arbonaut

Data Fusion System

User's Guide

10th April, 2019

Table of Contents

List of abbreviations and concepts

Alg@line data	Measurements on water quality from a certain ship. Includes line data (sensor measurements between 200 m) and bottle data (samples analysed later in laboratory.
EO data	Earth observation data, grid based remote sensing data to monitor and assess the status and changes in natural and built environments. In this project the EO data include data on water turbidity and chlorophyll-a.
SDFS	SYKE Data Fusion System, the system to be developed in this project
VESLA	Water quality part of the Environmental Information System of the Finnish Environment Administration. It includes physico-chemical measurement results of national and regional monitoring carried out by regional environment centres as well as local statutory monitoring results conducted by private companies and water protection associations. Data is available from lakes, rivers and coastal waters.
WCS	Web Coverage Service: OGC standardized interface for web-based retrieval of coverage, i.e. digital geospatial information on time/space-varying phenomena
WMS	Web Map Service: OGC standardized interface for web-based retrieval of georeferenced map images

1 Introduction

1.1 Purpose of the system

SYKE monitors sea and lake water quality by using different methods and instruments. Observations from satellites, commercial ships (ships-of-opportunity) and manual and automatic in situ sampling produce data with varying accuracy, precision and spatiotemporal coverage and representativeness. These observations need to be collated, harmonized and spatiotemporally interpolated to estimate (data fusion) ecological status of surface waters. There are gaps in SYKE's current data fusion system, which this new system intends to fill in to make it technically and operationally feasible for SYKE and for its clients.

The purpose of the Data Fusion System is the production of integrated water quality information and algal bloom forecasts for Finnish inland waters and for coastal sea areas. At the same time the system functions as an Open Source library that can be used for advanced data fusion and data assimilation tasks within SYKE in a wider sense in the future.

The system is targeted for an expert user that can conveniently carry out data fusions tests, validation and calibration tests, estimate the status of water bodies and produce algal bloom forecasts based on operational remote sensing and in-situ data available at SYKE, to design operational services or optimize the use of measurement resources.

1.2 Document purpose

This document is a user's guide for the Data Fusion System. The following chapters give an overview of the user flow (user process) in the Data Fusion System, provides examples as well as more detailed descriptions of the available commands in the system.

2 Overall description of the Data Fusion System

2.1 Data fusion process

SYKE's data fusion system combines sea and lake chlorophyll-a and turbidity data from different sources (such as remote sensing, manual in-situ, ships-of-opportunity and model simulations) and interpolate them into a regular spatiotemporal grid. The data fusion is mostly retrospective: the idea is to find the optimal estimate of the state of the system over the whole-time span of the observations. However, the results could be used to make predictions, too. The data fusion interpolation model is based solely on empirical information on the natural variability of chlorophyll-a, turbidity and auxiliary variables and on the measurement principles and processing of the observations. Thus, mostly the case of purely statistical estimation and interpolation are considered.

The data fusion process includes the following steps:

1. The implementation of harmonization procedure that
   a) define model domains for regions of interest
   b) acquires data from existing data sources (satellite (EO) data, manual in situ data, data from automatic stations (ferry or buoy) and ship-of-opportunity (Alg@line) data)
   c) transforms spatiotemporal coordinates and
   d) processes uncertainty of data (errors in measurements and sampling).

2. The implementation of generic state-space data fusion framework for spatio-temporal interpolation of information from available data sources.

    a. The data fusion takes the uncertainty of each measurement into account and produces both the best point estimate and the uncertainty in terms of standard deviation of the estimate error. The system has been tested on the chlorophyll-a and turbidity measurements available at the lake Säkylän Pyhäjärvi and the Porvoo coastal area sites. Covariate information such as nutrient concentration, temperature, salinity, etc. can be utilized by the data fusion algorithm, however the current system implementation does not read this information (with the exception of salinity and temperature provided by Alg@line). Directions on how to extend the data fusion algorithm to allow, for example, the use of covariate data are given in the "Data Fusiojn System – Maintenance Guide" document.

    b. The data fusion algorithm is implemented in open software tools such as Python, SciPy and NumPy to allow modifications and extensions to the system be done with relative ease.

    c. The algorithm uses various approximations following the best applicable knowledge in current science to allow also large spatial and temporal domains to be solved.

    d. Computation is accelerated by using highly optimized linear algebra implementation and takes advantage of available computing resources on a multi-core system.

2.2 User workflow in the Data Fusion System

Picrure 1 describes the usual user workflow in the Data Fusion System. On the beginning user defines the spatial area (polygon) for the analysis, so called Model domain (with command: manage_model_domain). User can use Model domain which already exists in the system of she can add a new one to the database. More detailed description for managing Model domains in the database is provided in chapter 4 Model domain management.

When the model domain is available in the database, the next step is to run Data harmonization (with command: calculate_data_harmonization). In Data harmonization the measurement data is read for the model Domain and harmonized (converted to the internal coordinate system of the database, adjusted for time and depth etc.) and then saved to the database. For more information on Data harmonizations see chapter 5 Data harmonization.

Picture 1. Process flow-chart.

After the Model domain and the observation data is processed and saved to the database the data fusion can be run (Command: calculate_data_fusion). After the data fusion is ready the results are saved to the database. The more detailed description of data fusion in Data Fusion System is given in chapter 6 Data Fusion.

The Data fusion results as well as source data can be viewed in QGIS. The DFS browser plugin for QGIS provides an easy to use user interface for the selection of the data from the data base (see chapter 7.2 Viewing observation datasets and 7.3.1 Adding result rasters to map for more details). Also the results time

series and results raster maps can be exported with DFS Browser (see chapters and for the details).

3 Accessing the Data Fusion System

The DFS tools and QGIS with DFS Browser plugin, are installed on specific Windows server. Users access the Data Fusion System (DFS) via remote desktop. The DFS system administrator has configured the database connection, and user should not have to worry about that. Administrator will provide instructions on how to access the remote desktop, e.g. user name and password. User should just

- Log in to the Windows server using remote desktop
- On the remote desktop, user has two different ways to access the system
- DFS tools
  - Access through command line
  - The administrator may have prepared a shortcut, that will prepare everything in command line console with opening just one link from e.g. Desktop
  - If you have no instruction for shortcut, then look for Anaconda Prompt form the desktop or Start menu. If they have no 'Anaconda Prompt', then select
    - Start Menu / All Programs / Anaconda 3 (64-bit) / Anaconda Prompt
    - In Anaconda command window, type *activate idp* and press <Enter>
    - Now the DFS command line tools are ready to be used
  - Each DFS tool creates a log file after it has run. Where the log file goes, can be specified on command line. By default they go to the current working directory.
- DFS Browser QGIS plugin
  - Access through the Quantum GIS desktop software, or QGIS
  - Start QGIS from Desktop or Start Menu, or if it is not there, select
    - Start Menu / All Programs / OSGeo4W / QGIS Desktop 3.X.Y
    - X,Y may vary depending on the installed version
  - Open the DFS Browser window in QGIS by clicking the icon
  - If the DFS Browser icon is not visible, the plugin needs to be activated. To do that, select Plugins / Manage and Install Plugins… from the main menu and make sure the "DFS Browser" plugin is ticked (enabled)

3.1 Common optional arguments for all tools

All tools implemented in the Data Fusion System accept the following optional parameters. For brevity, the parameters are listed here and not repeated later in this document.

```
-h, --help
```

Show help message and exit

```
--log-dir LOG_DIR
```

Directory where the log file is output (default: ['.'])

```
--log-level LOG_LEVEL
```

Logging level in the log file (default: ['INFO']) Possible values:
[DEBUG|INFO|WARNING|ERROR|CRITICAL]

```
--verbosity VERBOSITY
```
Verbosity level on console (default: ['INFO']) Possible values:
[DEBUG|INFO|WARNING|ERROR|CRITICAL]

4   Model domain management

For management of model domains, the Data Fusion System has a `manage_model_domain` command line command which can be run in Anaconda prompt.

