tool dialog for grid points.

4.1.1 Masks

Masks can restrict modifications done with different tools. You can, for example, set up a mask for all land areas (e.g. topo>0), if you want changes to apply only to land areas. Different tools have a setting that takes masks into consideration. SmartTool (See section 9) cannot be restricted with masks, because the SmartTool language makes it possible to set up masks with more conditions and more versatile parameters. Masks are **added/removed** in the Mask dialog, which opens up by clicking with the right mouse button on the letter M on the mapviews upper left corner. Different masks can be the parameters of editable data, the parameters of different models and geographical masks.

4.1.1.1 Calculated masks

To open the Mask dialog, click with the right mouse button on the letter M on the mapviews upper left corner (see Fig. 5). You can enable calculated masks by choosing the **Add** function of masks. In this version, possible calculated masks are **Latitude, Longitude** and **Elevation Angle**. Elevation Angle (EA) is the angle of altitude of the sun. If EA is over zero, it means that the sun has risen. If EA is below zero, it is currently night time. 90 degrees means that the sun is at its zenith. With latitude and longitude masks you can limit rectangular areas.

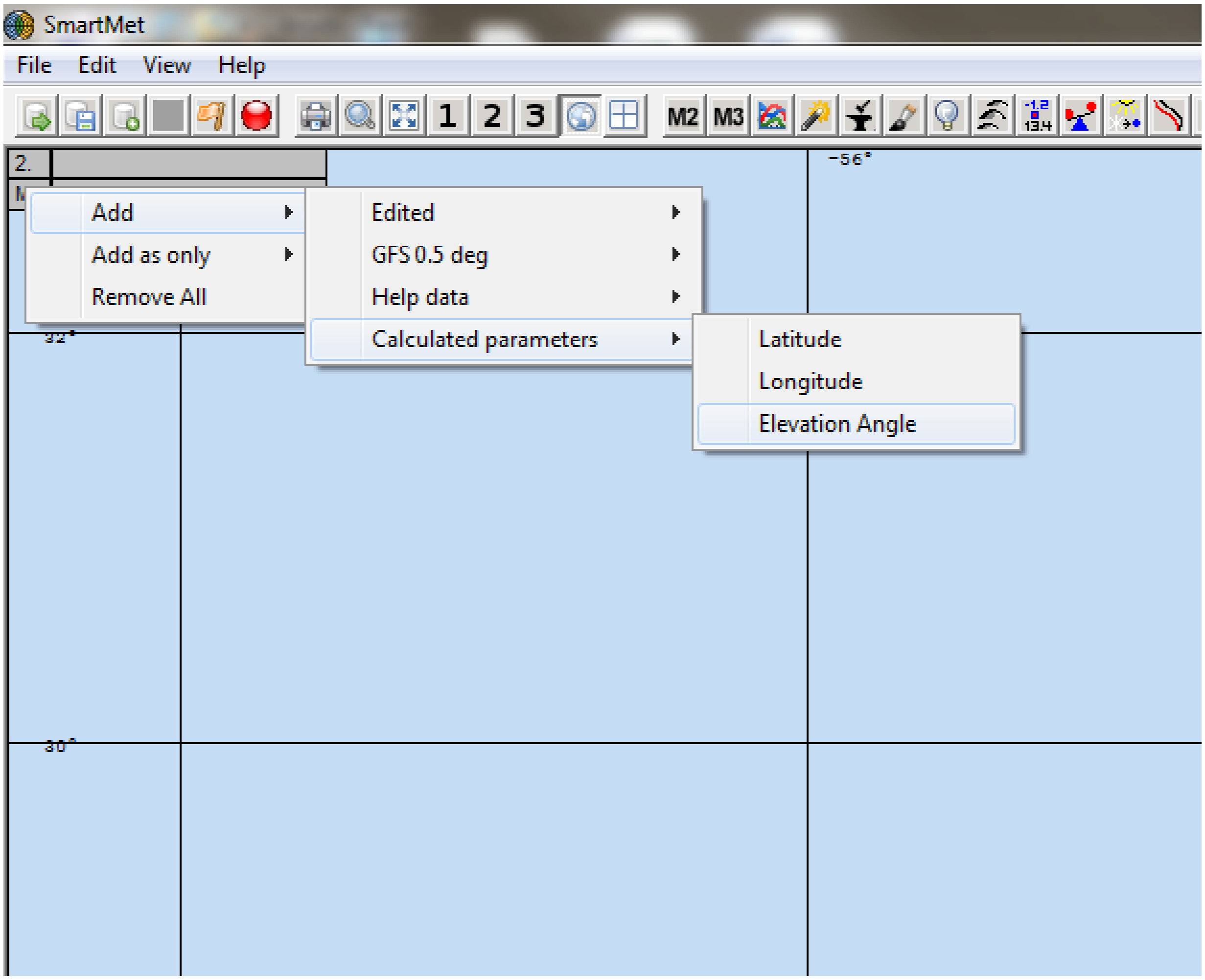


Figure 13 Selection of new masks.

4.1.1.2 Mask view (CTRL+W)

Used masks can be visualized on the map. When you press **CTRL+W**, the editor opens or closes the view (on/off). Masks are shown in a way, that areas with no enabled mask (i.e. modification applies) look normal. Areas, in which a mask prevents a modification, are covered with a dark color (Figure 14). The mask prevents modifications, if the tool in question has been to set to use masks. With ramp masks (See section 9.3.8.7), you can see different shades of color. The darker the color, the less modifications affect the values of the area. The lighter the color, the more changes you can do to the area. “Transparent” areas allow full changes.

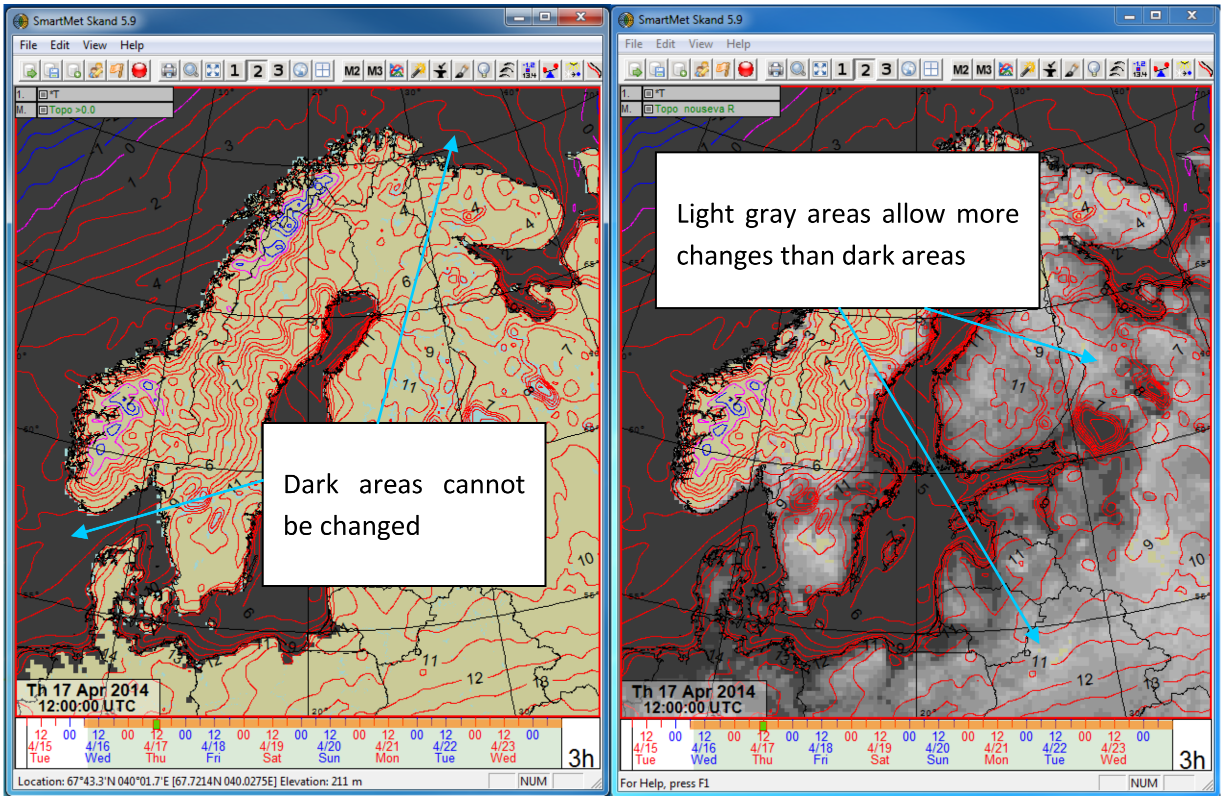


Figure 14 Visualization of masks. The first image shows a land mask (topography > 0) and the second a rising ramp mask of topography from 0 to 250 meters (Topo nouseva R).

5. Data filter tool

The data filter tool can de used for heavy and extensive data modification. It can be used for evening data in relation to time or location, for moving data or for both moving and evening data between locations or times.

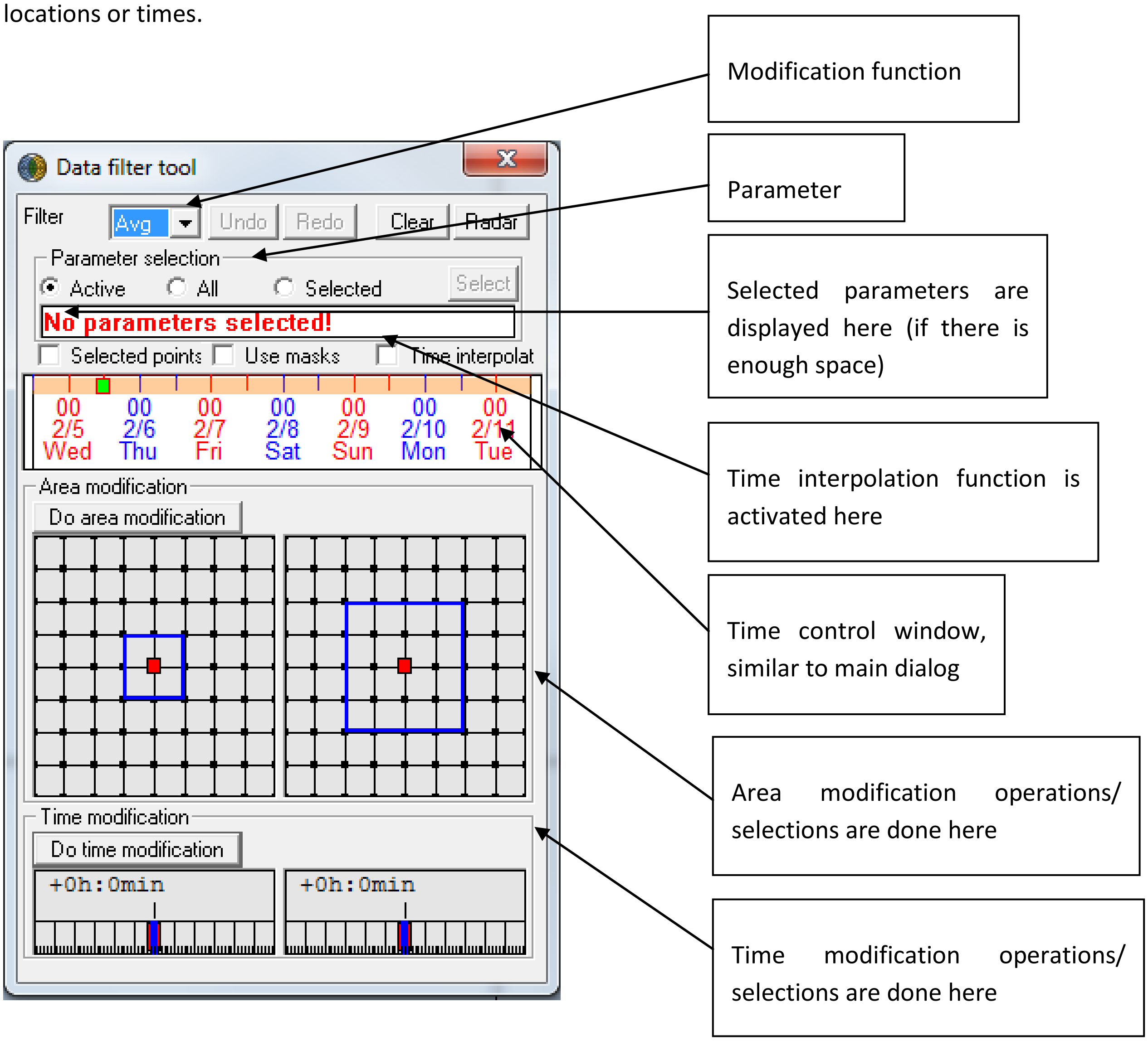


Figure 15 Data filter dialog.

Modifications are made in the data filter dialog shown in Figure 15. In the **Filter** field, you can choose whether you want to calculate the average, the **minimum** or the **maximum** value of the selected points. Changes can be applied to the **activated parameter**, to the **selected parameters** or to **all parameters**. If you choose to apply the change to the activated parameter, you should check in the parameter row (Figure 15) which parameter is activated (which line is framed with red and which parameter is marked with * in the parameter display).

NOTE! If you modify data and the result is not what you expected (if, for example, nothing seems to have changed), you should undo the change (with the undo button).

If you choose **Parameter selection** the **Select button** becomes active. If you click Select opens a Parameter activation dialog where you can select the desired parameters.

Selected points (Figure 15) means that the modification is done only to activated grid points. If you don’t have a marker here, the modification is done automatically to all the grid points.

Use masks (Figure 15) means that in all the modifications the (possibly) activated mask is used.

In the time control window the time line is identical to the one in the map view. The time where the modification is done can be selected with the right mouse button on the time lines both endings.

5.1 Area modification

With the area modification tool you can do different kinds of data evening or moving. In Figure 16 there is the area modification part of the data filter dialog. The area modification is executed by pushing **Do area modification**.

For example: we want to do an evening in the temperature field, which is in the starting time point small but grows towars the ending time point. In Figure 15 we select Average (**avg**) to be the Filter. From the time line we select the desired start and end times for the modification. We select temperature to be the parameter. In the left box in Figure 16 we select the starting time´s area of evening and in the right box the ending time´s area of evening. The little red box is the grid point where the calculated new value is set. The blue box is the area from which the new value (for the red box) is calculated. In Figure 16 example the average is calculated in the starting time point from 9 grid point`s area, and in the ending time point from 25 grid point`s area. Within the time line the calculated area changes linearly from the starting time point to the ending time point.

The size and position of the blue boxes are changed with the mouse as following:

- left click moves the blue box to the clicked place
- right click does nothing
- with the left mousebutton you can drag the box or strech it from the corners
- Ctrl + left click makes the blue box smaller
- Ctrl + right click makes the blue box bigger
- Ctrl + Shift + left click makes the grid point area smaller
- Ctrl + Shift + right click makes the grid point area bigger

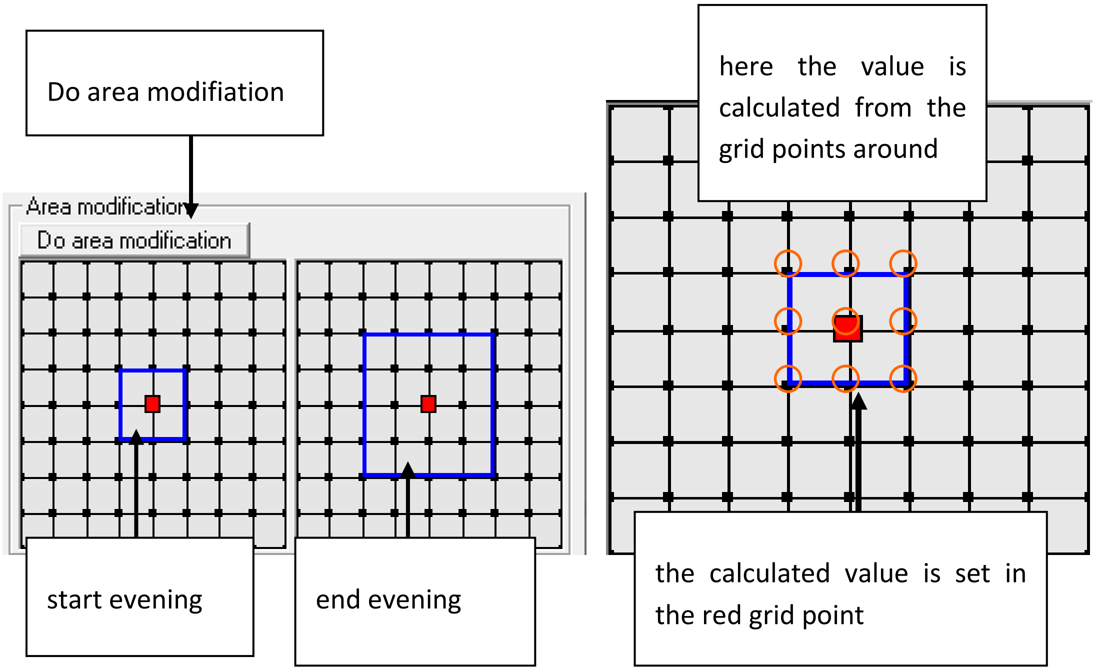


Figure 16 Area Modification part of the data filter tool.

5.2 Time modification

With the time modification you can move the data in time forwards or backwards. The data can also be evened over time steps or both moved and evened at the same time.

The idea in the time modification chart is the same as in the area modification. The left part means the desired modification in the start time point and the right means the desired modification in the end time point. The vertical lines in the chart are time steps of the data. The red box (it cannot be moved) is the time point in which the calculated value is set.

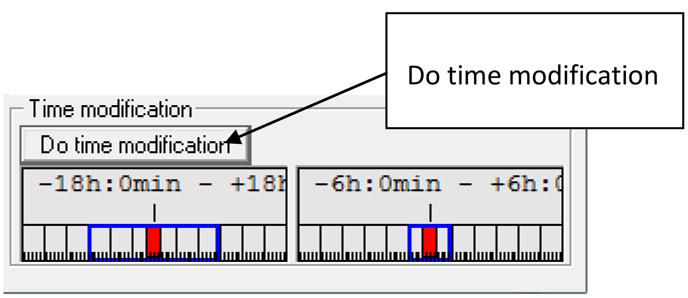


Figure 17 Time modification part in the data filter tool.

The blue box is for the time span from which the value for the red box is calculated. If the blue box is small like in Figure 17 on the left side, it means the data is moved from the time point of the blue box to the red box. In the Figure 17 on the left hand side the data is moved 1 hour and 24 minutes forward. On the right hand side, in Figure 17, for example the average is calculated from five time steps, from which the one in the middle is the time where the new calculated value is set. So an evening over time is done.

The time modification is executed by clicking **Do time modification**.

The size and position of the blue boxes are changed with the mouse as follows:

- left mouse click moves the blue box to the clicked place
- right click does nothing
- with the left mouse button you can drag the box or strech it from the sides
- Ctrl + left click makes the blue box smaller
- Ctrl + right click makes the blue box bigger
- Shift + left click moves the blue box to the nearest hour

6. Spreader Tool 

With the Spreader tool you can edit the parameter values directly on the map view and the changes spread in the surroundings in space and time. The Spreader tool is activated by clicking the Spreader tool button (illustrated in the heading) and a Spreader Tool dialog opens (Figure 18).

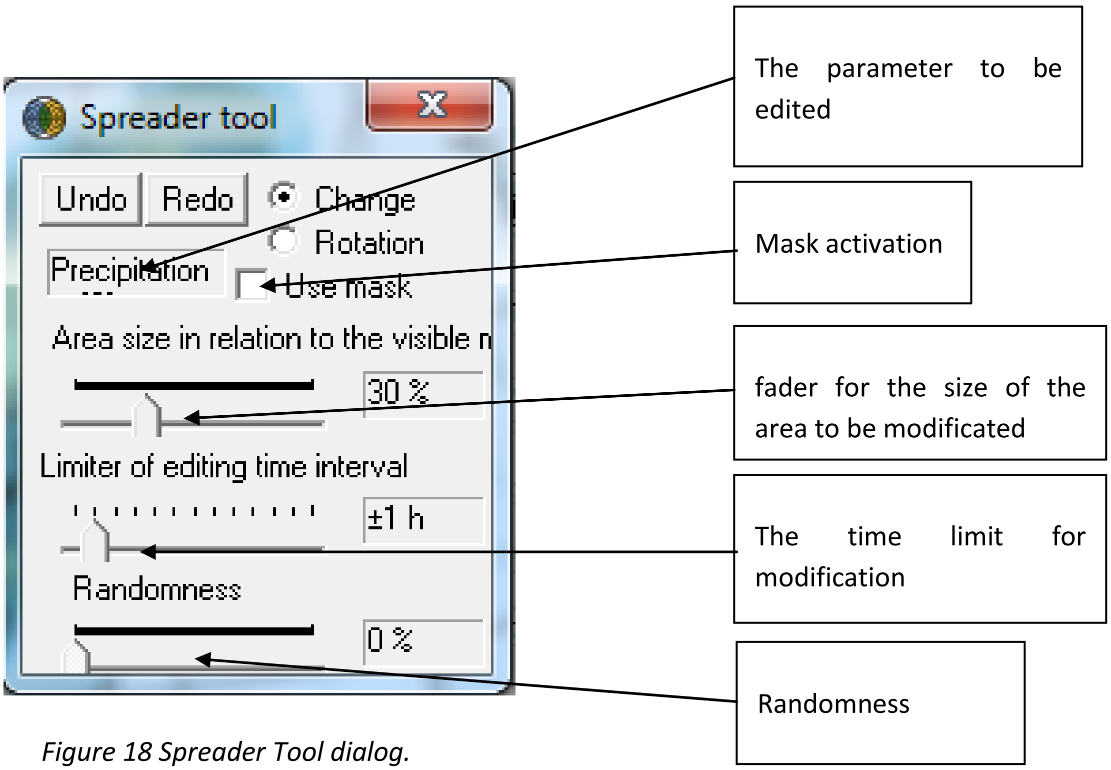


Figure 18 Spreader Tool dialog.

When the spreader tool is on, put the cursor on the map view on that place you want to edit. Then click the left mouse button and keep it clicked. On the map appear a purple cross and a circle around it (Figure 19). This indicates the area to be edited. The circle size is adjusted with the **area size** – fader (Figure 18). In Figure 19 example precipitation intensity is to be edited and the Spreader tool options are set like they are in Figure 18. The area size is 30% in relation to the visible map area and the spreading is done +/-1hour from the time point selected (Here Fr 13/10 21:00 +/- 1 hour).

The tool tip box (Figure 19) shows the original value (the value in brackets after word **orig**.) of the parameter and also the change (the value in brackets after word **diff**:) and the new value (first value). Now when you move the cursor upwards on the map, the values grow – you can see this in the tool tips output. When you move the cursor downwards, the values get smaller – this is also seen on the tool tip. If you now release the left mouse pad the modification is executed (Figure 20). You can cancel the started change by pressing **Esc** and then releasing the mouse button.

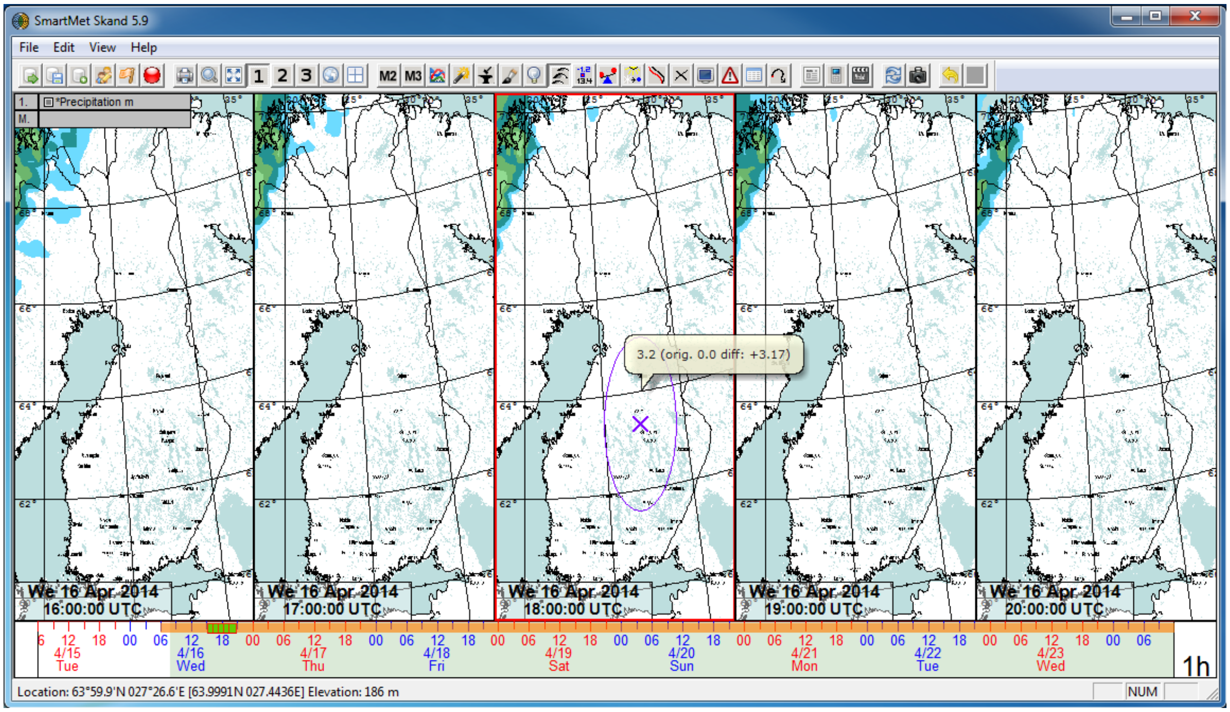


Figure 19 Spreader tool on the map view, no modification done yet.

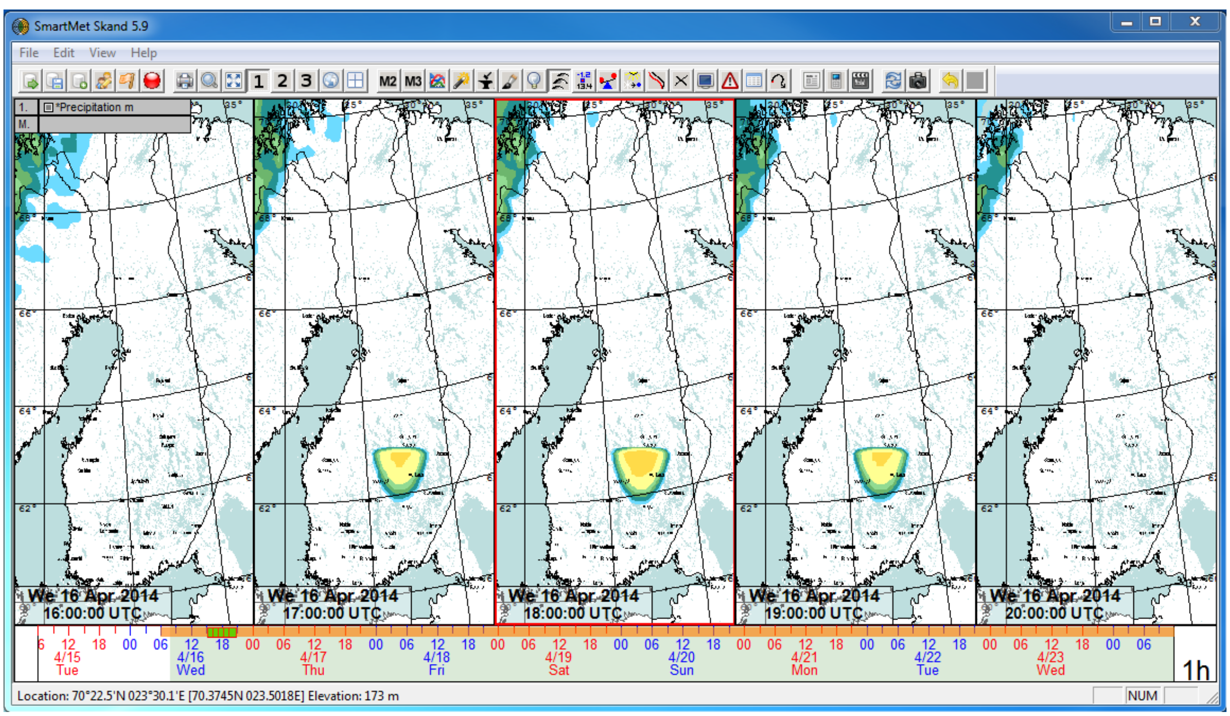


Figure 20 Modification done with the spreader tool. The modification was done in the picture in the middle and spread +/- 1 hour.

7. Control point tool 

Control point editing means that a few points are edited as a time series and the changes spread from these points to the surrounding locations and points of time. The Control point button (see heading) is situated in the toolbar. When you press this button, the editor goes into the control point editing mode (the button remains down). You can quit this mode by pressing the button again.

7.1 Control points on the map

When you are in the CP (**CP = Control Point**) editing mode, CP’s are displayed as boxes and numbers on the map (Figure 21). A CP consists of a box (location on the map) and the figures above and below it. The figure above indicates how much the value of the point changes each time. The figures are color-coded. A red-colored figure indicates that the value is increasing. A green-colored figure indicates that no changes will occur and a blue-colored figure indicates that the value is decreasing. The figure below the CP indicates the value of the point after the editing is accepted. If the change is 0, the present value of the point is displayed. If the inside of the box is blue, the CP is in normal use. If the inside is gray, the point will not be edited. If the box has a green frame, it is not activated. If the box has a red frame, it is activated and the value of the point will be affected by modification.

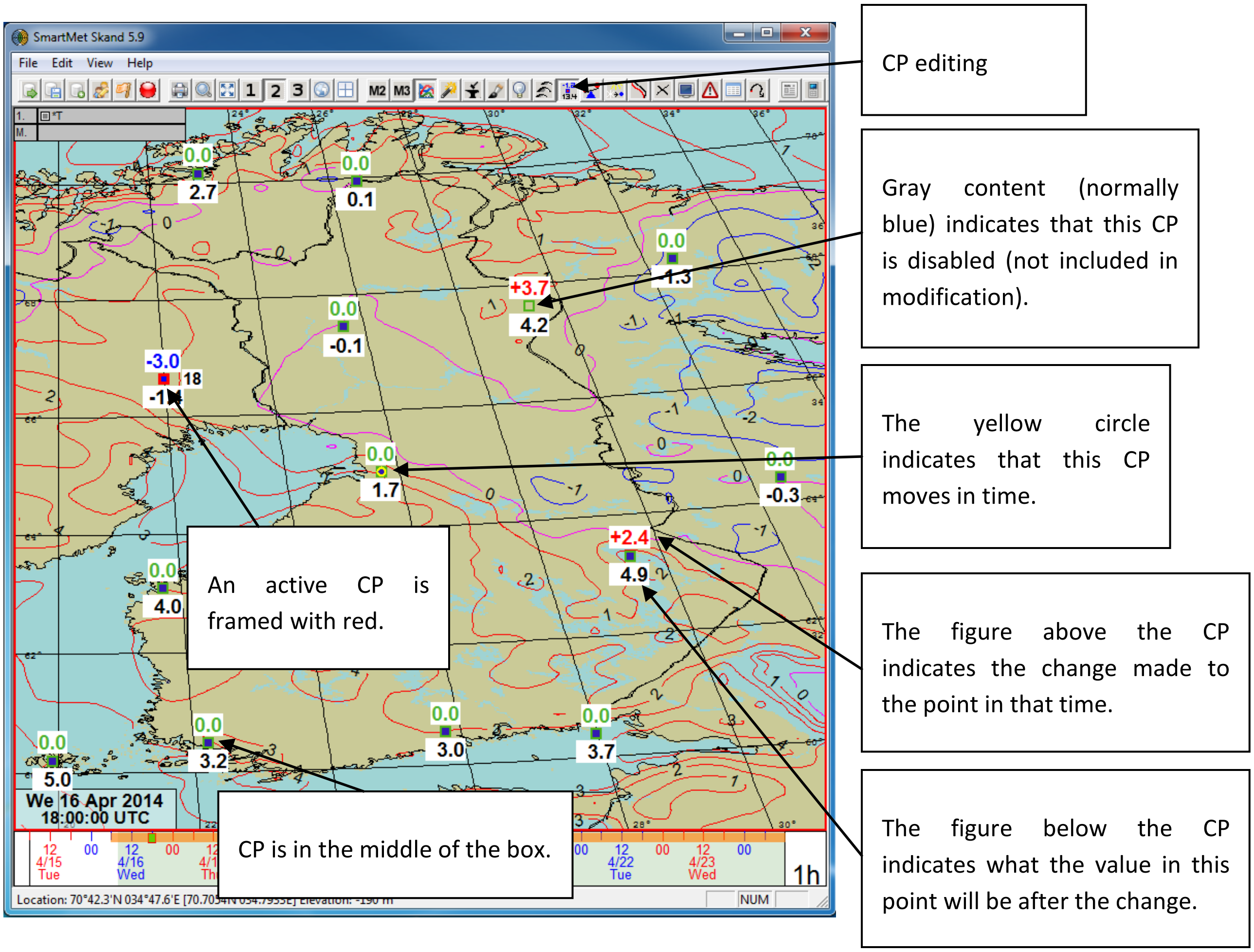


Figure 21 Editor in CP editing (CP = Control Point) mode. Five CP's are displayed on the map.

7.1.1 Manipulating a control point

A CP can be moved with the mouse. Activate the CP by clicking it with the left mouse button. The frame of the CP turns red. Push the left mouse button, keep it pressed down and drag the CP where you want it.

You can add as many CPs as you want on the map. Keep **CTRL+SHIFT** pressed down and click the map with the left mouse button. By trying out different numbers of points you can find out what is optimal for your purpose. This depends on the situation you are editing. You can delete a CP by activating it and pressing the **DEL** key.

When you close the editor, the editor will store the present (changed) CP's in the file. When you open the editor, the CP's stored in the file are brought on display. Only the number of the CP's and their latitude and longitude are stored. No other information is available.

Normally, the inside of a CP is blue, which means that the CP is affected by modification calculations. However, you can disable a CP if you do not want it to be affected by modifications. Press SHIFT and click the CP with the left mouse button to either enable or disable it (depending on its status before). The content of a disabled CP turns gray. By clicking the CP with the right mouse button you can display its modification menu (Figure 22).