Model_domain_management has three main actions - list, del and add:

`list`   List model domains in database

`del`   Remove model domains from database

`add`   Create new model domain in database

4.1   Model_domain_management add

The command `model_domain_management  add` is used to add a new model domain into the database. User must give the model domain geometry, i.e. water body polygon, in a WKT file or a WKT text string on the command line, and an EPSG spatial reference system code, i.e. the system where the WKT coordinates are represented in. User must also specify the cell size for the data fusion model grid already when defining the model domain. This is because the next step in the work flow, data harmonization, already converts the gridded observation data to the final cell grid. This data can then be used as such by the data fusion tool, no conversions needed.

A model domain WKT geometry can be produced in QGIS, for example. One way of doing it, would be

1. Install the *Get WKT* plugin in QGIS
2. Activate a polygon map layer
3. Use a selection tool to select one polygon
4. Apply the Get WKT tool by pressing the blue tool button with white text 'WKT' in it
5. A pop-up window will appear, that has a text starting 'MultiPolygon'
6. The Get WKT seems to produce always a multi-polygon, even if the geometry is a single-part polygon. User should somehow check it, and if it is already single-part, it can be converted to such, by simply replacing the word MultiPolygon by POLYGON and removing outermost parenthesis.

Note, that the geometry represented by the WKT must be single part polygon geometry. During the development, we ran into a situation, were QGIS was used to union the 10 test water bodies on the coastal area, and the combined geometry was transformed to a WKT geometry using QGIS. We noticed in some development testing, that data harmonization imported point observations only to a small North-West part of the whole coastal region. The problem seemed to be, that the Union + WKT combination produced WKT content looking like

POLYGON (((x1 y1) … (xN yN)) ((p1 q1) … (pN qN)))

which, in fact, is a multi-part polygon, and should read 'MULTIPOLYGON (((…'. Some time was spent trying to work around this, but it was decided to leave it to the user, to check, e.g. using a GIS program, that the model domain geometry is a proper single-part polygon. In the coastal (rannikko) case, there was a small

gap between the North-West part and the rest, about 50 meters. This was fixed by first buffering with positive radius, and then negative. The radii were somewhere around plus 70 m and minus 60 m.

If the user tries to add a model domain with a name that already exists in the database, the tool will fail show the user the related database uniqueness constraint exception. This is not serious, just pick another name or remove the old model domain first. Just remember that when removing a model domain, all other database objects related to the model domain are also removed, e.g. all observations for the removed model domain.

`model_domain_management add` has the following optional arguments:

`--model-domain MODEL_DOMAIN`

Name of the model domain in the database

`--wkt-file WKT_FILE`

Text file containing WKT polygon geometry

`--wkt-geometry WKT_GEOM`

WKT polygon geometry

`--srid SRID`

Spatial reference system of the geometry

`--cell-size-x CELL_WIDTH`

Cell width (m)

`--cell-size-y CELL_HEIGHT`

Cell height (m)

4.2 Model_domain_management del

`model_domain_management del` command can be used to delete a model domain from the database. Deleing the model domain deletes also all the results and the observation data linked to the model domain.

`model_domain_management del` has one mandatory argument:

`--model-domain MODEL_DOMAIN`

Name of the model domain in the database

4.3 Model_domain_management list

`model_domain_management list` action is used for listing the existing model domains in the database. It does not have any mandatory arguments.

`model_domain_management list` has no optional arguments (besides those mentioned in Section 3.1):

8

4.4 Model domain management examples

Add a new model domain to the database with name Rannikko_500 from WKT- file. the geometry in WKT-file is in 3047 projection and use 500m x 500m cell size:

```
> C:\Users\pasiroti>model_domain_management add --model-domain Rannikko_500
--wkt-file "Rannikko_union_single_part_trunc_to_1m.epsg3047.wkt" --srid 3047
--cell-size-x 500 --cell-size-y 500
```

List existing model domains in the database:

```
> C:\Users\pasiroti>model_domain_management list
```

produces output similar to:

```
Juhani_Pyhäjärvi_100
Juhani_Pyhäjärvi_200
Juhani_Pyhäjärvi_300
Juhani_Pyhäjärvi_500
Juhani_Rannikko_100
Juhani_Rannikko_250
Juhani_Rannikko_500
etc.
```

Delete model domain Rannikko_500 and all data related to it:

```
> C:\Users\pasiroti>model_domain_management del –model_domain Rannikko_500
```

5 Data harmonization

In data harmonization the observation datasets are read and transformed to the format that can then be used for data fusion i.e. observation data is harmonized.   In harmonization process the coordinate system of the source data is reprojected to srid:3067 (ETRS-TM35FIN) coordinate system. Also, the adjustments for observation time within a day and measurement depth (point observations) are performed.

Each external data source and the related import process is pre-configured in the data fusion database by the system administrator of the Data Fusion System. The currently available external sources are

- *Vesla_raw* (import all VESLA data, Chlorophyll or Turbidity, no filtering or processing, default uncertainty SD = 0)
- *Vesla_rannikko* (import according to coastal region rules in the project specifications)
- *Vesla_rannikko_Manual* (as Vesla_rannikko but import only Manual station type)
- *Vesla_rannikko_AutoSensor* (as Vesla_rannikko but import on AutoSensor station type)
- *Vesla_pyhajarvi* (import according to Pyhäjärvi rules in the project specifications)
- *Vesla_pyhajarvi_Manual* (as Vesla_pyhajarvi but import only Manual station type)
- *Vesla_pyhajarvi_AutoSensor* (as Vesla_pyhajarvi but import on AutoSensor station type)
- *Alg@line_linjadata* (Alg@line sensor data imported from a CSV file provided by SYKE)
- *Alg@line_pullodata* (Alg@line bottle data imported from a CSV file provided by SYKE)
- *EO_rannikko* (remote sensing data imported from EO service WCS coverages MER_FSG_L3_rp2017_chla and MER_FSG_L3_rp2017_turb)
- *EO_pyhajarvi* (remote sensing data import from EO service WCS coverages PQD_EO_MR_LAKES_CHL and PQD_EO_MR_LAKES_TURB)

5.1 Managing imported dataset definition

5.1.1 Adjustment for the time period

User decides which time period is imported. The time range is specified with two command line parameters `--start-date` and `--end-date`. The dates are specified after the parameter in ISO date format YYYY-MM-DD, e.g. `--start-date 2009-04-01 --end-date 2009-09-30` would be used to import data between beginning of April and end of September.

If user likes, long period can be specified for the import, and the data fusion calculation can use small parts in different fusion calculation runs, or even the whole import season. Other option would be to import once per data fusion.

5.1.2 Management of uncertainties

The data fusion calculation needs information on how accurate the provided observations are. The DFS system takes the input as the statistical *standard deviation* (SD). They can be provided as absolute values or relative values, which means that the uncertainty SD depends on the magnitude of the observed value.