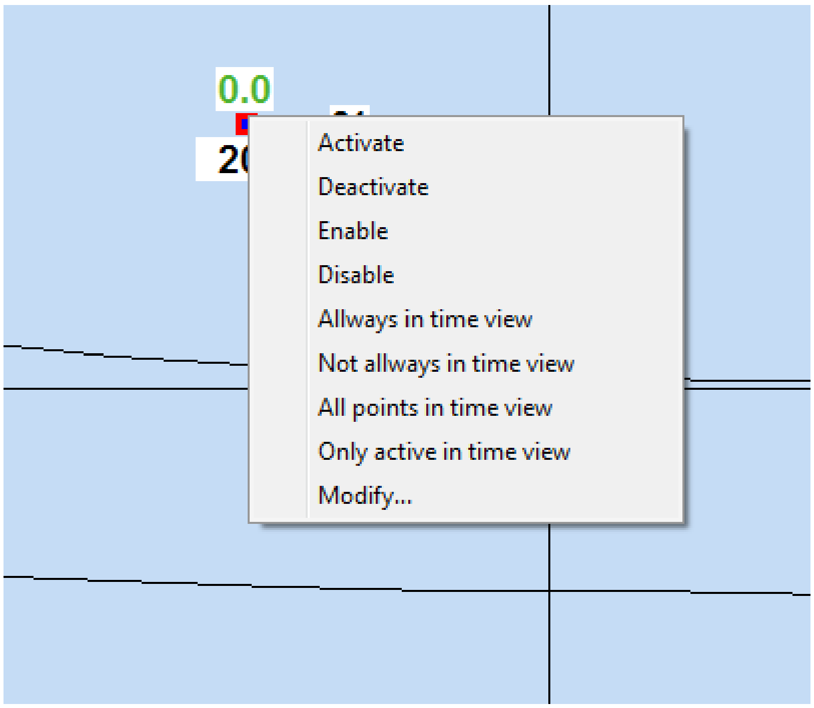


Figure 22 CP modification menu on the map.

7.1.2 Changing and editing values using a control points

Activate the point you want to edit on the map. The grid point closest to this CP will also be activated, and you can view the values of the grid point in the time series editor. You can make parameter-specific change curves for each CP. For example, temperature and total cloudiness have their own curves. You are able to modify the existing curve in the time serial view for the parameter by keeping left mouse button pressed down and drawing the curve. After drawing the curve but before any modifications have been made the parameters curve is a solid blue line (Figure 23) and the new curve that is drawn is a dashed blue line. The amount of change in each timestep is illustrated with red circles connected by a thin black line and the value scale for changes is in the right side of the time series view. There is also an observation curve for the parameter from the nearest observation station (thick red line) and values of different models (thin coloured lines) in the time series view.

After you have drawn new curves ther changes will remain when you activate another CP or change another parameter. Make the necessary changes for the parameter in question in all the CP's and finally press the **Modify button**. Note that the amount of the changes you made will only be reset when you press the **Clear button** or close the editor. This means that once you have made the changes you wanted and pressed modify and you are satisfied with the results you need to press the clear button.

When the modify button is pressed the editor starts calculating change grids starting from the first modified time. Using the change grids, all the grids for that point of time will be modified. This process will be performed for all points of time, until the last point of time that needs to be modified is reached.

Changes that are made to curves are not reset when the Modify button is pressed, because the curves are often laborious to build and it is important that they can be fixed after modifications. Thus you can freely experiment with changes, **but do remember to undo your experiments!**

Masks can be used in CP editing just like in ordinary time series editing.

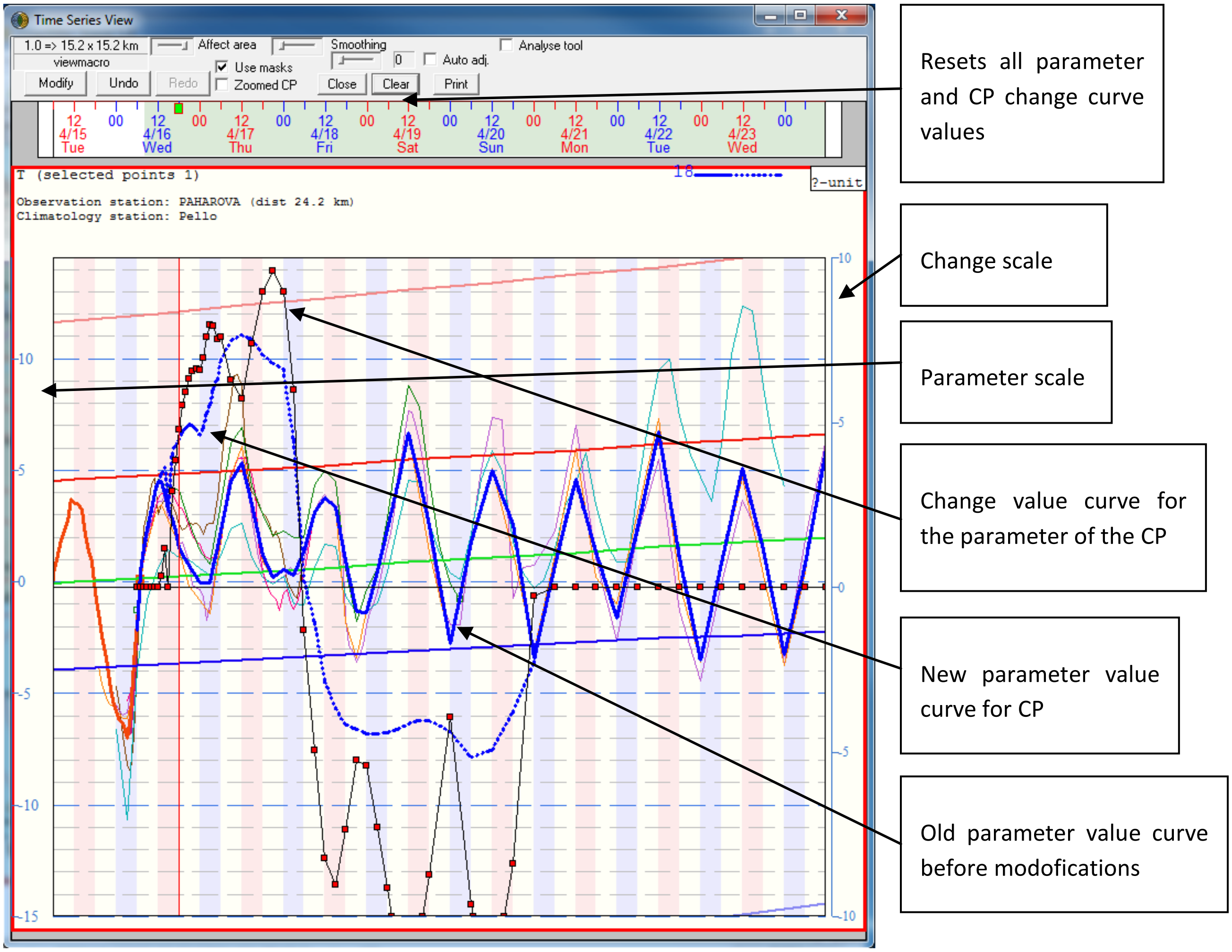


Figure 23 Modifying the change values of the CP selected in the time series editor.

7.1.2.1 Zoomed control points

Editing data with control points is somewhat resource eating procedure for computer and may be slow in some occations. To save the computing time there is an option Zoomed CP in time series window (Figure 24).

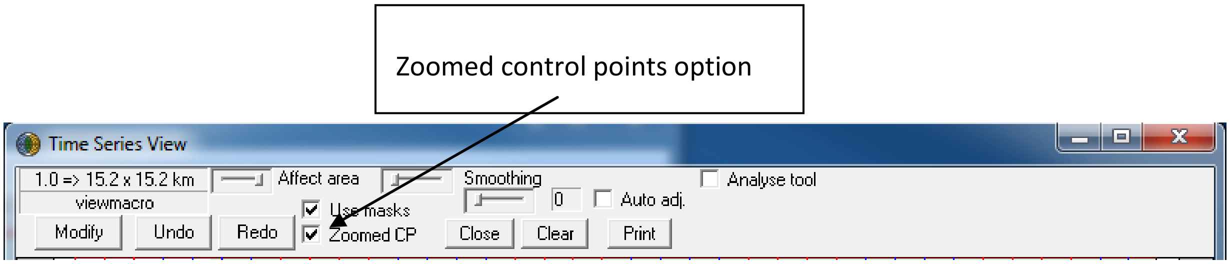


Figure 24 Zoomed control points option in time series view.

Once the Zoomed CP is marked there appears two purple dash-lined rectangles on the main map view of SmartMet (Figure 25). The control point that is modified needs to be inside of these rectangles. When using zoomed CP mode the modifications will only apply inside of the rectangles which means the change won't spread outside of the rectangles. The modification is applied fully inside of the inner rectangle and there is a linear smoothing between the inner and outer rectangle (see axample in figure 25).

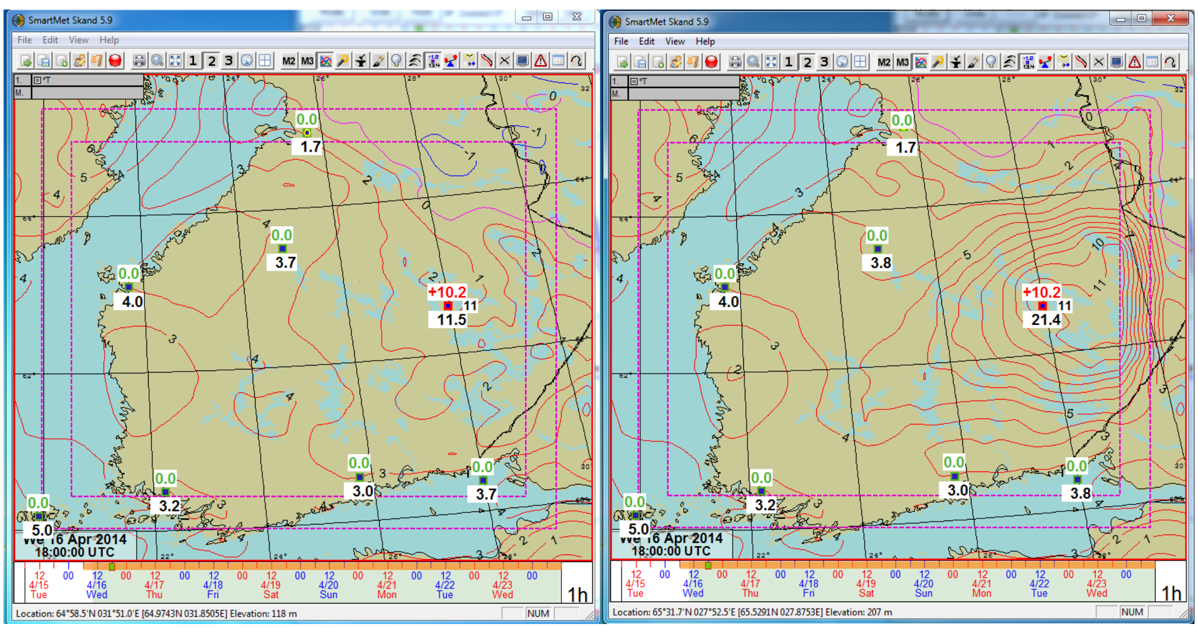


Figure 25 Image of main map view before and after modification when using zoomed control points option. Note the smoothing in the right side image between the two rectangles.

The area of the rectangles is constant in respect of the window size (i.e. when zooming in the area inside of the rectangles gets smaller) so the anly way to decrease the size of the rectangles is to zoom in in main map view. There is however a minimum limit of the area inside of the rectangles. If you zoom in too much a red cross will appear on the map view and modifications can't be made (Figure 26).

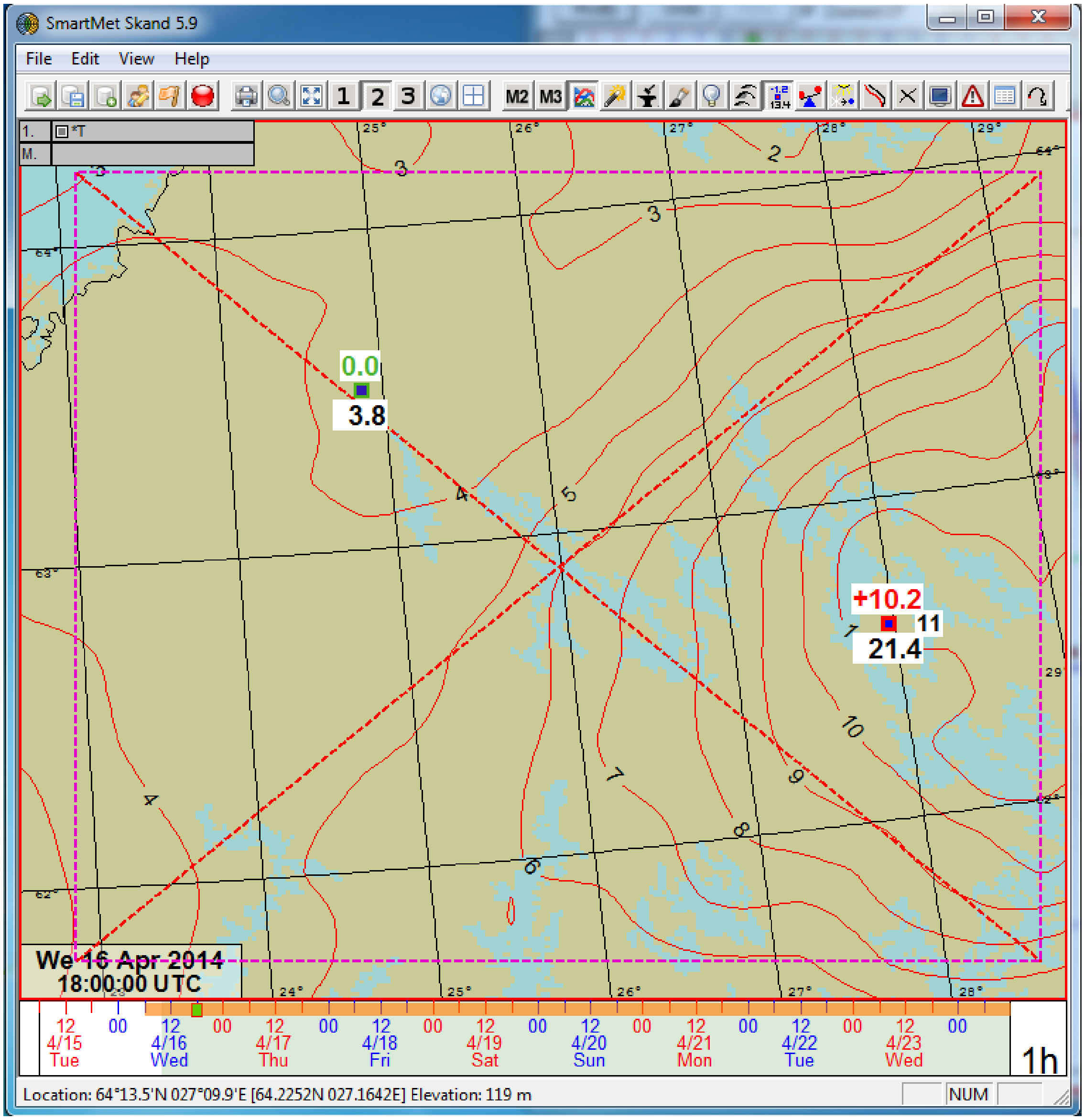


Figure 26 Main map view is zoomed in too much while using zoomed control points.

8. Analyze tool

The name Analyze tool does not refer to making an actual analysis, but to using the analysis field of measured meteorological values (for example temperature, humidity or pressure) for modifying/correcting short-term (the next 12 hours) forecasts. You can use the Analyze tool in the time series view by choosing Analyze tool in the upper part of the screen (Figure 27). Using the mouse, you can drag the vertical red dash line on the screen to where you want it. The line indicates the end of the zone where the tool is applied, which means that the data is modified only up to this point of time. Between the end point of analysis and current time the analysis data is smoothed to the current forecast data. By pressing the **Modify** button you can start modifying either the activated parameter or all the analysis parameters at once. On the right there is two drop down menus. If Use both data isn't marked analysis is made from the first analysis on the menu (i.e. the upper analysis model). (Select **All parameters** from the options in the upper part of the screen.) The result of the modification is shown with a red dash line. Modifications are only applied to the selected points. Modifications can also be limited with **masks**.

The modification algorithm of the Analyze tool:

1) Goes through the selected grid points (depending on the mask).
2) Calculates the difference in the edited data between the most recent analysis time and the present time (Δa).
3) Changes the value of the analysis time directly into the value of analysis in the edited data.
4) Reduces the performed modification (Δa) as time proceeds so that in the last time point there is no modification at all. In other words, the effect of modification decreases gradually the further you move in time.

With analysis tool you can also use two different analysis fields simultaneously. This feature may be uselful for example in cases where you have an analysis field that is working well in some region but not in whole area of the data or in cases where one analysis field doesn't cover the whole area of the data. The two ananlysis field mode is selected from the time series window by marking the **Use both data** selection. Once it is marked the two drop down menus on the right will become valid (Figure 27). The upper is the primary analysis and lower is the secondary analysis. Now if you select points from the map and press modify the primary analysis is applied for the selected points area and the secondary for the rest of the area.

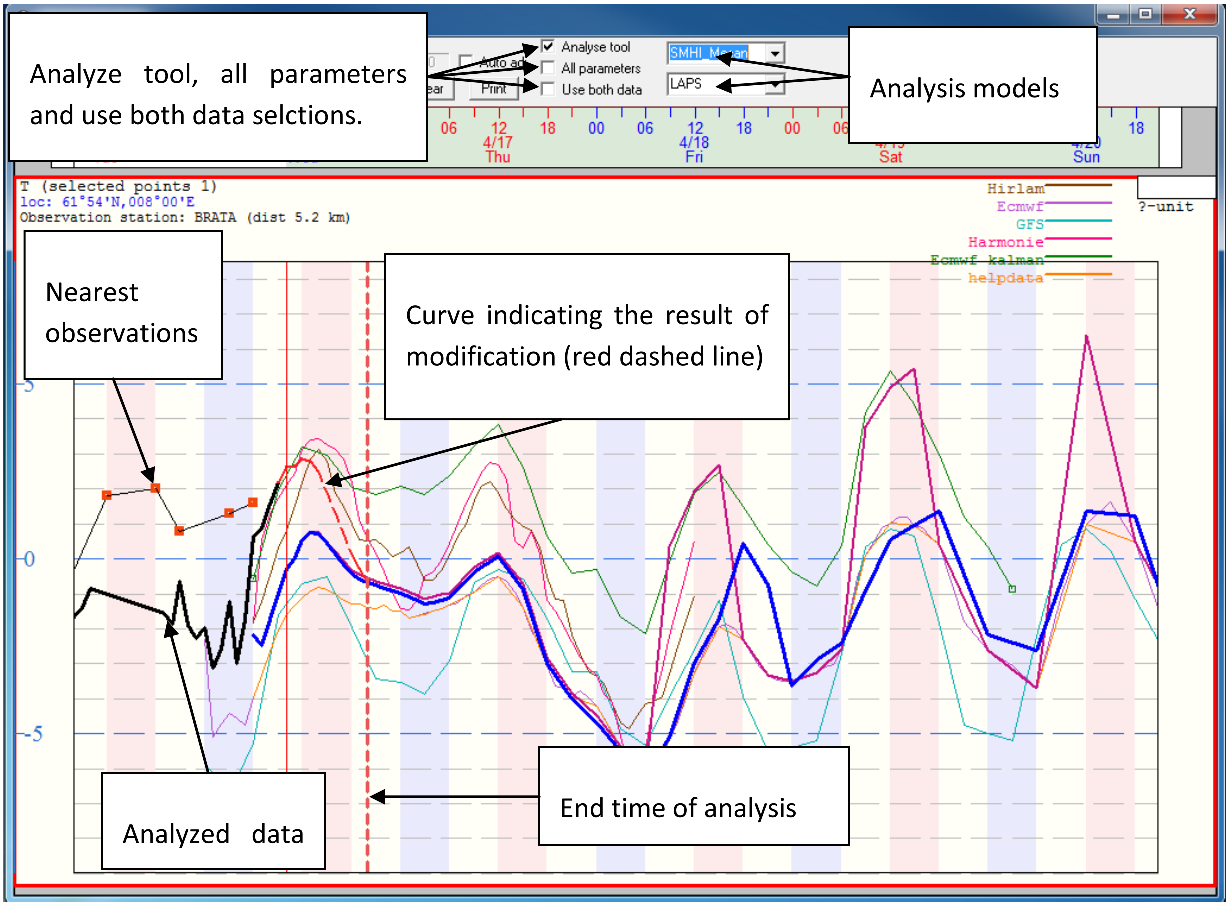


Figure 27 Analyze tool at work. Correction is made for the first 5 hours (the blue line represents the data that is edited and the purple line the most recent data in the database).

9. SmartTool

9.1 Principle

The purpose of the tool is to modify meteorological data with different formulas and rules. The editor uses a "programming" language designed for this purpose. The language is text-based, case insensitive, and it contains different statements and conditions. For example, the statement

```
T = T + 1
```

increases the value of the parameter T (=temperature) with one degree. The change can be applied to selected times and selected areas. By default, the calculations are applied to all locations at all times.

Because the language is purely text-based, different modification statements (=macro or script) can be saved in files for later use or distribution.

In addition to different modifications, the language can be used for sanity checks (which of course also modify the data). For example, the condition and possible modification "if there are no clouds, there can be no rain, so rain is set to zero" consists of the statements:

```
IF(N == 0)
        RR = 0
```

NOTE! The statements used in this document are generally not valid and are only meant to illustrate technical points.

Examples of SmartTool language expressions are commented on in this document. Comments are color-coded. Green color indicates that the code is acceptable and red color indicates an error.

```
T = T + 1 // Comments on normal code with green text

T = T +* 1 // Warnings and comments on erroneous code with red text
```

The SmartTool macro language has many words reserved for different variables or functions. Often there are alternative ways of expressing them. For example temperature, **T**, can be written in formula as either **T** or **t**. Precipitation form **PREF** can also be written **PreF**, **Pref** or **pref**. Words consisting of two parts, such as DISTSEA (=DISTance to SEA) can also be written DistSea or distsea. This flexibility with lower case and upper case is intended to facilitate making macros. However, the system is not entirely case-independent, because in future versions certain signs may be interpreted differently depending on capitalization.

9.2 Checking the minimum and maximun values of parameters

When macros are used to calculate values for different parameters, minimum and maximum values are checked automatically. If the result would be above or below the value limits set for the parameter, it is changed into that limit value. Settings for minimum and maximum values of each parameter can be made separately in the parameter's Drawing properties dialog. The dialog can be opened by clicking on the parameter (in the gray parameter selection field) with the right mouse button and selecting Properties… in the pop-up menu. The limits are set in the dialog in Parameter value, Low limit ____ High limit ____.

9.3 SmartTool dialog

Modifying macros are first executed with a simple dialog where different statements can be typed and executed by pressing the action button. The dialog opens by choosing View – SmartTool dialog in the editor. When the dialog (see fig 63) is open, it always floats on top of the main window (the map). It can be hidden beneath other dialogs/windows. The present dialog version is a pretest version and can only be used for testing purposes.The macros displayed on the screen are executed by pressing the **Execute macro** button.

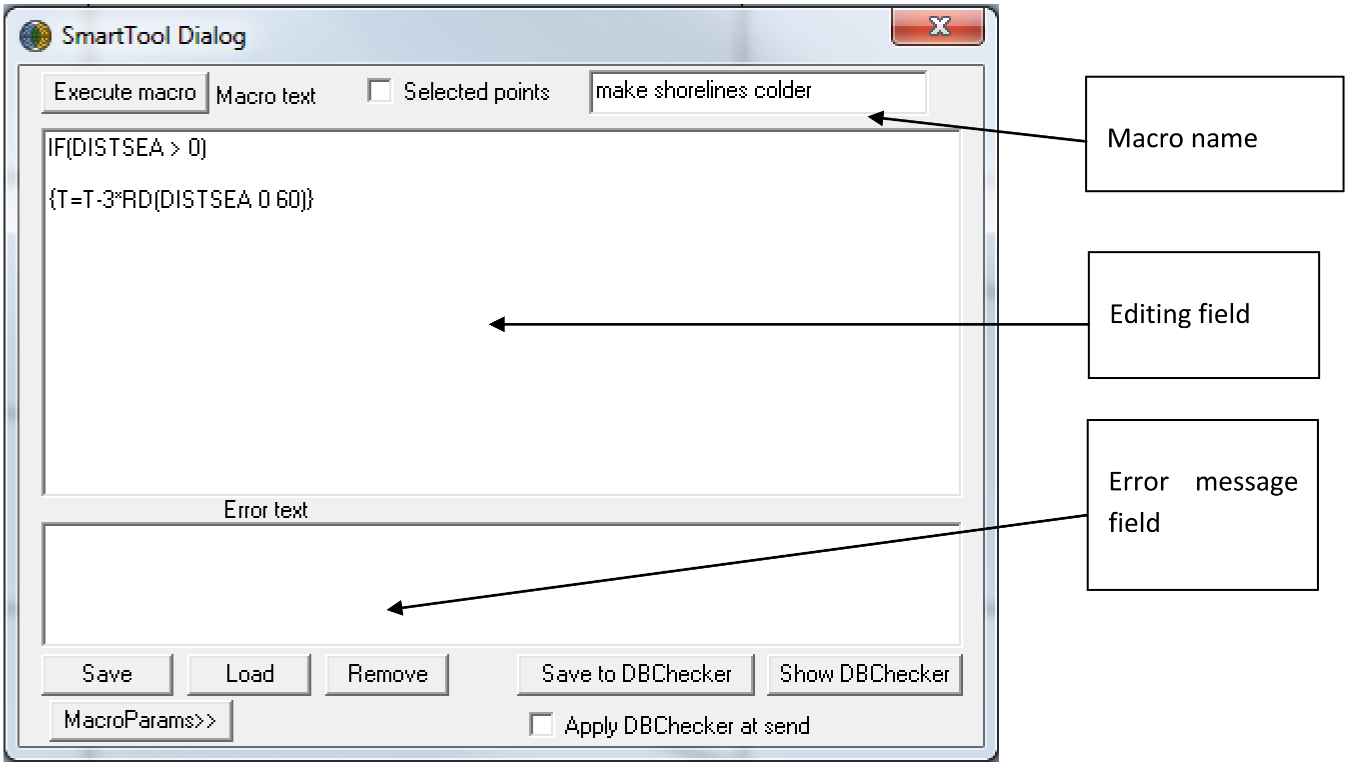


Figure 28 SmartTool dialog is open and contains sample statements.

9.3.1 Editing dialog

Macros are written in a normal text editing window, where text can be typed, copied and pasted. Text can be several lines long and continue above or below the borders. If all text does not fit on the screen, scroll bars will appear automatically so that you can scroll to the part of the text that you want.

9.3.2 Loading and saving macros

To facilitate the use of macros they can be saved in files. Operations are always applied to the macro text in the editor. If you choose Save as, the program will ask for a file name. If you choose the Save button, the macro is automatically saved in the most recent file. It is recommendable to save macros in a file with the extension st (SmartTool), for example "My_macro1.st". When you press the Load button, the dialog will ask you which macro you want to load on the screen. When the editor starts up, it tries to load the most recently saved/loaded macro.

9.3.3 DBChecker

DBChecker is an attempt to make automatic sanity checks to the data with the help of macros. The editor will automatically load a file named DBChecker.st (DO NOT DESTROY THE FILE), which can be applied to the edited data when exported into the database. For this, the “Apply DBChecker at send” checkbox needs to be checked. You can load the DBChecker on display, modify it and save it. You cannot change the name or location of the file.

9.3.4 Time limiter (time line control)

By default, macros are executed for all edited times. This can be limited with the orange time line in the time line control window. The limiter used in the Modification dialog works similarly with SmartTool, which means that calculations are performed for the times inside the orange line segment. There are two ways of adjusting the time limiter: First you can drag the start and the end time from the orange bar by pressing and holding down the left mouse button in the ends of the orange bar. Secondly you by keeping CTRL+SHIFT pressed down the left mouse click will adjust the start time and the right mouse click will adjust the end time.

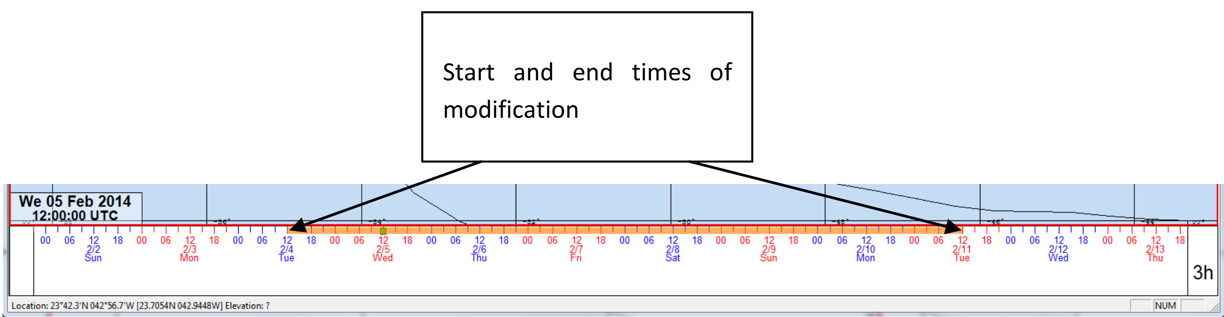


Figure 29 The limiter in the time line control window can also be applied for macros.

9.3.5 Location limiter (selected points)

By default, macro calculations are performed for all locations (grid points). However, you can limit this not only with condition statements (macro masks, see section 9.3.8.3) but also by selecting the grid points you want by highlighting them and by checking “Selected points” in the SmartTool dialog. The points are selected with the selection tool (see section 4.1) while keeping the CTRL key and the left mouse button pressed down. Keeping SHIFT pressed down the let mouse click will unselect points.

9.3.6 Error situations

With macros, many errors can occur: Statements may be constructed wrong, typing errors can be made, a parameter may not be found etc. In this case, all calculations are interrupted and canceled. An explanatory text will appear in the error field of the SmartTool dialog. This can be helpful for correcting statements.

If the macro was performed successfully, the following message appears in the error field:

1. Macro execution status: OK

Messages always have consecutive numbering (1, 2, 3 …) to indicate that the program has attempted to execute the macro. Macros or the editor may contain errors, due to which nothing happens, or seems to happen: In this case you can look at the consecutive numbering to make sure that the program has executed or attempted to execute the macro.

9.3.7 Comparison to original data

If you want to examine the changes made to the parameters more carefully, you can choose to show the difference compared to the original data. Differences are displayed in the same way as differences in time steps. The type of difference is indicated with colors familiar from temperature values. Positive isolines are colored red and negative ones blue. Difference fields are displayed with dense isolines and can only be modified by changing the difference parameter’s isoline settings for calculating the density of the isolines.

You can activate differences to the original data by clicking the right mouse button on the parameter you want (in the gray parameter selection field) to display the popup menu of the parameter and then selecing **Diff orig. on**.

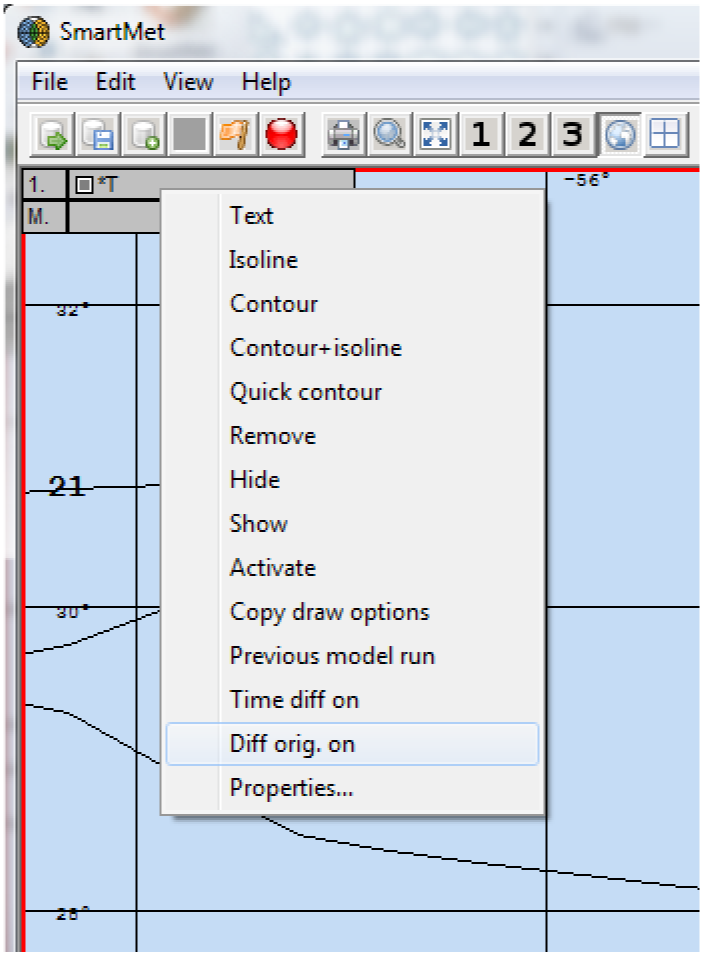


Figure 30 Differences compared to the original data (Orig.) are activated in the parameter menu.

You can disable the difference field in the same menu where you activated it. Now the menu command is **Diff. orig. off**.

The original data is generated from loaded data. You can update this “original data”. This is done by taking pictures of it, for example, by pressing the camera button (Make edited data copy) or in the Edit menu. This allows for step-by-step comparison between modifications.

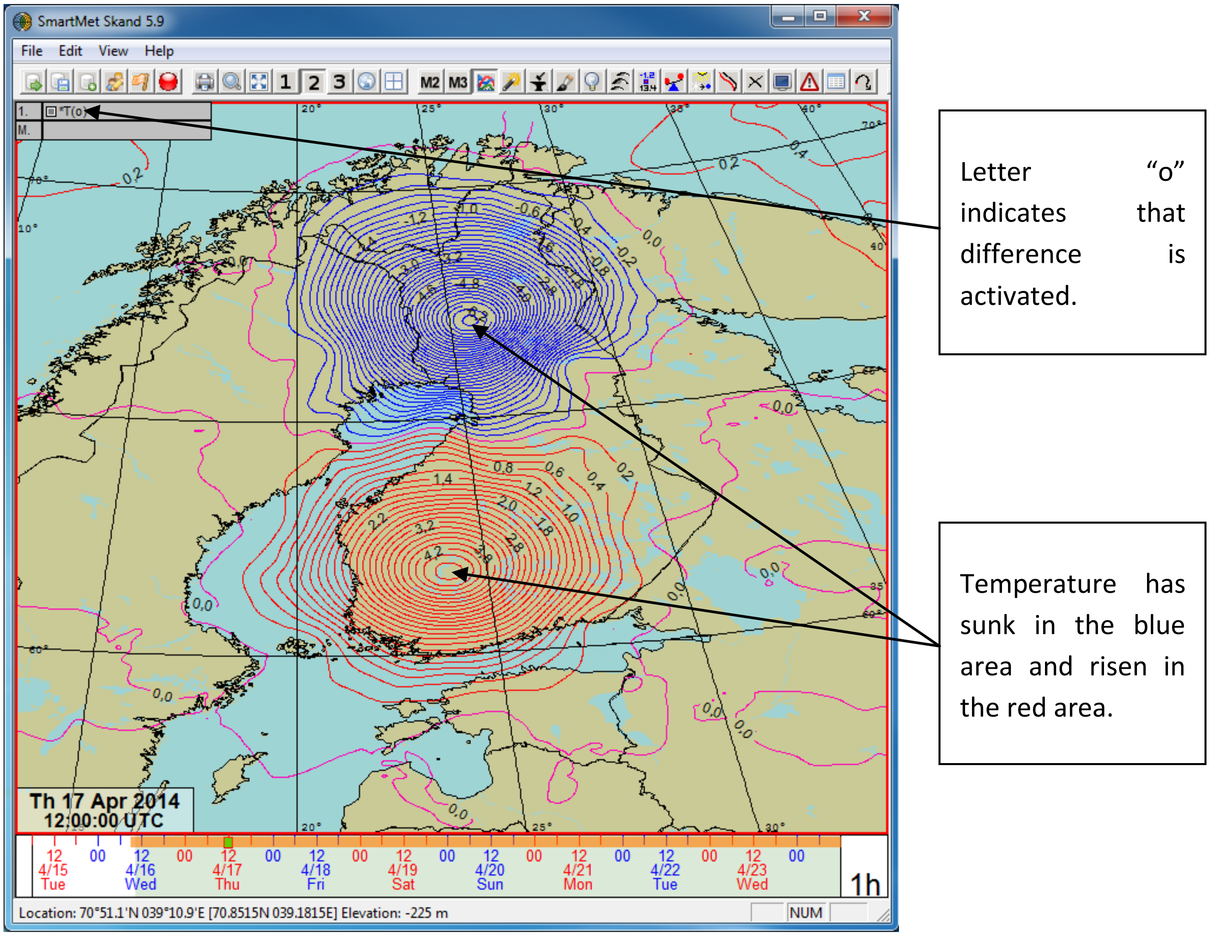


Figure 31 Difference between modified and original data can be seen in the difference field.