The VESLA data service provides *measurement uncertainty* (MU, 95% level of confidence) for some observations. SD is calculated from MU by SD = 0.5 * MU. For everything else, there are two ways provided in the system for specifying uncertainty SD:

- The pre-configured external data source definition lists default uncertainty SD. The specification format varies, and for some sources you can have general overall SD and per quantity, or even per processing rule uncertainty SD.

- The command line parameter `--default-uncertainty` can be used to override the pre-configured uncertainty SD's. The uncertainty SD coming from VESLA service will always be used, if it exists for an observation.

The uncertainties can change according to the harmonization process rules, and finally the uncertainty SD will be stored in the DFS database. For point data, the final storage of uncertainty SD is in the the same record with the observation value, and that uncertainty SD is already numerical, because the import has the numerical value which it is bound to, even if uncertainty SD specification was relative. With raster data, however, the uncertainty SD is stored in the dataset record referred by the raster blocks. It will be stored as relative SD if that was the specification, because it needs to be bound to each pixel value separately, and this cannot be done by data harmonization in the import stage, but it the final binding is done when the data fusion reads the pixels.

5.2 Point data harmonization

5.2.1 Management of uncertainties

For point data, uncertainties are managed just as described above. But harmonization rules are used to 'harmonize' or normalize the observation values and uncertainties. The adjustment can be done based on attributes like the water sampling depth (filtering of observations in VESLA) and time of year or time of day (harmonization of observation and uncertainty in both VESLA and Alg@line). The exact harmonization process is defined in the configuration of each external data source. The details are explained in the technical system description and the maintenance guide.

5.2.2 Adjustment of the measurement time within a day and different measurement depth

The samples taken at different time of the day are tuned to better correspond to the datafusion modelling time (0900 UTC). Similarly, observation and uncertainty SD can be tuned based on the sampling depth in the name of normalization. This sampling depth correction can vary by month. These things are pre-configured in the data source.

5.2.3 Adjustment of the measurement depth

Processing rules can be used to imitate a sample that is aggregated on the field, by taking an average from specific samples from different depths in the same location and at the same time. In coastal waters the mean of Chl-a measured at 1, 5 and 10 m is calculated if composite Chl-a observation is missing in VESLA data. In lakes 1 m Chl-a measument is used in case of missing composite Chl-a observation. This is pre-configured in the data source.

5.3 Running the data harmonization

Data harmonization is run in the Anaconda prompt. In the Anaconda prompt, calculate_data_harmonization command is used to run the data harmonization.

5.3.1 Import and harmonize the observation data

Calculate_data_harmonization command Imports and harmonizes data inside the given model domain and time frame for the given quantities from the given external data sources. The mandatory parameters for calculate_data_harmonization command, are:

```
--model-domain MODEL_DOMAIN
```
     Model domain name (default: None)

```
--start-date START_DATE
```
Start date (ISO format YYYY-MM-DD) (default: None)

```
--end-date END_DATE
```
End date (ISO format YYYY-MM-DD) (default: None)

```
--ext-sources SRC [SRC ...]
```
External data sources (default: None)

```
--quantities QTY [QTY ...]
```
Quantities (only those in sources are imported) (default: None)

5.3.2 Optional data harmonization parameters

```
--default-uncertainty SD
```
Uncertainty for raster datasets or point observations that have none.

This overrides the 'default_uncertainty_sd' configured in external source and updates the uncertainty_sd of related imported datasets in data fusion DB. SD must be either a number (e.g. 0.34), a relative SD (e.g. 12%) or empty '', which is the same as not giving it and results in using only the external source uncertainty for a raster dataset and to fill in any missing point data SD's. (default: [None])

5.4 Data harmonization examples

Harmonize Vesla_rannikko, turbidity and clorophyl from alg@line pullo data and EO data for Model domain Rannikko_500 with observations between April 1, 2011 and June 30, 2011 and set verbosity level to INFO.

```
(base) C:\Users\pasiroti>calculate_data_harmonization --verbosity INFO
--model-domain Rannikko_500 --start-date 2011-04-01 --end-date 2011-06-30
--ext-sources Vesla_rannikko Alg@line_pullodata EO_all --quantities TURB CP
```

The same as in previous example but in addition the default uncertainty is set to 5%, overriding whatever default uncertainty SD's are configured for the imported ext. sources:

```
(base) C:\Users\pasiroti>calculate_data_harmonization --verbosity INFO
--default-uncertainty 5% --model-domain Rannikko_500
--start-date 2011-04-01 --end-date 2011-06-30
--ext-sources Vesla_rannikko Alg@line_pullodata EO_all
--quantities TURB CP
```

6 Data fusion

6.1 Implemented algorithms

The Data Fusion System contains two implementations of a Kalman Smoother for state space estimation:

1. (stochastic) Ensemble Kalman Filter (EnKS)
2. Error Subspace Transform Kalman Smoother (ESTKS)

6.2 Logarithmic vs linear scale

By default, data fusion is performed in logarithmic scale in order to prevent negative values for the estimated quantities and to account for the heterogeneity of the model error (i.e. model error is expected to be larger for larger estimated values). When using logarithmic scale, the model error (the `--modelerror` parameter) is assumed to be in logarithmic scale as well. It is possible to revert to the linear scale by including the `--linscale` flag, its use is however not recommended as it generally leads to worse results.

6.3 Accounting for model and data uncertainty

The Kalman filter framework allows both the model uncertainty and observation data uncertainty to be accounted for. In both cases, the underlying assumption is that the model and observation errors are Normally-distributed random variables with zero mean and given covariance (equivalently, one can consider the system state and observations to be random variables with Gaussian distribution with specific mean and covariance). When the system state and observations are transformed to the logarithmic scale (which is the default system behavior), the distributions of the state and observation errors are in effect log-normal in the original scale.

The uncertainty in the system state is the product of 1) the uncertainty in the initial state and 2) the uncertainty due to additive model error (sometimes called process noise) - i.e. the inability of the model to track the true system state from one time step to the next. Because in the current implementation the model does nothing (it is an identity operator), the additive model error term thus defines the "magnitude" of the random walk. In both cases, the uncertainty is implemented by perturbing each ensemble member by a random Gaussian noise with covariance given by the covariance function

$$C(h) = \begin{cases} \sigma\left[1 - \left(\dfrac{3}{2}\dfrac{h}{l} - \dfrac{1}{2}\left(\dfrac{h}{l}\right)^3\right)\right], & \text{if } h < l \\ 0, & \text{otherwise} \end{cases}$$

where σ is the standard deviation of the normal distribution and l is the correlation length. This corresponds to the variogram function $2\gamma(h)$ through the relation

$$2\gamma(h) = C(0) - C(h)$$

In case of the initial state, the random Gaussian noise is applied to the initial ensemble while the additive model error is applied to the ensemble after it is propagated from current time step to the next. Parameters for both distributions can be given separately.

The uncertainty in observation data, due to measurement errors, modelling errors etc., is similarly represented by a covariance matrix that defines both the variance of the observation error as well as covariance between individual measurements. In the current implementation, the covariance matrix is assumed to be diagonal with observation error variance on the diagonal and zero (no covariance) elsewhere. In other words, the observations errors are assumed to be uncorrelated, allowing the covariance to be stored in sparse form.

6.4 Executing data fusion

The data fusion process is executed via the `calculate_data_fusion` executable available in the Python environment. The executable works in two modes – "check" and "calc". The parameters to both modes are identical and the "check" mode can be used to verify parameter values and the overall data fusion setup without actually running the data fusion. It can be used, for example, to see which observation data is going to be used, including the temporal coverage of available measurements of each observation dataset.