9.3.8 SmartTool language (= macro language)

9.3.8.1 Assignment statement

A simple assigment statement, where temperature is increased by one degree:

T = T + 1

Assignment statements always have a variable, to which values are assigned (T = ?). The variable has to be a parameter in the edited data (for example temperature, T). The statement needs to be in a line of its own, but its length is unlimited and it can contain as many different variables, operators and functions as you want.

Arithmetic operations

In SmartTool language all the normal arithmetic operations can be done in plain language. You can make as complex statements as you want and define the order of the calculations with brackets. You can use any variables and functions from the editor database, as well as other variables and functions. For example:

T = T + P * 0.123 – RH/100 * WS + (T - DP) ^ 2

Arithmetic operation	Description	Order of calculations: 1 first, 2 next etc.
+	Addition	3
-	Subtraction	3
*	Multiplication	2
/	Division	2
%	Remainder. e.g. 8 % 3 = 2	2
^	Exponentiation, e.g. 2 ^ 3 = 8, 3 ^ 2 = 9 and 9 ^ 0.5 = 3 (square root!)	1

Table 1 Arithmetic operations and their calculation order.

9.3.8.2 Statement block

A statement block consists of a group of assignment statements. The number of statements in a block is unlimited. You can also define a block by putting it inside curly brackets, {}. For example, a block of three statements can look like this:

T = T + 1

```
P = P + 1

RH = RH + 1
```

You can also write it like this:

```
{
    T = T + 1

    P = P + 1

    RH = RH + 1
}
```

In the example, indents are used only to facilitate reading. Combining statements into a block means that calculations within one block are always performed as one entity (see section 9.3.8.4).

9.3.8.3 Condition expressions or masks

With condition expressions, you can set conditions to limit certain modifications. You can, for example, limit changes to land areas or rain areas etc. A condition expression block always begins with an IF expression. Assignment statements in the IF expression are calculated, if the conditions are met. If necessary, you can continue an IF expression with ELSEIF or ELSE branches. In this way, you can have several different operations with different conditions side by side. A simple IF expression:

```
IF(T – DP > 2) // If the difference between temperature and DP is over 2 degrees, do   something…

    T = T + 1
```

You can add as many calculations as you want to an IF expression either by simply placing them after the IF expression or by placing them into a calculation block:

```
IF(T – DP > 2)
```

T = T + 1

P = P + 1 // Both assignment calculations are performed only, if the condition is true.

IF(T – DP > 2)

{

T = T + 1 // Only the calculations inside the block are performed, if the condition is true.

}

P = P + 1 // This calculation is always performed irrespective of any conditions.

You can create complex condition expression constructions, if you use other condition expressions besides the IF expression:

IF(T – DP > 4)

{

T = T + 1

P = P + 1

ELSEIF(T – DP > 2) // Actually means that (T-DP) is between 2 and 4.

T = T + 2

P = P + 2

ELSE // otherwise, if T – DP <= 2, the ELSE branch is executed

T = T + 3

P = P + 3

}

Condition expression	Description
IF	If you use condition expressions, you have to start with this.
ELSEIF	This can be used to add more conditions to calculations. It can only be used after IF statements. **NOTE! In the present version you cannot use several ELSEIF consecutive statements!**
ELSE	If other conditions are not met, this calculation is performed. It can follow an IF expression or an ELSEIF expression.

Table 2 Condition expressions

Comparison operations

Several comparison operators can be used in condition expressions to determine, whether the condition is true. For example:

IF(T > 4)

IF(T >= 4)

IF(T < 4)

IF(T <= 4)

IF(T == 4) // A bad example for T, because the value is hardly ever exactly 4.

IF(T != 4) // Not very useful for T, because almost all values meet the condition.

Comparison operator	Description	Means the same as (you can use also)
>	Greater than	
>=	Greater than or equal to	
<	Lesser than	
<=	Lesser than or equal to	
==	Equal to	=
!=	Not equal to	<>

Table 3 Comparison operators for condition expressions.

Combining conditions

You can combine conditions in condition expressions without limitations. All conditions together may need to be true or one of the combined conditions may need to be true. In the following example, two meterological masks have been used alone and combined in different ways (Figure 32):

```
IF(T > 14) // Mask nr. 1

IF(P < 1020) // Mask nr. 2

IF(T > 14 AND P < 1020) // Masks nr. 1 and 2 are both true at the same time (intersection)

IF(T > 14 OR P < 1020) // Either mask nr. 1 or 2 is true (union)
```

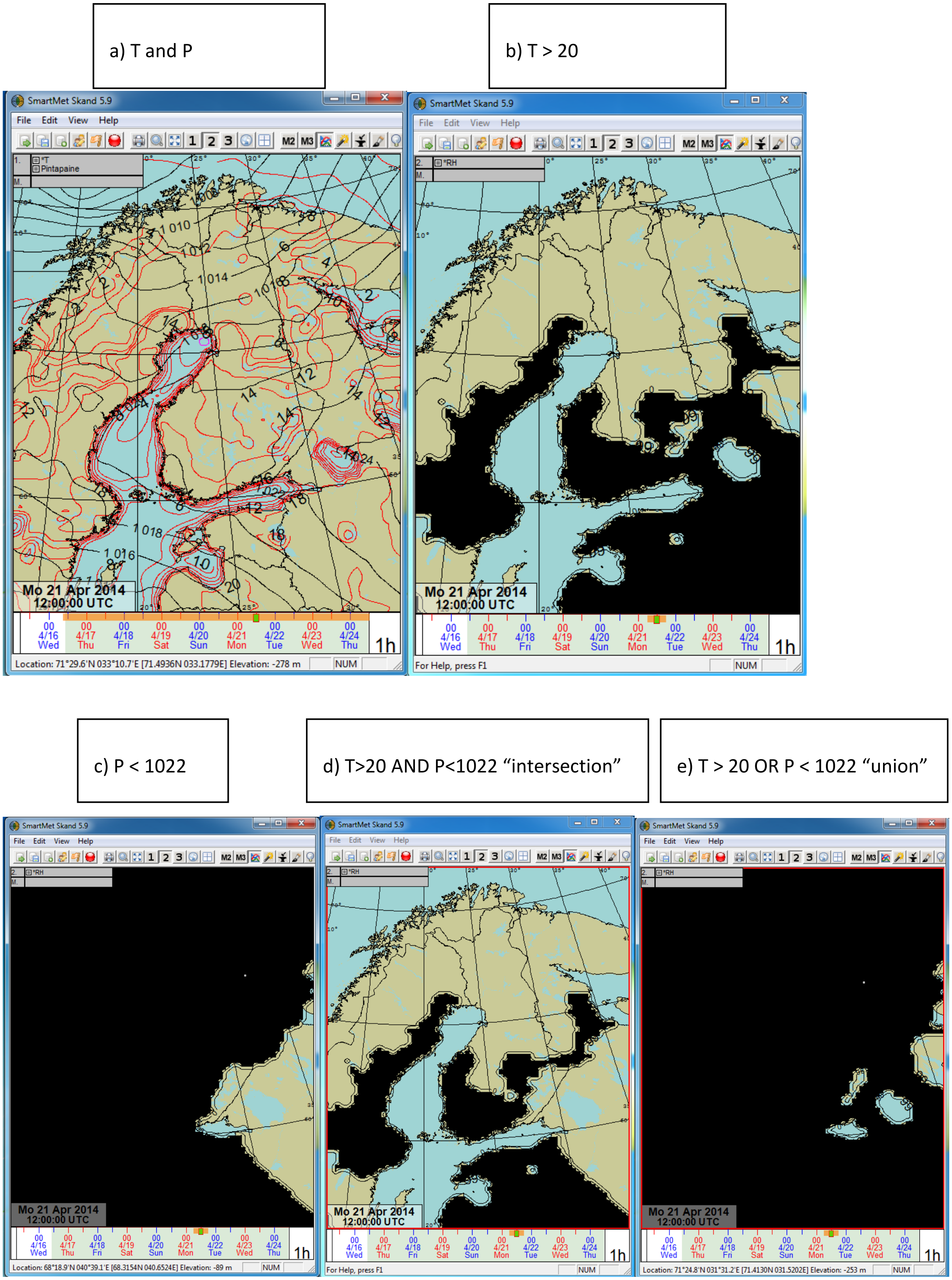


Figure 32 Figure a) shows the temperature and pressure status at a given point of time. Figures b), c), d) & e) have different condition masks related to the situation. The masked area is shaded.

Combining operators	Description	Means the same as (you can use either)
AND	Both conditions need to be true, intersection.	&&
OR	One of the conditions needs to be true, union.	||

Table 4 Combining operators for conditions.

9.3.8.4 Order of performing calculation blocks in the language

Independent calculation blocks are performed completely before moving on. This means that calculations are performed for all the times and locations that you want to use. IF-ELSEIF-ELSE structures are performed as one block. In the following example system calculations are performed as explained:

```
Statement_block1  // This is first performed completely.

IF(Condition_expression1)  // IF expression is performed completely until Statement_block4 is reached.

        Statement_block2

ELSEIF(Condition_expression2)

        Statement_block3

ELSE

        Statement_block4

Statement_block5  // Finally, this is performed completely.
```

9.3.8.5 Nested condition expressions

You can make condition expressions with the macro language almost without limitations. They can be consecutive or nested. Each condition expression can contain an unlimited number of calculation blocks and their calculation statements can also contain condition expressions with an unlimited number of calculation blocks. The only restriction is, that an IF expression can be followed by only one ELSEIF expression. Example:

```
IF(....)
{
  IF(...)
  {
    T = ....
    IF(...)
    {
      T = ....
    }
    ELSE
    {
      T = ....
    }
    T = ....
    IF(....)
    {
      T = ....
    }
    ELSEIF(....)
    {
      T = ....
    }
    ELSE
    {
      T = ....
    }
  }
}
ELSE
```

```
{
   T = ....
}
etc.
```

9.3.8.6 Variables, producers, levels and constants

Different variables, producers etc. need to have their own names in the SmartTool language. The names have been kept as short as possible and are derived from English terms. For example T stands for temperature, and so on.

Variables

Variables can be used in macro language for assigning values, for calculations and for different conditions.

Meteorological variables

The following table briefly presents meteorological parameters used as variables. Parameter units are not named, because they depend on the employed data.

Name	Description
T	Temperature
P	Air pressure
RH	Relative humidity
KIND	K-index
DP	Dew point
LRAD	Long-wave radiation, earth radiation
SRAD	Short-wave radiation, solar radiation
WD	Wind direction
WS	Wind speed
N	Total cloudiness

CL	Amount of low clouds
CM	Amount of middle clouds
CH	Amount of high clouds
RR	Precipitation
PREF	Precipitation form (see values in Table 9)
PRET	Precipitation type (see values in Table 10)
THUND	Probability of thunder
FOG	Fog density (see values in Table 11)

Table 5 Different available meteorological variables.

There are some variables, to which you cannot assign values, but which you can use in calculations and conditions. For example:

```
IF(HSADE == 60)  // This is ok. (60 = Occasional slight rain)

HSADE = 61  // This cannot be done, no assigning!
```

Name	Description
FL1BASE	Flight Level 1 cloud Base
FL1TOP	Flight Level 1 cloud Top
FL1COVER	Flight Level 1 cloud COVER
FL1CLOUDTYPE	Flight Level 1 cloud TYPE
FL2BASE	Flight Level 2 cloud Base
FL2TOP	Flight Level 2 cloud Top

FL2COVER	Flight Level 2 cloud COVER
FL2CLOUDTYPE	Flight Level 2 cloud TYPE
FL3BASE	Flight Level 3 cloud Base
FL3TOP	Flight Level 3 cloud Top
FL3COVER	Flight Level 3 cloud COVER
FL3CLOUDTYPE	Flight Level 3 cloud TYPE
FL4BASE	Flight Level 4 cloud Base
FL4TOP	Flight Level 4 cloud Top
FL4COVER	Flight Level 4 cloud COVER
FL4CLOUDTYPE	Flight Level 4 cloud TYPE
FL5BASE	Flight Level 5 cloud Base
FL5TOP	Flight Level 5 cloud Top
FL5COVER	Flight Level 5 cloud COVER
FL5CLOUDTYPE	Flight Level 5 cloud TYPE
FL6BASE	Flight Level 6 cloud Base
FL6TOP	Flight Level 6 cloud Top
FL6COVER	Flight Level 6 cloud COVER
FL6CLOUDTYPE	Flight Level 6 cloud TYPE
FL7BASE	Flight Level 7 cloud Base
FL7TOP	Flight Level 7 cloud Top
FL7COVER	Flight Level 7 cloud COVER
FL7CLOUDTYPE	Flight Level 7 cloud TYPE
FL8BASE	Flight Level 8 cloud Base

FL8TOP	Flight Level 8 cloud Top
FL8COVER	Flight Level 8 cloud COVER
FL8CLOUDTYPE	Flight Level 8 cloud TYPE
FLCBBASE	Flight Level CB BASE
FLCBCOVER	Flight Level CB COVER
FLMINBASE	Flight Level cloud minimum BASE
FLMAXBASE	Flight Level cloud maximum BASE
AVIVIS	Aviation Visibility
VERVIS	Vertical Visibility

Table 6 Aviation parameters

Producers

Normally, when you use a meteorological variable as such, it refers to the parameter of the edited data. So T means the temperature in the edited data. If you want to refer to different model datas, you need to specify this in the parameter. If you want to use the temperature from ECMWF in the data you are editing, you can do it like this:

T = T_EC

The parameter is followed by the underline character, after which comes the identifier of the producer/model. At this stage, it is possible to assign only into the data that is edited. Therefore the following expression does not work:

T_EC = T_HIR // This is not possible, assigning into EC data.

The producer of the most recent edited data is identified with MET.

IF(T > T_MET) // If T is higher than T in the last edited data,

```
T = T_MET // the last edited T will be used.
```

Name	Description
HIR	Hirlam model data
EC	ECMWF model data
ORIG	Original data that is edited
CLIM	Climatological data
MET	The most recently edited data
HELP	The help data (data edited in the help mode)
GFS	GFS Model data
SYNOP	Obsevation data (†)
METAR	Metar-observation data (†)
TEMP	Sounding obsevrvation data (†) (example: T_temp_850)

Table 7 Different producers used in the SmartTool language. † Smartmet is able to calculate analysis field directly from the observations by interpolating the data. ***NOTE!*** *The Smartmet doesn't analyze the density of the observation network. Therefore data might be unusable, especially in the cases where the observation network is sparse.*

Levels

Calculations can also have parameters on different levels. This is indicated by adding a level identifier to the parameter. For example, pressure level temperature for 500hPa is expressed with T_500. If you want to indicate pressure level temperature from Gfs, you need to have both identifiers T_500_GFS. Examples:

T = T_850 * 0.93 // Pressure surface of the edited data (if available)

T = T_850_GFS * 0.93 // GFS's temperature for 850

T = T_GFS_850 * 0.93 // GFS's temperature for 850

The examples above show that you can have the pressure level and producer in which ever order you want.

Level name	Description
925	Identifier of the pressure level
850	Identifier of the pressure level
700	”
500	”
300	”

Table 8 Applicable pressure levels.

Instead of using pressure levels one may also use metric heihgt in SmartMet. This is done by using notation Z(HEIGHT IN METERS). Example: Insert to surface temperature the temperature in the height of 100 meters from GFS- model.

T = T_GFS_z100

Named variables (var x = ?)

Script language has variables which are named. Variables are introduced with “var” as follows:

var x = 5 * T

After this, you can use the variable x like a meteorological variable, meaning that you can assign new values to it and use it in calculation and condition statements. After you have introduced the variable, you refer to it using only its name (without “var”).

```
IF(x > 15)

        T = x – Td
```

A variable is not a single value used in calculations. Its value can vary depending on time and location. It is calculated separately for each time and grid point (even if a constant value was assigned to it).

Identifiers of parameters, producers and layers

The SmartTool translator in the editor understands many predefined variables, such as T for temperature and P for pressure. All the most important variables are already available in the translator. However, if the translator does not understand a parameter and you know its ID number, you can use that in the name of the parameter variable. The ID for temperature is 4, so if you want, you can use the name of the variable, par4 instead of T:

```
IF(par4 < 0)

        par4 = par4 * 1.1
```

The most important producers are known by the translator, but if necessary, you can use a number to identify the producer. For instance, prod131 is the identifier for ECMWF (T_EC = T_PROD131). Example:

```
IF(T > T_prod131) // If the forecast temperature is higher than the EC temperature,

        T = T_prod131  // use T from EC.
```

The translator also understands the most important pressure levels. If you need other levels, you can make them available as follows: T_LEV500 (= T_500).

```
IF(T_LEV32 > 12) // for example temperature hybrid level 32
```

```
T_LEV32  =  T_LEV32 * 1.1
```

You can also combine all the above, if needed. PAR4_PROD131_LEV500 (or alternatively with producer and level in reversed order PAR4_LEV500_PROD131) is EC temperature at 500mb pressure surface.

Variable values for certain discontinuous parameters

Some parameters are discontinuous and have codes for their values. In macro language, you need to use these code numbers directly to construct conditions, such as "if it is raining watery snow":

```
IF(PREF == 2)
```

Precipitation form (PREF)

Values	Description
0	Drizzle
1	Water
2	Sleet
3	Snow
4	Freezing drizzle
5	Freezing rain
6	Hail
7	Value missing

Table 9 Different forms of precipitation.

Precipitation type (PRET)

Values	Description
0	No value
1	Broad-scale rain
2	Convective rain
3	Value missing

Table 10 Different types of precipitation.

Fog density (fog)

Values	Description
0	No fog
1	Light fog
2	Dense fog

Table 11 Different densities of fog

Static variables

Name	Description	Unit
TOPO	Topographic altitude	m
SLOPE	Slope of ground surface (from horizontal plane, values between 0 and 90, at sea 0)	degree
SLOPEDIR	Direction, where the slope goes down steepest (between 1 and 360, at sea 0)	degree
DISTSEA	Shortest distance to sea (at sea 0)	km
DIRSEA	Direction to sea (between 1 and 360, at sea 0)	degree
DISTLAND	Shortest distance to land (on land 0)	km
DIRLAND	Direction to land (between 1 and 360, on land 0)	degree
LANDSEEMASK	The propotion of land in the surface area (lakes included in calculations)	%
RELTOPO	Height in relation to surroundings (100 on top of a hill and 0 at the bottom of a hollow)	%

Table 12 Different static variables wich won't change over time.

Calculated variables

Name	Definition
LAT	Latitude of the calculation point
LON	Longitude of the calculation point
EANGLE	Elevation angle. How high (between -90 and 90 degrees) the sun is at the time in the location. If the number is above zero, the sun at least peeks above the horizon. The greater the number, the higher the sun is. At night the values are negative.

Table 13 Calculated variables.

Example:

```
IF(EANGLE > 0)  // If the sun is above the horizon.
   T = T + 1  // Increase temperature
```

Constants

You can add constant numerical values to the language.

```
T = T + 1.5
T = T + (T – DP) * -1.5
```

Named constants and a missing value

Script language has constants which can be used in scripts.

```
PI = 3.141…
```

MISS = 32700 (this means a missing value)

Using previous model runs

The values from previous model run is obtained with notation *parameter_producer***[-1]**. The values from the third newest model run is obtained with notation …**[-2]** respectively. Example:

T = T_GFS[-1] // insert to temperature the temperature from the previous model run of GFS.

Example 2: Calculate the average temperature of the last three model runs (GFS):

T = (T_GFS + T_GFS[-1] + T_GFS[-2]) / 3

Assigning a missing value

Script language has to be very precice with missing values.

Missing values easily occur in different situations and they cannot be placed freely in the edited data. This is confirmed by experience in operative editing.

You cannot assign a missing value for temperature using the normal method of assigning values. You can only assign a missing value for temperature with the following expression:

T = MISS

This expression must not contain anything else and MISS cannot be replaced with number 32700.

9.3.8.7 Functions

The SmartTool language has a great number of functions for various purposes.

Using integration functions

With integration functions, you can calculate statistical values, such as minimum and maximum values, averages etc. for areas and points of time. Whether integration is areal or temporal depends on the number of parameters in the function. If the function has three parameters, you use areal integration; if it has five parameters, you use temporal integration.

NOTE! You cannot assign formulas to parameters in integration functions. You need to use constants, such as –1 and 1.

NOTE! At least in the present version, integration function parameters do not change over time.

Integration functions

You can use the following functions:

Name	Description
Avg	Normal arithmetic average
Min	Function for finding the minimum value
Max	Function for finding the maximum value
Sum	Calculates the sum of the elements

Table 14 Integration functions.

Areal integration

Areal integration can be employed for evening radical changes or for transferring data onto a new area. Here, area means a grid point "box" of a certain size with a relation to the calculated grid point. For example, the request of fuction

T = MAX(T –1 –1 1 1)

searches from the location of each calculated grid point a maximum value for temperature within the desired box. Parameters for areal integration (MAX) are:

1. T, the desired parameter (temperature)
2. –1, left edge of the box (grid point move from the calculation point)
3. –1, bottom edge of the box
4. 1, right edge of the box
5. 1, top edge of the box

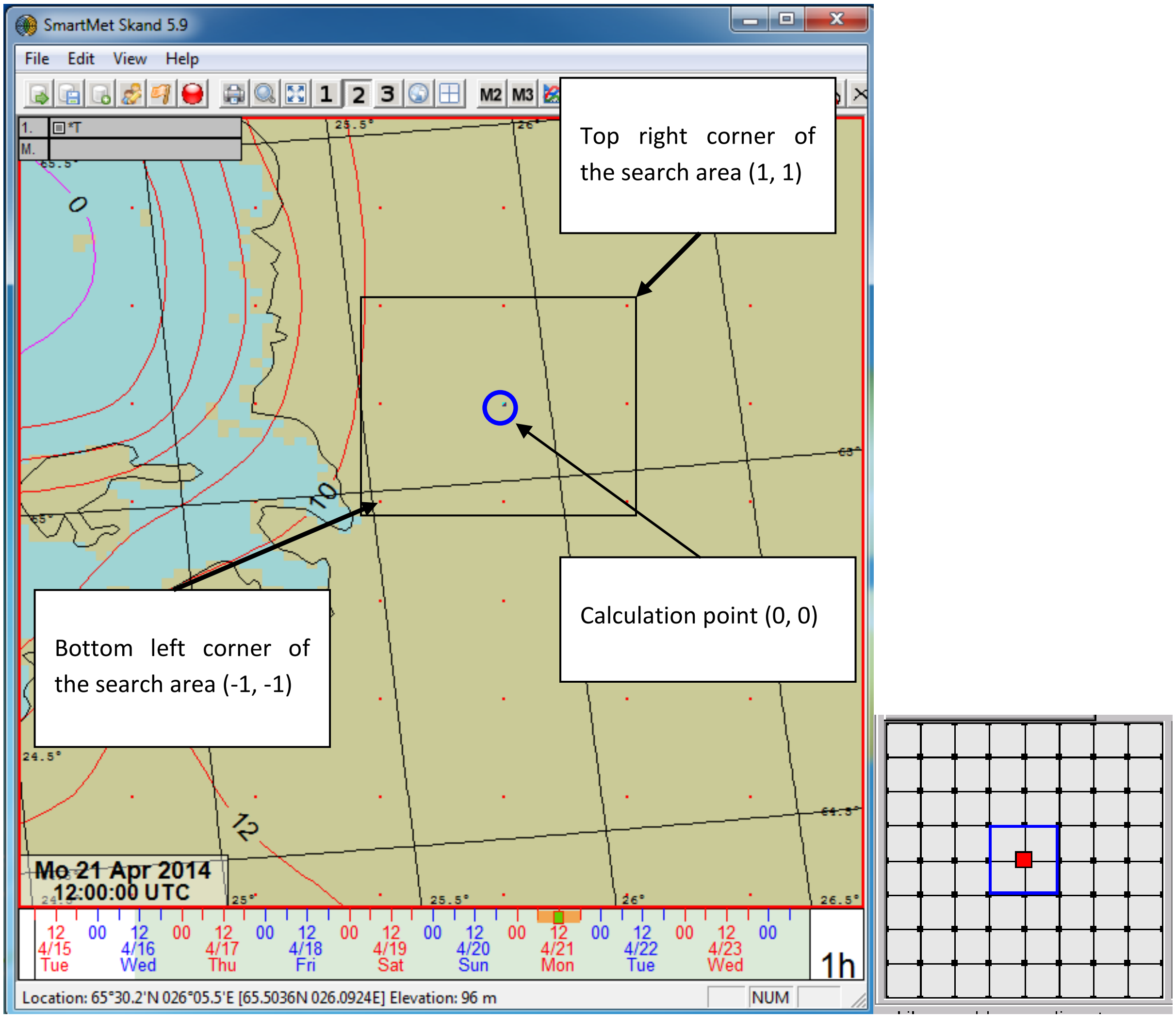


Figure 33 Example. Search box with grid points of expression (T = MAX(T –1 –1 1 1)). This means, that when the blue circle is reached in the calculation, maximum temperature is searched among the grid points in the box. The red points are temperature grid points. The area corresponds to the area setting in the Modification dialog figure.

Temporal integration

Temporal integration can be employed to even data over time (removing extreme values) or to move data in time. Time means the number of time steps you want to use. You can, for example, even temperature over three time steps:

T = AVGT(–1, 1, T)

In temporal integration, the sample parameters for functions (AVG) are:

1. T, the desired parameter (average of temperature)
2. –1, starting point for integration is one time step backwards (from the point of time in the calculated data)
3. 1, end point for integration is one time step forward (from the point of time in the calculated data)

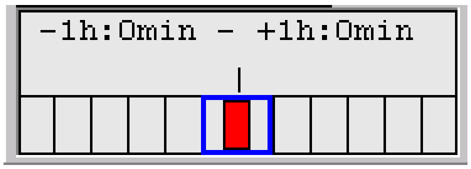


Figure 34 Function parameters correspond to the time settings in the Modification dialog figure.

The time integration can also (or even better) be done with the following functions:

Name	Description
Avgt	Normal arithmetic average
Mint	Function for finding the minimum value
Maxt	Function for finding the maximum value
Sumt	Calculates the sum of the elements

Table 15 Temporal integration functions

For example we make a MacroParameter (See manual part 1. chapter 4) to calculate the GFSmodel`s 24 hour precipitation sum (for the *past* 24 hours).

RESULT = SUMT(-23, 0, RR_GFS)

Note the commas and the order of the arguments in the bracklets. With the help of the commas you are able to set an equation in the sta of RR_EC, for example (RR_EC + RR_HIR) / 2.

Vertical functions

Maximum, minimum etc. values may also be obtained between different heights, pressure levels, model levels or flight levels by using vertical functions. In vertical functions the height limits are expressed either in heights (m), pressure (hPa), flight levels (hft) or model levels (number). Example: A MacroParameter that searches the maximum wind speed between 0 and 5 000 meters from GFS model:

result = vertz_max(ws_GFS, 0, 5000)

In this example **vert** means that it is a vertical function and right after that **Z** defines that the calculation is made by using metric height (for example notation **vertp** would do the calculation between pressure levels). **max** means that the maximum value is to be found. Inside brackets **ws_GFS** defines that the parameter is wind speed fom GFS model and values ater that, separated by commas, that the calculation starts from **0** meters and reaches up to **5000** meters.

The following vertical functions for metric heights are defined in SmartMet:

Metric height functions	Explanation
vertz_max(par, z1, z2)	Maximum value of the parameter between heights z1 and z2
vertz_min(par, z1, z2)	Minimum value of the parameter between heights z1 and z2
vertz_avg(par, z1, z2)	Average value of the parameter between heights z1 and z2
vertz_sum(par, z1, z2)	Sum value of the parameter between heights z1 and z2
vertz_get(par, z)	The value of the parameter in a certain height z
vertz_findh(par, z1, z2, value, nth)	Certain value (value) of parameter between heights z1 and z2. Nth means the ordinal of the value from the bottom (1 = first, 2=second, etc. 0 = last). If Nth is too big (i.e. value doesn’t occur that many times) the fuction returns the last one which is found.
vertz_findc(par, z1, z2, value)	Find count of certain value. Returns how many times certain value has occurred between heights z1 and z2.

vertz_maxh(par, z1, z2)	The height of the maximum value of the parameter. Unit is the same as in height limits.
vertz_minh(par, z1, z2)	The height of the minimum value of the parameter. Unit is the same as in height limits.

Table 16 Vertical integration functions for metric heights.

Corresponding functions of the last table for pressure levels, flight levels and model levels are as follows.

Pressure level functions	Flight level functions	Model level functions
vertp_max(par, p1, p2)	**vertfl_max(par, fl1, fl2)**	**vertlev_max(par, hyb1, hyb2)**
vertp_min(par, p1, p2)	**vertfl_min(par, fl1, fl2)**	**vertlev_min(par, hyb1, hyb2)**
vertp_avg(par, p1, p2)	**vertfl_avg(par, fl1, fl2)**	**vertlev_avg(par, hyb1, hyb2)**
vertp_sum(par, p1, p2)	**vertfl_sum(par, fl1, fl2)**	**vertlev_sum(par, hyb1, hyb2)**
vertp_get(par, p)	**vertfl_get(par, fl)**	**vertlev_get(par, hyb)**
vertp_findh(par, p1, p2, value, nth)	**vertfl_findh(par, fl1, fl2, value, nth)**	**vertlev_findh(par, hyb1, hyb2, value, nth)**
vertp_findc(par, p1, p2, value)	**vertfl_findc(par, fl1, fl2, value)**	**vertlev_findc(par, hyb1, hyb2, value)**
vertp_maxh(par, p1, p2)	**vertfl_maxh(par, fl1, fl2)**	**vertlev_maxh(par, hyb1, hyb2)**
vertp_minh(par, p1, p2)	**vertfl_minh(par, fl1, fl2)**	**vertlev_minh(par, hyb1, hyb2)**

Table 17 Vertical integration functions for pressure, flight and model levels.

The limits in vertical functions may be given either lower first, upper last or vice versa. However vertical functions are always calculated from the surface to upwards. This might be meaningful in minh/maxh functions where heights for minimum or maximum values is searched and if there happens to be two or more occurrences of min/max values in the data in different heights.

PEEKXY

The function PEEKXY(par xdiff ydiff) looks at a value, for example one grid point up and one to the right. Xdiff then means shift of grid point measured on the x axis and ydiff shift measured on the y axis. East and north are positive directions.

Example:

```
var TgradX = PEEKXY (T -1 0 -1 0) - PEEKXY (T 1 0 1 0)
```

The following 850 divergence calculations exemplify the use and results of the functions:

```
// Calculate divergence 850 on pressure surface from EC

// Arbitrary unit

var 85x = ( PEEKXY(u_EC_850 1 0) - PEEKXY(u_EC_850 -1 0) )

var 85y = ( PEEKXY(v_EC_850 0 1) - PEEKXY(v_EC_850 0 -1) )

var 85rv = 85x/2 + 85y/2

RESULT = 85rv
```

PEEKXY2

PEEXY works in different model´s own grids, and therefore you can´t build a comparable grid size, which could be used for example to divide the gradient. The PEEXY2 works always in the same gridsize and grid size functions: GRIDSIZEX and GRIDSIZEY, can be used as help. GRIDSIZEX and GRIDSIZEY return the value of the x- and y- grid size in meters.

For example we make a MacroParameter to calculate the temperature gradient of EC model, and divide it with the grid size and multiply it by 10 000, thus we get the gradient size relative to 10 kilometers:

```
var x = peekxy2(T_EC 1 0) - peekxy2(T_EC -1 0)
```

var y = peekxy2(T_EC 0 1) - peekxy2(T_EC 0 -1)

x = x / gridsizex * 10000

y = y / gridsizey * 10000

RESULT = x + y

Meteorological functions

Smarttool language also includes the following basic functions that are needed especially in meteorology: Gradient, divergence, advection, laplace and rotor. Notation is as follows.

Gradient:

result = grad(P_GFS)

Divergence:

result = div(WIND_GFS)

Advection:

result = adv(T_GFS)

Laplace:

result = lap(T_GFS)

Rotor:

result = rot(wind_GFS)

Mathematical functions

Mathematical functions included here have the following structure: The function is given a value as a parameter and returns a calculation value. The parameter given to the function can be, for example, a formula. Example: The square root of the difference between temperature and dew point is added to the temperature:

T = T + SQRT(T - DP)

The following table briefly introduces different functions.

Function	Description	Formula: y = result and x = argument	Example
EXP	e to the power of x	$y = e^x$	Exp(2) = 7.3891
SQRT	Square root	$y = \sqrt{x}$	Sqrt(9) = 3
LN	natural logarithm	$y = \ln(x)$	Ln(9) = 2.1972
LG	base 10 logarithm	$y = \lg(x)$	Lg(9) = 0.9542
SIN	sin	$y = \sin(x)$	Sin(120) = 0.8660
COS	cos	$y = \cos(x)$	Cos(120) = -0.5
TAN	tan	$y = \tan(x)$	Tan(120) = -1.7321
SINH	hyperbolic sin	$y = \sinh(x)$	Sinh(2) = 3.6269
COSH	hyperbolic cos	$y = \cosh(x)$	Cosh(2) = 3.7622
TANH	hyperbolic tan	$y = \tanh(x)$	Tanh(2) = 0.9640
ASIN	arcus sin (-1 <= x <= 1)	$y = \arcsin(x)$	Asin(0.5) = 30
ACOS	arcus cos (-1 <= x <= 1)	$y = \arccos(x)$	Acos(0.5) = 60
ATAN	arcus tan	$y = \arctan(x)$	Atan(0.5) = 26.6
CEIL	rounding up	$y = ceil(x)$	Ceil(1.1) = 2, Ceil(1.9) = 2
FLOOR	rounding down	$y = floor(x)$	Floor(1.1) = 1, Floor(1.9) = 1
ROUND	rounding to the nearest	$y = round(x)$	Round(1.1) = 1, Round(1.9) = 2
ABS	absolute value	$y = abs(x)$	Abs(-1.5) = 1.5, Abs(1.5) = 1.5
RAND	random number between 0 and x	$y = rand(x)$	Round(5) = 0 – 5 ?