In summary, the "check" mode performs identical actions to the "calc" mode except that the execution is stopped immediately before the data fusion starts.

Similarly to the remaining tools in the SDFS toolset, the tool prints usage instructions to the console when executed with the single argument `--help`:

```
> calculate_data_fusion --help
```

The data fusion process can be cancelled by pressing Ctrl-C while the console window is active. The result record in the database will remain but its status will be updated to "Cancelled". Please note that the check for cancellation cannot for technical reasons be made within the linear algebra Fortran routines and the Ctrl-C press may therefore take a moment to be registered.

6.5 Data fusion parameters

Parameters accepted by the `calculate_data_fusion` tool are listed below.

6.5.1 Required parameters

`--model-domain <name>`

Name of the model domain for which the data fusion is to be calculated.

`--description <text>`

Arbitrary description of the data fusion result.

The description can be used to locate the result in the DFS browser QGIS plugin. If not given, generic description is generated based on other supplied parameters.

`--ext-sources <source> [<source> …]`

Observational data to be used in the data fusion.

The parameter lists the names of external sources, defined in the SDFS database, for which corresponding observational data should be used I the data fusion. Multiple external sources can be given as a list of names separated by space, for example:

```
--ext-sources EO_all Vesla_rannikko Algaline_linjadata
```

Additionally, the special value '*' can be used to specify that data from all external sources, which is available for the model domain, should be used. Therefore, one may conveniently write:

```
--ext-sources *
```

`--quantities <quantity> [<quantity …]`

Names of quantities to include in the data fusion.

The names can be specified both in long and short format – as an example, the quantities "Klorofylli-a" and "CP" are identical. Multiple quantities can be given as a list of names separated by space:

```
--quantities CP TURB or
--quantities Klorofylli-a Sameus
```

Similarly to the external sources, the special value '*' can be used to specify that all quantities for which at least some data is available for the model domain should be used:

```
--quantities *
```

`--start-date YYYY-MM-DD`

Starting date of the data fusion window given in YYYY-MM-DD format.

`--end-date`

Ending date (inclusive) of the data fusion window given in YYYY-MM-DD format.

`--initmean expression [expression …]`

Mean value of the initial state.

The value can be given in the following format (*<n>* denotes a number, *<quantity>* denotes a name of a quantity, either in short or long format):

- *<n>* - The same value is assumed for all quantities
- *<quantity>=<n>* - The value *n* is assigned to the specified quantity

Multiple expressions can be given, separated by space. The definitions are applied in the order they are listed with definitions later in the list taking possibly re-defining those that occurred earlier. Therefore, following specifications are possible:

`--initmean 10` – Assigns value 10 to all quantities

`--initmean CP=10 TURB=5` – Assigns 10 to the "Klorofylli-a" quantity and 5 to the "Sameus" quantity

`--initmean 10 TURB=5` – Assigns 10 to be the "default" value for all quantities and redefines 5 to be the value for the "Sameus" quantity

If any of the expressions contains spaces, it must be enclosed in double quotes (""), otherwise the parameter parser will treat it as two distinct parameter values, very likely resulting in a syntax error.

Please note that the initial mean is given on the scale of the data fusion, i.e. logarithmic by default and linear if `--linscale` is used. Therefore, initial mean of 1.0 on logarithmic scale corresponds to $e^{1.0}$ ≅ 2.72 on the original linear scale. If no `--initmean` value is given, 0.0 is assumed for all quantities and dates if logarithmic scale is used and 1.0 if linear scale is used.

`--initerror expression [expression …]`

Standard deviation of the uncertainty of the model state at the initial time step.

This corresponds to the σ term of the covariance function described in section "Accounting for model and data uncertainty". The format of the expression is identical to the `--initmean` parameter.

Please note that the model error is given on the scale of the data fusion, i.e. logarithmic by default and linear if `--linscale` is used.

`--modelerror expression [expression …]`

Additive model error term applied to the ensemble at each time step.

This corresponds to the σ term of the covariance function described in section "Accounting for model and data uncertainty". The format of the expression is similar to the `--initerror` parameter but with possibility to also specify a range of dates (*<date>* denotes a date specification in ISO format YYYY-MM-DD):

- *<n>* - The same value is assumed for all quantities
- *<quantity>*= *<n>* - The value *n* is assigned to the specified quantity
- `@(`*<date>*`,`*<date>*`)=`*<n>* - The value *n* is assigned to all quantities for days between the given dates (both inclusive)
- *<quantity>*`@(`*<date>*`,`*<date>*`)=`*<n>* - The value *n* is assigned to selected quantity for days between the given dates (both inclusive)

Multiple expressions can be given, separated by space. The definitions are applied in the order they are listed with definitions later in the list taking possibly re-defining those that occurred earlier. Therefore, following specifications are possible:

`--modelerror 1` – Assigns value 1 to all quantities and all days

`--modelerror CP=1 TURB=0.5` – Assigns 1 to the "Klorofylli-a" quantity and 0.5 to the "Sameus" quantity for all days

`--modelerror 1 TURB=0.5` – Assigns 1 to be the "default" value for all quantities and redefines 0.5 to be the value of the "Sameus" quantity

`--modelerror CP=1 TURB=0.5 CP@(2018-05-01,2018-07-31)=1.5` – Assigns 1 and 0.5 to be the "default" values "Klorofylli-a" and "Sameus" quantities and then redefines the value for "Klorofylli-a" to 1.5 between May 1st and July 31st 2018 (both inclusive)

It is possible to leave one of the date specifications out, leading to a half-open interval (the comma separating the two must however be included!):

`--modelerror CP=1 TURB=0.5 CP@(2018-07-31,)=1.5` – Assigns 1 and 0.5 to be the "default" values "Klorofylli-a" and "Sameus" quantities and then redefines the value for "Klorofylli-a" to 1.5 after and including July 31st 2018

`--modelerror CP=1 TURB=0.5 CP@(,2018-07-31)=1.5` – Assigns 1 and 0.5 to be the "default" values "Klorofylli-a" and "Sameus" quantities and then redefines the value for "Klorofylli-a" to 1.5 until and including July 31st 2018

If the specification does not cover all quantities and days between `-start-date` and `-end-date`, the tool will report an error when the first missing value is encountered.

Please note that the model error is given on the scale of the data fusion, i.e. logarithmic by default and linear if `--linscale` is used.

`--corlength expression [expression …]`

Correlation length in metres.

This corresponds to the l term of the covariance function described in section "Accounting for model and data uncertainty". The format of the expression is identical to the `--modelerror` parameter. Therefore, it is possible to specify different correlation lengths for different quantities and date intervals.

6.5.2 Optional parameters

`--exclude-stations id [id …]`

List of stations to exclude.

When given, point observation data from matching stations is excluded from the data fusion. The match is based on the station ID attribute. Multiple stations can be specified as a list of values separated by space:

```
--exclude-stations 1234 5678
```

`--exclude-station-types type [type …]`

List of station types to exclude.

When given, point observation data from matching stations is excluded from the data fusion. The match is based on the station type attribute and both the long and short station type name can be used. Multiple types can be given as a list of values separated by space:

```
--exclude-station-types Manual V
```

The option above excludes data from "Manual" and "Virtual" stations.

`--exclude-waterbodies id [id …]`

List of waterbodies to exclude.