Table 18 Mathematical functions.

Sin, Cos and Tan functions

The normal sin, cos and tan functions require tan argument in degrees (between 0 and 360, no radians). If you enter the argument value 370, the calculation is actually performed with value 10. If the value of the argument is –10, the calculation is performed with value 350. The calculations are always performed with values between 0 and 360, and the given value is converted to the right place between these values. The return values for sin and cos functions are between –1 and 1 and for tan functions between -∞ and ∞.

The corresponding arcus functions Asin, Acos and Atan are reversed. Asin and Acos require a value between –1 and 1 and return a value between 0 and 360 degrees. Atan functions can be given values between -∞ and ∞, and return values between –90 and 90 degrees.

Sinh, Cosh and Tanh functions

You may find use for hyperbolic functions as factors in certain calculations.

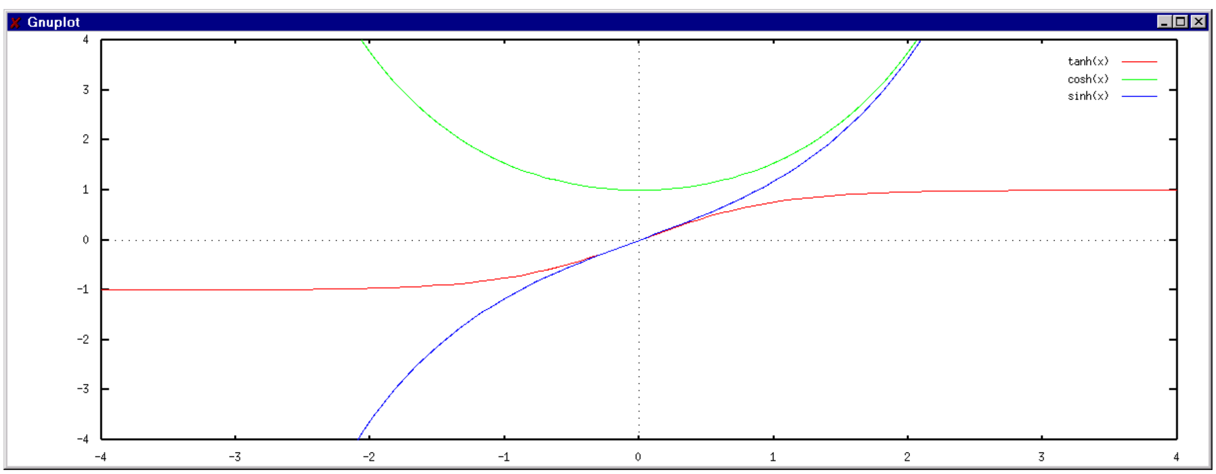


Figure 35 Values of hyperbolic functions between –4 and 4. The sinh curve is blue, the cosh curve green and the tanh curve red.

Ramp functions/masks (RU, RD and DD)

There are three different ramp functions or sliding factors for creating calculatory masks and soft changes in the edited data. They are RU (ramp up), RD (ramp down) and DD (double ramp).

Ramp functions can be used for gradient changes. Normally, the following mask:

IF(P < 1000)

T = T + 10

Produces abrupt changes in temperature in the borderline of areas where pressure drops below 1000 hPa (Figure 36).

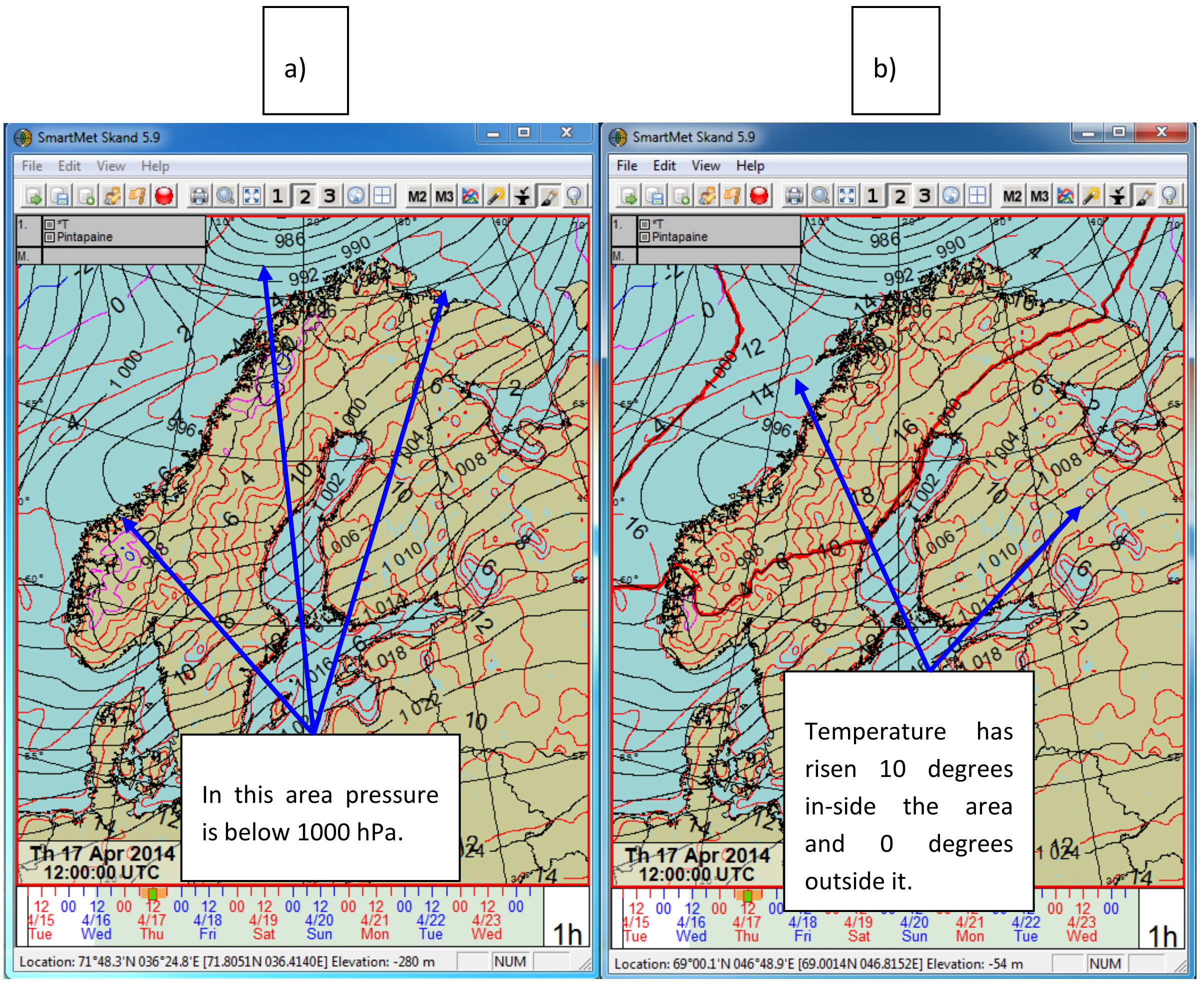


Figure 36 This This series of images illustrates how a simple mask can produce very unsatisfactory results. Figure a) shows the original situation with an area of less than 1000 hPa pressure in the middle. Figure b) shows a sharp borderline for changes in temperature.

If, instead, the factor is a ramp function, the change is considerably smoother and neater. The following formula increases the temperature by five degrees multiplied with a factor between 0 and 1. If the pressure is below 997 hPa, the factor is one, with pressures between 997 and 1000 the factor slides between 1 and 0, and when the pressure is above 1000 hPa, the factor is 0. In other words, the formula uses a ramp down mask, RD. Its parameters are the name of the meteorological parameter, its low limit and its high limit:

T = T + 10 * RD(P 997 1000)

No separate condition expression is needed, because if the ramp mask returns the value 0, no change occurs (Figure 37).

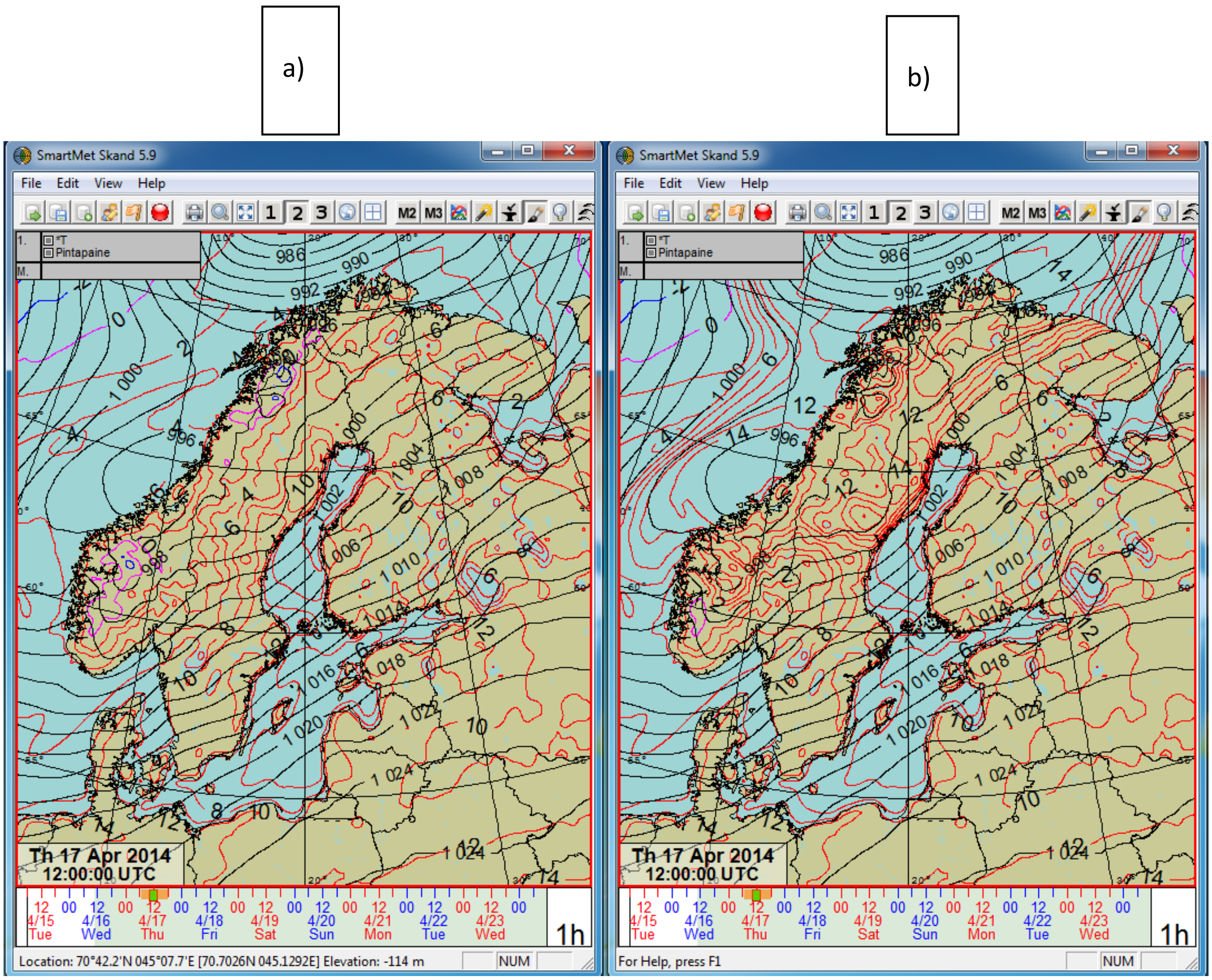


Figure 37 By using ramp masks you can create softer changes. Figure a) shows a modified temperature and Figure b) shows the change field.

Function	Description	Value below low limit	Change between limits	Value above high limit
RD	Ramp down	1	Decrease 1 -> 0	0
RU	Ramp up	0	Increase 0 -> 1	1
DD	Double ramp	-1	Increase -1 -> 1	1
-DD	Double ramp down (made using minus symbol)	1	Decrease 1 -> -1	-1

Table 19 Ramp functions.

NOTE! You cannot use any formulas as parameters in ramp masks. The parameters need to be constants like 998.5 and 1000.

Time functions (JDAY, LHOUR, FHOUR, MAXFHOUR)

The following time functions look like variables because no parameters are assigned to them (and no empty brackets are put behind them), but they return different values depending on the season and the time of day.

JDAY or Julian day, Day of year

The **JDAY** function always returns the Julian day, which means the number of days that have elapsed since the beginning of the year (according to the Gregorian calendar) at the calculated point of time. It can be used for defining a factor that affects strongly in winter and weakly in summer. In winter, **JDAY** returns values close to 0 and 360 and in summer values close to 180. **Cosine** returns its maximum value in winter and minimum value in summer. Furthermore, value 1 needs to be added to the expression to avoid having a negative factor in summer. The expression of the example is

T = T + 3 * (COS(JDAY) + 1)

This expression increases temperature by 6 degrees in mid-winter and 0 degrees in mid-summer and by a value between these over the rest of the year.

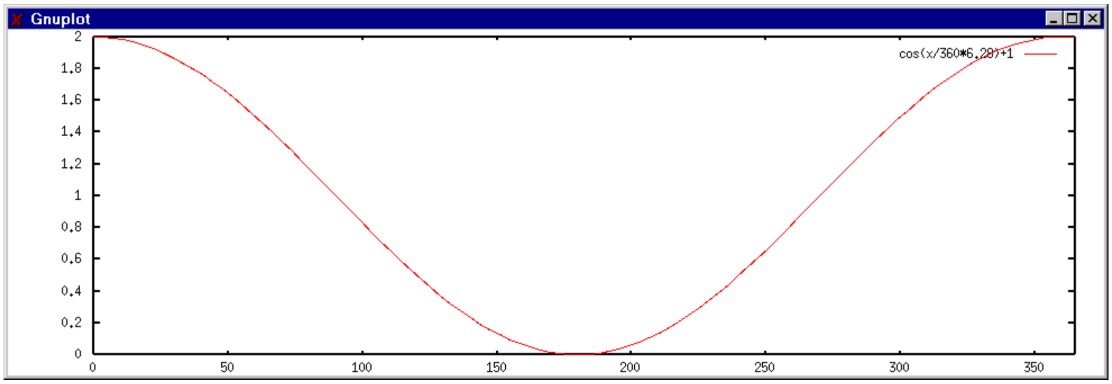


Figure 38 How the JDAY-function behaves with cosine (low values in summer and high values in winter).

If you want to have a reversed situation with a high factor in summer and a low one in winter, you can use a shift of 180 degrees:

T = T + 3 * (COS(JDAY + 180) + 1)

LHOUR or local hour

The LHOUR function returns the local time instead of the used forecast time (UTC). Forecasts are given in UTC time, but if you want to create changes that are strong at night and weaker during daytime, you can use the LHOUR function with sine or cosine functions. See the example in previous chapter.

The LHOUR function returns a value between 0 and 23. Local hour is calculated using the longitude and UTC time, meaning that national peculiarities and state borders are ignored. You need to multiply the figure you get from the LHOUR function by 15 if you want it to be useful with sine and cosine (360/24 = 15).

Forecast hour FHOUR ja MAXFHOUR

FHOUR (=Forecast HOUR) returns the forecast hour of the point of time. This is a value between 0 and **MAXFHOUR**. **MAXFHOUR** always returns the length of the forecast in hours.

Visual testing aid for time-dependent functions

You will probably often need a factor for a formula to modify for instance temperatures. For example:

T = T + 1 * factor

Here you also want the **factor** in the formula to be dependent on the time of the day. You can then test your formula by assigning its value directly to the parameter. For example:

T = 5 * cos(LHOUR * 15 - 180) + 5

You can then see the result in the time series dialog. When you think that the factor is behaving the way it should, assign it to the original formula. The formula in the figure requires a sine curve which uses factor 0 at midnight and factor 10 at noon.

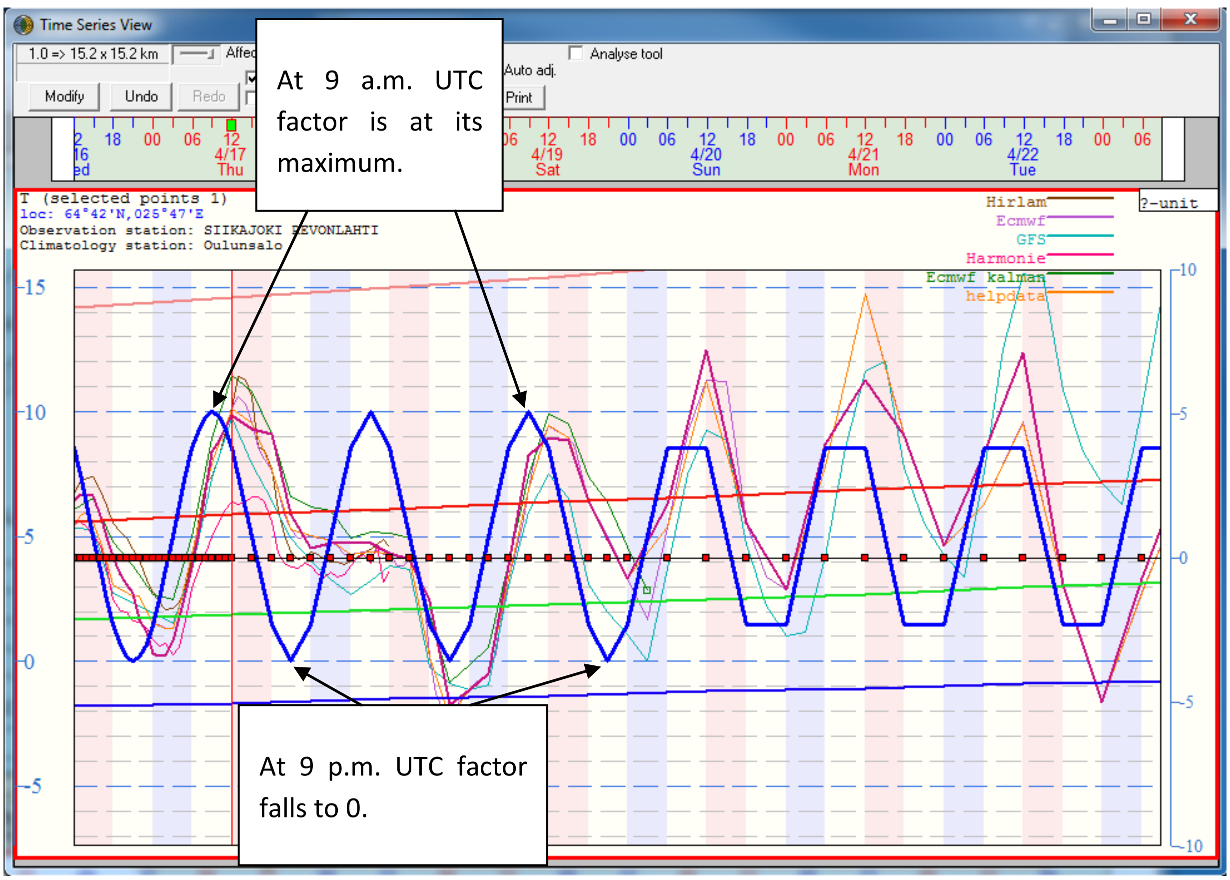


Figure 39 Test different factor statements by assigning them directly to the parameter.