This is functionally similar to the `--exclude-stations` option but the match is based on the waterbody ID attribute.

`--edit-station-error expression [expression …]`

Explicit uncertainty assigned to station observations.

This allows the uncertainty associated with certain measurement station to be altered directly when executing data fusion. The format of the expression is similar to the `--initerror` parameter but with possibility to also specify a station by its ID (*<id>* denotes a numeric ID of a measurement station):

- *<n>*[%]   - The same value is assumed for all quantities and stations.
- *<quantity>*= *<n>*[%]   - The value *n* is assigned to the specified quantity
- @*<id>*=*<n>*[%]   - The value *n* is assigned to all quantities for measurement station with given ID
- *<quantity>*@*<id>*=*<n>*[%] - The value *n* is assigned to selected quantity for measurement station with given ID
- If '%' is given after the value, *n* is interpreted as a relative uncertainty in percent

Multiple expressions can be given, separated by space. The definitions are applied in the order they are listed with definitions later in the list taking possibly re-defining those that occurred earlier. Therefore, following specifications are possible:

```
--edit-station-error  0.1
```
– Assigns uncertainty value of 0.1 to all quantities and stations

```
--edit-station-error  10%
```
– Alters the uncertainty to 10% the original value for all quantities and stations

`--edit-station-error CP=10%` – Alters the uncertainty to 10% the original value for 'Klorofylli-a' on all stations. Any other observations are left unaffected

`--edit-station-error CP=10% TURB=0.1` – Alters the uncertainty to 10% the original value for 'Klorofylli-a' on all stations and sets uncertainty of 'Sameus' to 0.1 on all stations. Any other observations are left unaffected

`--edit-station-error CP@1234=10%` – Alters the uncertainty to 10% the original value for 'Klorofylli-a' on station with ID 1234. Any other observations are left unaffected

`--ensemble-size number`

The number of individual state vectors that are used to approximate the error covariance.

The value should be a whole number between 30 and few hundred. Smaller ensembles may suffer from sampling noise, especially when using the ENKS algorithm and when the correlation length is long. If no value is provided, the ensemble size of 40 members is used.

Please note that the default value of 40 is a somewhat low-quality setting for fast execution time. Depending on the size of the model domain and the smoothness resp. roughness of the modelled fields, you may want to select a higher value to avoid accumulation of noise in the result. Allowed values are in the range from 5 (extremely low quality, fast) to 1000 (slow and memory-intensive).

`--ks ENKS|ESTKS`

Kalman Smoother implementation to use.

Allowed values are "ENKS" for the Ensemble Kalman Smoother and "ESTKS" for the Ensemble Subspace Transform Kalman Smoother. If no value is provided, the ESTKS algorithm is used.

`--covinf number`

Additional multiplicative covariance inflation term.

The default value is 1.0 (no inflation).

`--smlag number`

Smoother lag as the number of time steps.

Because both the EnKS and ESTKS implementations are currently fixed-interval smoothers, the lag parameter is ignored.

`--smdeflation number`

Smoother cross-correlation deflation factor.

The deflation factor is used for reducing the influence of temporal cross-correlation between two time steps during the smoothing stage. It is valid for both the EnKS and ESTKS algorithms.

Due to the limitations of all Kalman Filters (assumption of linearity and Gaussian errors), the smoother corrections further in the past may be degraded by the presence of sampling noise and the fact that the system dynamics is becoming strongly nonlinear. Therefore, the smoother covariance deflation factor is a practical tool for eliminating long-range (temporal) correlations that are nevertheless judged not to be physically justified.

The deflation factor typically ranges from 0.5 to 0.98 depending on how strongly nonlinear the system is and the time step. The default value is 0.9.

```
--max-grid-obs number
```

The maximum number of observations extracted from a gridded dataset when using the EnKS algorithm.

This is a quality setting allowing the EnKS algorithm to run faster at the cost of reduced accuracy. Since EnKS performs analysis in the observation space, the computational cost is directly proportional to the number of observations being included. If the number of observations (cells) in a gridded dataset is larger than this limit, the observations are systematically pruned by using a two-dimensional sparse grid. The origin of the grid is randomized at each time step. Please note that the limit is applied separately per each gridded data source.

This setting has no effect when using the ESTKS algorithm. If not given, the default value is 50000.

```
--batch-size number
```

Number of observations assimilated in a batch when using the EnKS algorithm.

This is a performance tuning option that can be used to improve performance for the given hardware. It does not affect the quality of the result.

This setting has no effect when using the ESTKS algorithm. If not given, the default batch size of 200 observations is used.

```
--block-size number
```

Size of a processing block (in cells) when using the ESTKS algorithm.

This is a quality setting that allows trading accuracy for speed as larger blocks will run significantly faster but may eventually become visible. The lowest allowed value is 1 (each grid cell is run separately), yielding highest quality result.

This setting has no effect when using the EnKS algorithm. If no value is given, the default block size is 3.

```
--linscale
```

Enables computation in linear scale.

If given, the data fusion is performed in linear scale instead of the default logarithmic scale. Use of this setting is discouraged.

6.6 Data fusion examples

The data fusion setup, including information about datasets that will be used can be checked with the "check" mode. It is recommended to supply all parameters in their final form to "check" and once all is correct, simply run the same with "calc".

```
> calculate_data_fusion check --model-domain Pyhajarvi --ext-sources *
--quantities * --start-date 2009-05-01 --end-date 2009-10-30
--modelerror 0.2 --corlength 10000 --smlag 5 --smdeflation 0.9
```

In the example above, all external sources and all quantities, for which data was imported into the model domain, are selected to be used via the '*' argument. The tool will print the data that will be used before exiting:

```
> Observation datasets to process:
> ID: 38    source: EO_all            quantity: Klorofylli-a    meas. period: 2011-05-01 - 2011-10-29
> ID: 46    source: EO_all            quantity: Sameus          meas. period: 2011-05-01 - 2011-10-29
> ID: 39    source: Vesla_rannikko    quantity: Klorofylli-a    meas. period: 2011-05-03 - 2011-10-26
> ID: 48    source: Vesla_rannikko    quantity: Sameus          meas. period: 2011-05-02 - 2011-10-27
> ID: 49    source: Alg@line_linjadata quantity: Klorofylli-a   meas. period: 2011-05-20 - 2011-08-15
> ID: 51    source: Alg@line_linjadata quantity: Sameus         meas. period: None - None
```

Similarly, we can restrict data fusion to only some external sources or quantities by explicitly listing them in `--ext-sources` and `--quantities`:

```
> calculate_data_fusion check --model-domain Pyhajarvi --ext-sources EO_all
Vesla_pyhajarvi --quantities CP --start-date 2009-05-01 --end-date 2009-10-
30 --initerror 0.5 --modelerror 0.2 --corlength 10000
--smdeflation 0.9
```

Executing the data fusion for the Pyhäjärvi lake, using a model domain created for the purpose, model error of 0.2 (log scale, applied to all quantities and days) and smoother deflation factor of 0.7. The remaining settings are left at their default values:

```
> calculate_data_fusion calc --model-domain Pyhajarvi
--description "Example 1" --ext-sources * --quantities CP TURB
--start-date 2009-05-01 --end-date 2009-10-30
--initerror 0.5 --modelerror 0.2 --corlength 10000 --smdeflation 0.9
```

Executing the data fusion for the Rannikko area, using similar settings as in the example above but excluding point observations from waterbodies 1 and 2:

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 2" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30 --initerror 0.5
--modelerror 0.2 --corlength 10000 --smdeflation 0.9
--exclude-waterbodies 1 2
```

The filtering expressions can be combined and observations can be excluded based on multiple criteria, such as the station type and station ID. In the example below, data from all manual stations *and* data from stations 1234 and 5678 is not going to be used:

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 2" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30 --initerror 0.5
--modelerror 0.2 --corlength 10000 --smdeflation 0.9
--exclude-station-types M --exclude-stations 1234 5678
```

The parameters of the initial and model error covariance functions can be given separately for individual quantities:

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 2" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30 --initerror CP=0.5 TURB=0.2
--modelerror CP=0.25 TURB=0.1 --corlength CP=10000 TURB=5000
--smdeflation 0.9
```

If any of the `−initmean`, `--initerror`, `--modelerror` or `--corlength` expressions contains spaces, it must be enclosed in double quotes, such as in

```
     --initerror "TURB = 0.2" "CP = 0.5"
```

Without quotes, the parameter parser will treat the expression as two distinct parameter values, very likely resulting in a syntax error.

Additionally, the error specification can be refined for a range of dates. In the example below, the error and correlation length is first defined for the entire duration and then redefined for the days between May 1st 2018 and July 31st 2018 (both inclusive). Since all specifications that apply for a given day and quantity are applied in the order they are defined, the least-specific expressions should be listed first.

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 2" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30
--initerror CP=0.5 TURB=0.2 CP@(2018-05-01,2018-07-30)=0.3
--modelerror CP=0.25 TURB=0.1 CP@(2018-05-01,2018-07-30)=0.15
--corlength CP=10000 TURB=5000 CP@(2018-05-01,2018-07-30)=8000
--smdeflation 0.9
```

It is possible to alter the input point observation uncertainty directly via the data fusion tool parameters. In the example below, the uncertainty of 'Klorofylli-a' is changed to 10% of its database value for station with ID 1234 while the uncertainty of 'Sameus' is changed to twice its original value for all stations:

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 2" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30 --initerror 0.5
--modelerror 0.2 --corlength 10000 --smdeflation 0.9
--edit-station-error CP@1234=10% TURB=200%
```

To produce better looking output, one may want to increase the number of ensemble members from the default setting. In the example below we use an ensemble of 200 members instead of the default 40:

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 2" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30 --initerror 0.5
--modelerror 0.2 --corlength 10000 --smdeflation 0.9 --ensemble-size 200
```

Executing the basic example using the ESTKS algorithm instead of the default (EnKS currently).

```
> calculate_data_fusion calc --model-domain Rannikko
--description "Example 3" --ext-sources * --quantities CP TURB
--start-date 2011-05-01 --end-date 2011-10-30 --initerror 0.5
--modelerror 0.2 --corlength 10000 --smdeflation 0.9 --ks ESTKS
```

7 Batch export of results

The data fusion results, including the point estimate and uncertainty, can be exported into a set of rasters in GeoTiff format for further processing, e.g. when using third-party software or if connection to the Data Fusion System database cannot be used for other reasons. The export process is executed via the `export_results` executable available in the Python environment.

Similarly to the remaining tools in the SDFS toolset, the tool prints usage instructions to the console when executed with the single argument `--help`:

```
> export_results --help
```

The export process can be cancelled by pressing Ctrl-C while the console window is active. Please note that the check for cancellation cannot for technical reasons be made within the linear algebra Fortran routines and the Ctrl-C press may therefore take a moment to be registered.

The export tool stores all exported datasets into a folder specified by the user. The names of the exported files can be customized via name templates. The tool exports result data for all results that match search criteria given by the user, such as the model domain name, time interval, quantities to be exported and the result description.

7.1 Export tool parameters

Parameters accepted by the `export_results` tool are listed below.

7.1.1 Required parameters

`--model-domain <name>`

Name of the model domain for which the results should be exported.

`--output-dir <name>`

Name of the model domain for which the results should be exported.

7.1.2 Optional parameters

`--quantities <quantity> [<quantity …]`

Names of quantities to export.

The names can be specified both in long and short format – as an example, the quantities "Klorofylli-a" and "CP" are identical. Multiple quantities can be given as a list of names separated by space:

> `--quantities CP TURB` or

> `--quantities Klorofylli-a Sameus`

If not given, all quantities are exported.

`--start-date <date>`

First day for which data should be exported, given in YYYY-MM-DD format.

If not given, the first day of the result is assumed.

```
--end-date <date>
```

Last day (inclusive) for which data should be exported, given in YYYY-MM-DD format.

If not given, the last day of the result is assumed.

```
--description <text>
```

Selects results by description.

If given, only results whose description matches the text will be exported. The parameter can contain wildcards: "*" (matches zero or more characters) and "?" (matches single character):

`--description "*_FINAL_*"` – Only results whose description text contains the sequence "_FINAL_" will be exported

If not given, description is not taken into account when matching results to be exported. Please note that the parameter value should be quoted if it contains spaces.

```
--result-id <id>
```

Selects result directly by the ID.

```
--status <status>
```

Selects result by status.

Allowed values are: "DONE", "CANCELLED" and "FAILED". Multiple values can be given to select results with any of the listed statuses.

Example:

`--status DONE` - Select results that have been completed successfully

```
--output-value-filename <text>
```

Estimated value file name template.

This specifies the file name where the value of the estimated quantity is stored within the output directory and should end with ".tif". The template can contain placeholders:

- `%r` – Substituted by the result ID
- `%q` – Substituted by the (short) quantity name
- `%d` – Substituted by the currently exported date
- `%i` – Substituted by a unique increasing counter (starts from "00001")

If not specified, the default value is "`value-%r-%q-%d.tif`" and leads to files such as "value-00001-CP-2018-06-01.tif"

```
--output-error-filename <text>
```

Uncertainty file name template.

This specifies the file name where the uncertainty estimate is stored within the output directory and should end with ".tif". The template can contain placeholders, just like the `--output-value-filename` parameter.

If not specified, the default value is "`error-%r-%q-%d.tif`"

7.2 Examples

Export all results for model domain "Pyhajarvi":

```
> export_results --model-domain Pyhajarvi --output-dir c:\exported
```

Please note that this is often not a good idea as there may be multiple results calculated for the domain. Therefore, it may be beneficial to limit the export, for example by utilizing the description attribute of the result:

```
> export_results --model-domain Pyhajarvi --description "My test"
--output-dir c:\exported
```

It is also possible to use wildcards, "*" (matches zero or more characters) and "?" (matches single character) in the description filter. The example below only selects results that start with "Experiment1" and contain the text "FINAL" somewhere later in the description:

```
> export_results --model-domain Pyhajarvi
--description "Experiment1*FINAL*"
--output-dir c:\exported
```

It is possible to limit the days that are exported:

```
> export_results --model-domain Pyhajarvi
--start-date 2018-06-01 --end-date 2018-06-10
--output-dir c:\exported
```

Likewise, it is possible to select only certain quantities to export:

```
> export_results --model-domain Pyhajarvi
--start-date 2018-06-01 --end-date 2018-06-10 --quantities CP
--output-dir c:\exported
```

To make sure only the results that were calculated successfully to the end, use the `--status` option:

```
> export_results --model-domain Pyhajarvi
--start-date 2018-06-01 --end-date 2018-06-10 --quantities CP
--status DONE
--output-dir c:\exported
```

We may want to use simpler file names than the default, for example omitting the result ID:

```
> export_results --model-domain Pyhajarvi
--start-date 2018-06-01 --end-date 2018-06-10 --quantities CP
--status DONE
--output-dir c:\exported
--output-value-filename value-%q-%d.tif
--output-error-filename error-%q-%d.tif
```

This will result in files such as "value-CP-2018-06-01.tif" being exported.

8 DFS Browser

DFS Browser is a QGIS addin for visualizing the source and results data in QGIS
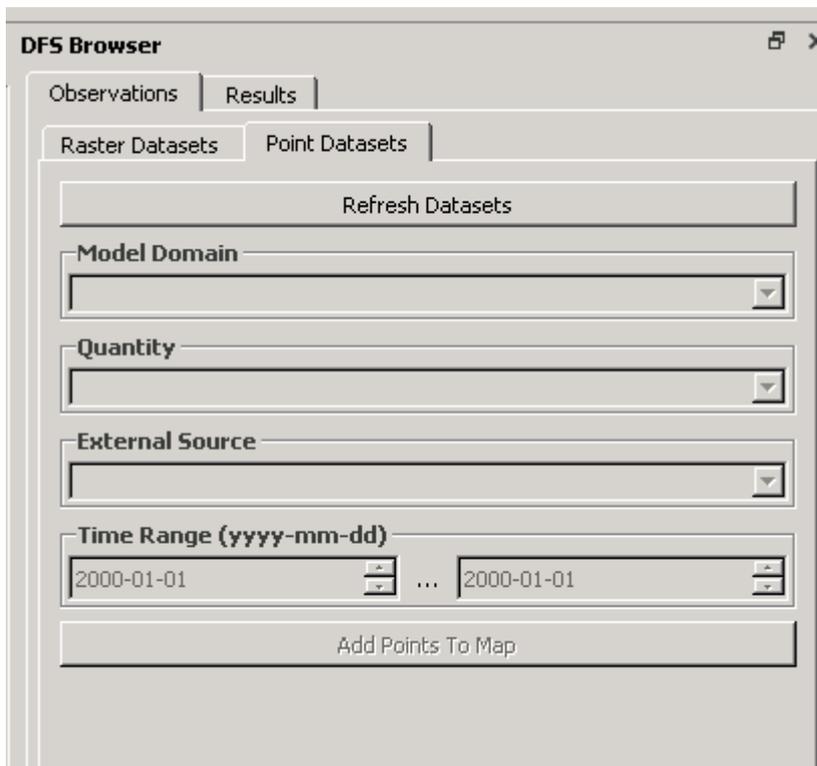
8.1 *Starting DFS browser*

Before you can use DBF Browser you need to start QGIS.

If you do not have QGIS icon on your desktop or Start menu then:

1. Open start menu
2. Select "all programs"
3. Open QSGeo4W menu
4. QGIS Desktop

After QGIS has started start DBF Browser by clicking the DFS Browser icon ![icon](DFS QGIS) in the toolbar and the DFS Browser will open:

and the DFS Browser layers will be added into the QGIS Layers list.
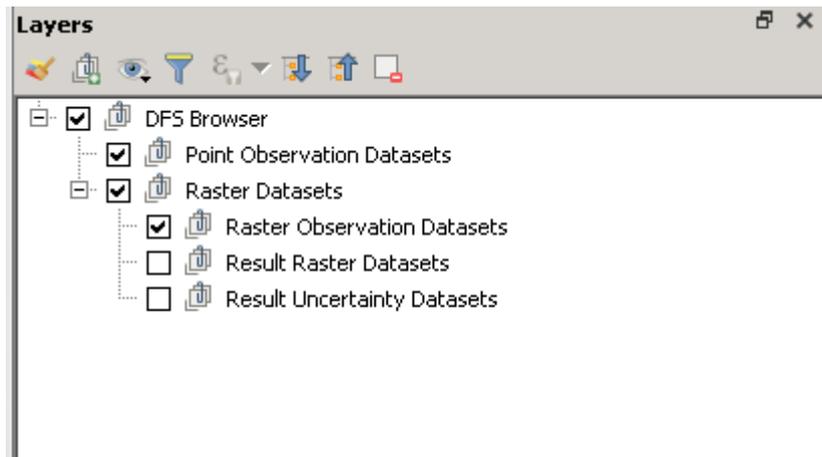
8.2   Viewing observation datasets

To view observation datasets, select Observations tab in DFS Browser. Under the Observations tab, there are two sub tabs:

- Point Datasets for selection of Point observations datasets
- Raster Datasets for selection of Raster observation datasets
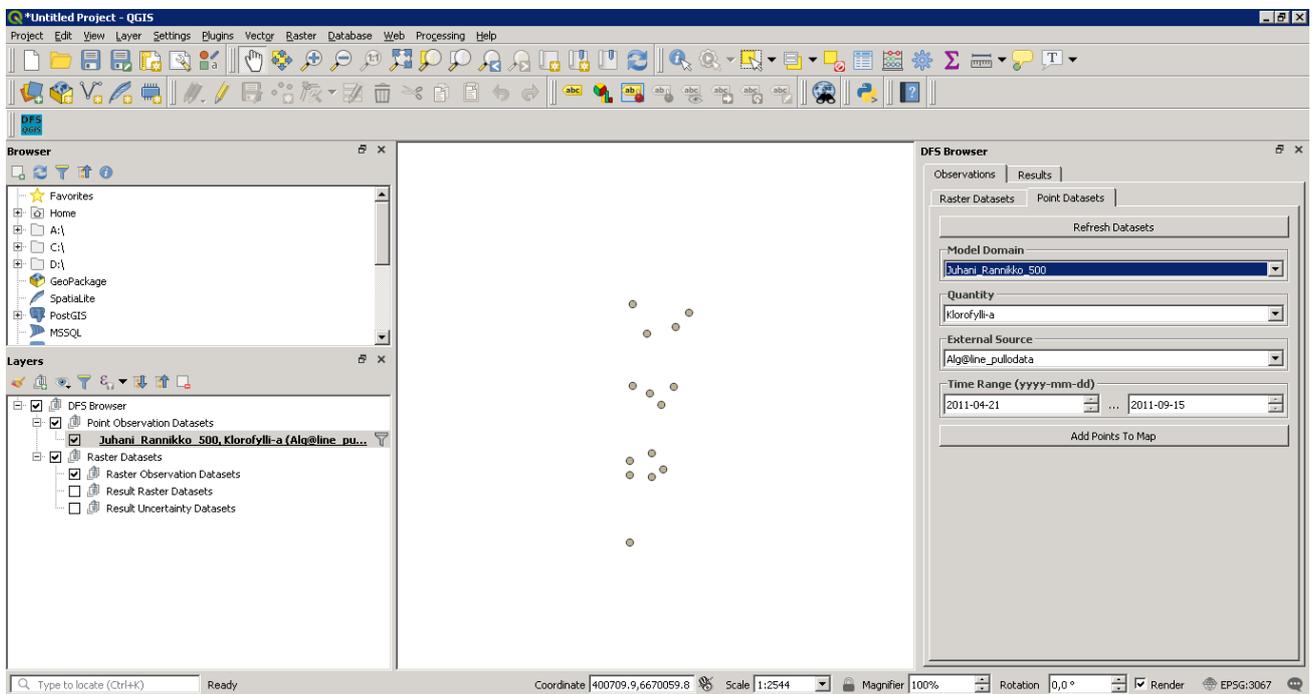
8.2.1   Point datasets

Select Point Data Sets sub tab and press "Resfresh datasets" -button to read the list of the available datasets in the database.