9.3.8.8 Including other macros

Macros can also execute other macros by including them. Note! This inclusion command comes directly from C programming language, which explains the syntax used in it. The example consists of two macros, MACRO1.st and MACRO2.st. MACRO1.st has the following content:

```
// ****** MACRO1.st *******

IF(T > 5)

    T = T * 1.1
```

and MACRO2.st has the following content:

```
// ******* MACRO2.st *******

IF(T < 5)

    T = T * 0.9

#include “MACRO1.st”
```

Now MACRO2.st “reads” the text contained in MACRO1.st into itself before it is executed. This means that the real content of MACRO2.st is going to be:

```
// ******* MACRO2.st *******

IF(T < 5)

    T = T * 0.9

// ******* MACRO1.st *******

IF(T > 5)

    T = T * 1.1
```

NOTE! You cannot include a macro in itself, because this would cause the inclusion to happen recursively over and over again.

9.3.8.9 Comments

In order to make documented macros, it is necessary that you can comment on the macro language when you need to. In the macro language of the editor, you can use C++ language comments. You can add comments always at the end of a line or for a whole block. Comments for lines are identified with //, and comments for blocks are between /* and /*. Example:

```
// THIS IS A COMMENT

T = T + 1 // THIS IS A COMMENT

IF(T > 1)

        T = T + 2

/*

        THIS IS A COMMENT

*/

T = T + /* THIS IS A COMMENT */ 2
```

Comment symbols will not be confused with multiplication and division because they are omitted from the text before performing calculations.

10. Harmonizers

Harmonizers are SmartTool scripts that are used for creating dependencies between different parameters. If you modify different parameters, the editor keeps a record of what parameters have been modified. When you decide that it is time to harmonize the data, you can press a button and the editor deduces what kind of harmonization script it needs to produce and execute.

Harmonizers need to be prioritized and harmonization chains must be made so that they can only go downwards in prioritization.

Harmonizer dependencies could look like this (the actual statement is only given with nr. 1):

1. T, RH -> Td (No other harmonizers depend on this.)

var x = (log(RH/100.) + 17.27 * (T / (T + 237.3))) / 17.27

Dp = 237.3 * x/(1-x)

2. T -> PreForm (Nr. 3 depends on this.)
3. PreForm -> Vis (Nr. 4 depends on this.)
4. Vis -> rr (Nr. 5 depends on this. Is that sensible?)
5. rr -> Vis (Nothing depends on this. Nr. 4 has a higher priority.)
6. P -> Wd,Ws (Nothing depends on this.)

Case 1: You change temperature and press the harmonization button. The editor creates the script:

1 + 2 + 3 + 4 + 5

Case 2: You change visibility, Vis, and press the harmonization button. The editor creates the script:

4 + 5

Case 3: You change temperature, T, and precipitation, rr, and press the harmonization button. The editor creates the script:

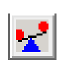
1 + 2 + 3 + 4 + 5

In this case editing rr may have been a waste of time, because it was already deduced using Vis.

NOTE! Dependencies are made manually so that when you create a harmonizer you determine which parameters it depends on and further on what other harmonizers each harmonizer depends on.

10.1 The principle for executing harmonizers

1. The editor keeps a record of which parameters and times have been changed.
2. The editor examines harmonization scripts to find out to which parameters assigments are made and concludes what harmonizers need to be included in each harmonization execution.
3. The user sets dependencies between harmonizers.
   - The editor does not execute a dependency that refers to a harmonizer with a higher priority.
4. The user needs to decide, when harmonization is executed. The options are “smart” and FORCE.
   - “Smart” execution analyzes what times and parameters have been executed and deduces what kind of harmonizer should be constructed. It is only executed for changed times.
   - FORCE marks all parameters and times “dirty” and then executes “smart” harmonization.

10.2 Harmonizer dialog 

You can open the harmonizer dialog either by pressing the button in the toolbar or by choosing Harmonizer dialog in the View menu (Figure 40).

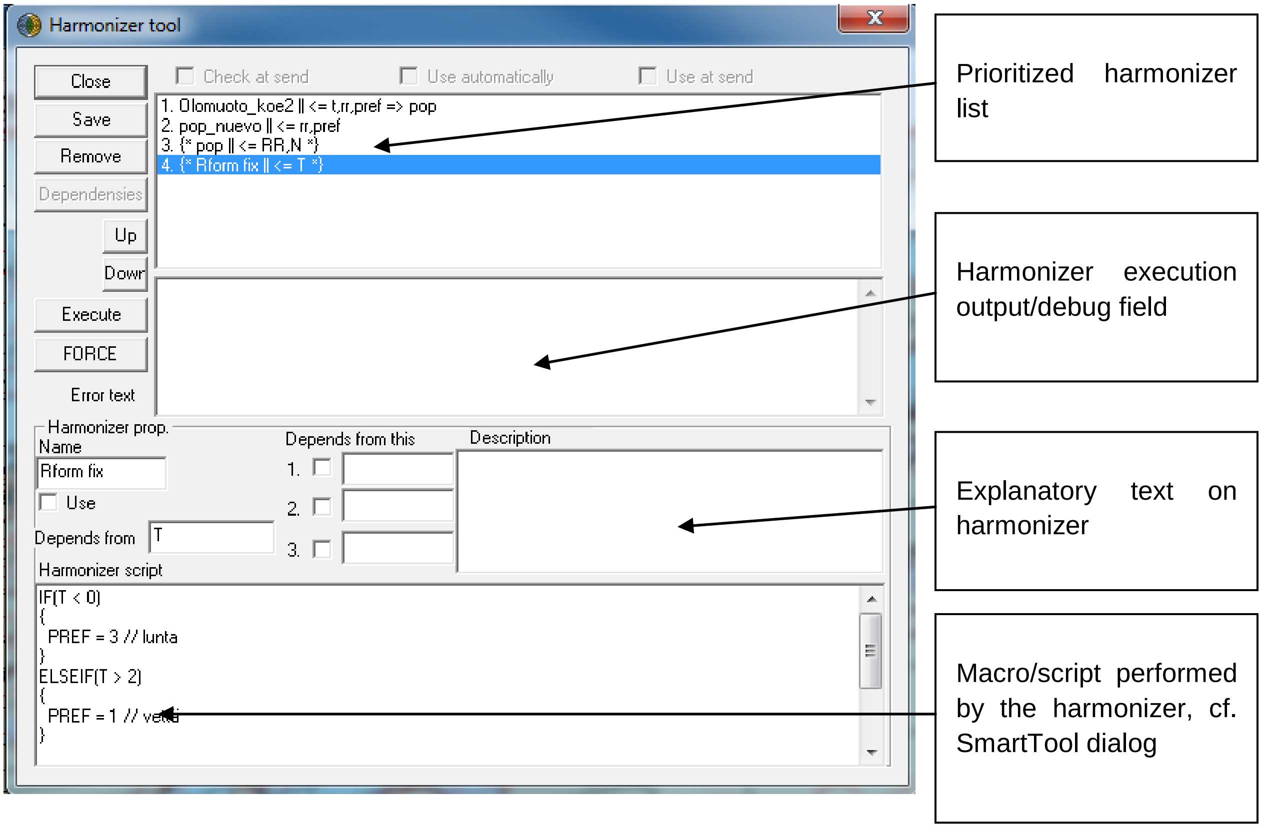


Figure 40 Harmonizer dialog

Creating a new harmonizer

Enter a new name in the name field of the harmonizer property section and modify the information you want. The most important information is the name, possible dependencies and, of course, the executed fix/check macro script. The macro script is written in the SmartTool language and subject to all the rules of the language.

When you have set all necessary properties for the harmonizer, it is recommendable to save it on your disk using the **Save** button. The macro cannot be applied before it has been saved for the first time.

A new harmonizer is always put last in the prioritization list, but you can move it anywhere you want.

Saving

When you create a new harmonizer or make changes to an old one, it is usually recommended to save them on disk. You can do this using the **Save** button. If you want to replace a previous file, the program will ask you if you are sure of this.

Closing the editor will also save all changes to harmonizers on your disk.

Modifying an old harmonizer

You can select on old harmonizer from the list with your mouse and modify it. The changes to an existing harmonizer are applied immediately, but they are not saved on disk until you press the **Save** button (or close the editor).

Removing

You can remove an old harmonizer. The editor will ask you, if you are sure that you want to remove the harmonizer. The harmonizer will also be deleted from the disk, and therefore it may sometimes be advisable to just disable the harmonizer.

Dependencies

There are two kinds of dependencies: dependencies of the executed harmonizers on the changed parameters and dependencies of harmonizers on other harmonizers.

Parameter dependencies

If you want the harmonizer to depend on a parameter, you enter the name of the parameter (for example T = temperature) in the field indicating what the harmonizer depends on. If the harmonizer depends on several parameters, they are separated with commas (for example T,RH,P).

Harmonizer dependencies

You should create a dependency between harmonizers, if some changes require further modification of another parameter.

For example, you may modify temperature and have a harmonizer that modifies the precipitation form based on the temperature. The editor notices the dependency and executes the harmonizer.

When changes in precipitation form are made, you should check visibility, because this can change when the form changes. To do this, you need to create a dependency between the precipitation form harmonizer and an appropriate visibility harmonizer. The visibility harmonizer in turn could have a dependency with a fog situation harmonizer, and so on.

You can set dependencies by naming a dependency from one harmonizer to another. In the present version, the maximum number of such dependencies is 3. Naming the harmonizer is not enough, though. You also need to check the checkbox of the harmonizer which activates the dependency.

Dependency does not work, if the priority of the dependent harmonizer is higher than that of the harmonizer it should depend on.

The harmonizer list also shows dependencies with other harmonizers indicated with the following symbol:

||

Arrows (=>) after the symbol || indicate which harmonizers depend directly on this harmonizer. (Dependencies derived from other dependencies are not listed!)

Changing priorities

You can change priorities with buttons that move harmonizers up and down. Select the harmonizer the prioritization of which you want to change, and press the buttons to move it up or down in the list.

Change in priority can affect the functionality of dependencies. If you move a harmonizer so that the dependent harmonizer is higher than the harmonizer from which the dependency begins, the dependency no longer works. (These situations are always noted in debugging text.)

Enable/disable

You can enable or disable a harmonizer with the **Enable** checkbox.

You should not remove harmonizers unnecessarily. Sometimes all you need to do is disable them.

Executions

The **execute** button starts a so-called “smart” execution, which inspects what times and parameters have been changed and deduces what kind of harmonizer to construct. It is only used for changed executions.

If you press the **FORCE** button, the harmonization will mark all parameters and times “dirty” and then execute a “smart” harmonization.

When the Harmonizer is executed, the editor writes an execution log (debugging text). You can read the log after the execution.

Inspecting the information on harmonizator execution

When you execute a harmonization, information on the operation will appear in the error text window.

Appendix 1 – Short cut Keys

** General map view short cuts **

- CTRL + A Selects all points

- CTRL + SHIFT + A Deselects points

- CTRL + B Toggles over map help images

- SHIFT + B Toggles over map help images over data (over/under)

- CTRL + C Copy active views active parameter values

- CTRL + D Toggle symbol spacing out mode

- CTRL + E Toggle data point markers

- CTRL + F Toggle used map

- CTRL + G Store active views active parameter values to file 'grid.txt'

- CTRL + H Show/hide parameter box

- CTRL + I Open view macro dialog

- CTRL + K Open harmonizer dialog

- CTRL + L Toggle regional weather text language (fi -> se -> en)

- CTRL + M Move parameter box

- CTRL + P Print map view

- CTRL + Q Toggle country border line color

- CTRL + SHIFT + Q Change country border line width

- CTRL + R Toggle projection line draw mode

- CTRL + S Store edited data to working data file

- CTRL + T Toggle time system draw mode

- CTRL + V Paste copied data to active views active param

- CTRL + W Toggle mask visualization
- CTRL + Z Undo data modification
- CTRL + Y Redo data modification
- < - key Move time backwards
- > - key Move time forwards
- UP - key move map row upwards
- DOWN - key Move map row downwards
- 1-5 Sets map row to selected
- F1 Opens short cuts dialog (this dialog)
- F2 Toggle observation comparison mode.
- SHIFT + F2 Set observation comparison mode symbol.
- CTRL + F2 Set observation comparison mode symbol size.
- CTRL + SHIFT + F2 Set observation comparison mode symbol border draw.
- F3 Open synop plot settings dialog.
- F4 Toggle cross section mode
- F5 Refresh all views
- F7 Toggle help cursors on map view on/off.
- F8 Execute harmonization
- CTRL + F8 Execute FORCE harmonization
- F9 Opens warning center dialog.
- F10 Keep correct map ratios on/off.
- F11 Tooltip on/off.

CTRL+SHIFT+wheel Changes textgen calculation range radius

** Mouse clicking on map view **

- Mouse clicking on map activates map row, view and time

- Left mouse click: selects modified points, clears old selection first

- CTRL + left mouse click: selects modified points, added to old selection

- SHIFT + left mouse click: removes from selected points

- mouse right click: selects point to displayed in timeserial view, clears old selection first

- CTRL + mouse right click: selects point to displayed in timeserial view, added to old selection

- SHIFT + mouse right click: removes from selected (timeserial diaplayed) points

** Time Series view dialog **

- right mouse click on cray areas opens parameter selection popup

** Time Series view **

- left mouse click adjust data modifying points

- left mouse click on value scale adjust the scale to smaller values

- right mouse click on value scale adjust the scale to bigger values

- CTRL + left mouse click on scale moves scale values downward

- CTRL + right mouse click on scale moves scale values upward

** parameter box **

- mouse right click on map row number opens param selection popup

- mouse right click on parameter opens parameter adjusment popup

** Mask selection box **

- mouse right click opens add mask popup

- mouse right click on mask opens mask adjustment popup

** Zoom view **

- left click centers zoomed area rect

- hold left mouse button to drag zoomed area rect

- right bottom corner of zoomed area rect is hot spot and you can drag the corner to change rect

- CTRL + left mouse click decreases zoomed area rect

- CTRL + right mouse click increases zoomed area rect

- CTRL + SHIFT + left mouse click maximizes zoomed area rect

** Time Series view dialog **

- right mouse click on cray areas opens parameter selection popup

** Time Series view **

- left mouse click adjust data modifying points

- left mouse click on value scale adjust the scale to smaller values

- right mouse click on value scale adjust the scale to bigger values

- CTRL + left mouse click on scale moves scale values downward

- CTRL + right mouse click on scale moves scale values upward

- left mouse click on modification scale decreases scale values

- right mouse click on modification scale increases scale values

- in views upper right corner you can change the modification unit (absolute/relative)

** Time Series view time control **

- left mouse click moves time control views start/end time backward

- right mouse click moves time control views start/end time forkward

- SHIFT + left mouse click moves start and end time backward

- SHIFT + right mouse click moves start and end time forward

- CTRL + SHIFT + left mouse click resets start and end time

** Data filter tool area modification **

- left mouse click moves modification rect (blue rect)

- right mouse click does nothing

- you can move the modification rect by dragging it with left mouse button

- CTRL + left mouse click decreases the modification rect size

- CTRL + right mouse click increases the modification rect size

- CTRL + SHIFT + left click decreases the modification grid size

- CTRL + SHIFT + right click increases the modification grid size

** Data filter tool time modification **

- left mouse click moves modification rect

- right mouse click does nothing

- you can move the modification rect by dragging it with left mouse button

- CTRL + left mouse click decreases the modification rect size

- CTRL + right mouse click increases the modification rect size

- SHIFT + left click moves modification rect to nearest hour

** Control point tool mode on map view **

- left mouse click activates the control point (CP)

- you can drag the active CP with left mouse button

- Right mouse click opens CP popup

- SHIFT + left mouse click enable/disable CP

- SHIFT + CTRL + left mouse click adds new CP

- DEL - key deletes active CP

** Cross section view **

- Left click ????

- Right click opens parameter selection popup

- CTRL + Right ????

- Mouse wheel up moves view row up

- Mouse wheel down moves view row down

- CTRL + Mouse wheel up increases vertical resolution

- CTRL + Mouse wheel down decreases vertical resolution

** Cross section view mode on (on map view) **

- Left mouse click sets start point

- Right mouse click sets end point

- Middle mouse button click sets middle point (if in 3-point mode)