After that select from pulldown menus the

- Model domain
- Quantity
- External source
- Time range of the observations
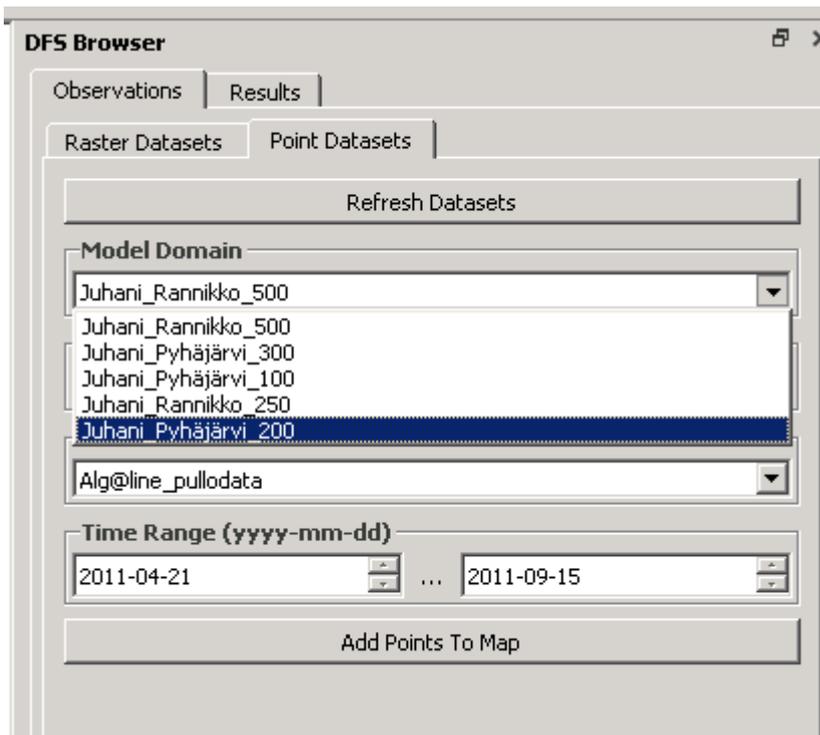
and press "Add Points To Map" to load the data on the map. The selected point observations are drawn to the map and the selection is shown in the Layers list.
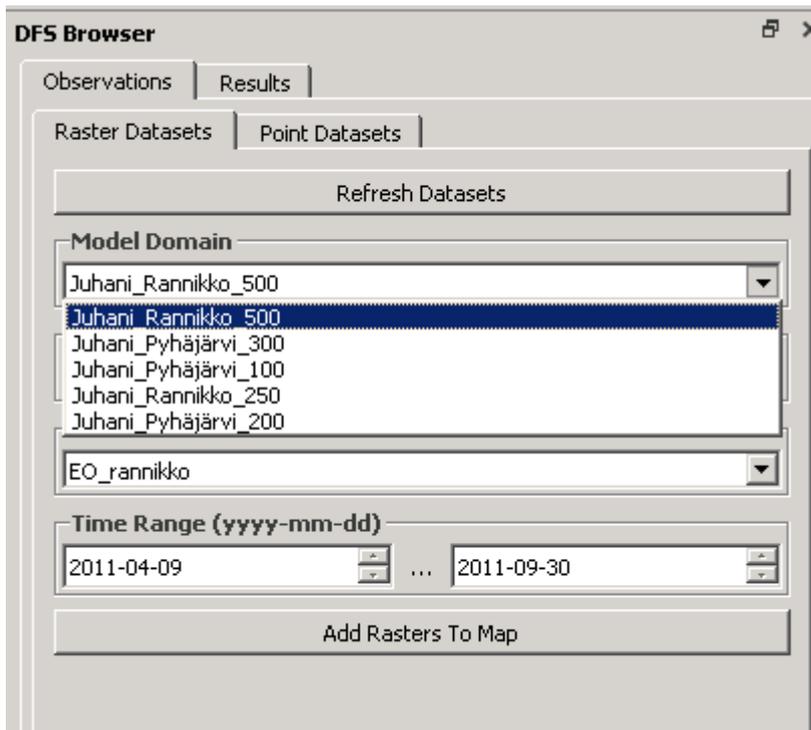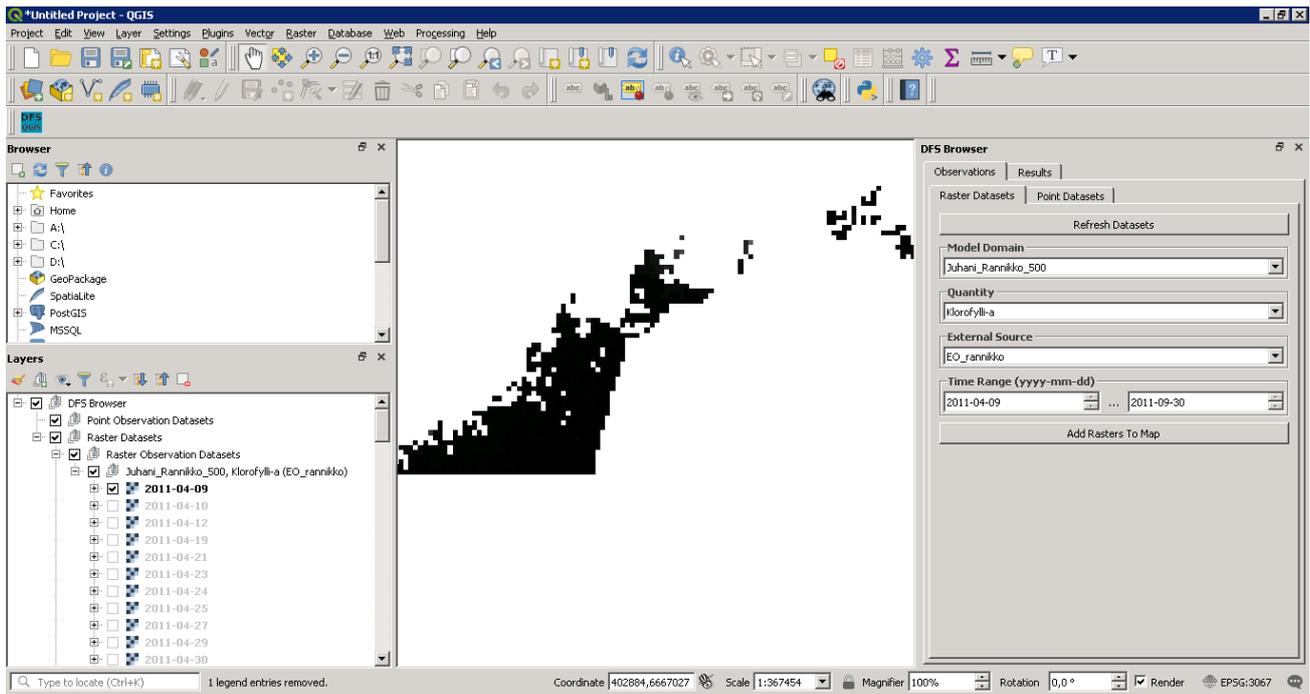
8.2.2 Raster datasets

Select Raster Data Sets sub tab and press "Resfresh datasets" -button to read the list of the available datasets in the database.

After that select from pulldown menus the

- Model domain
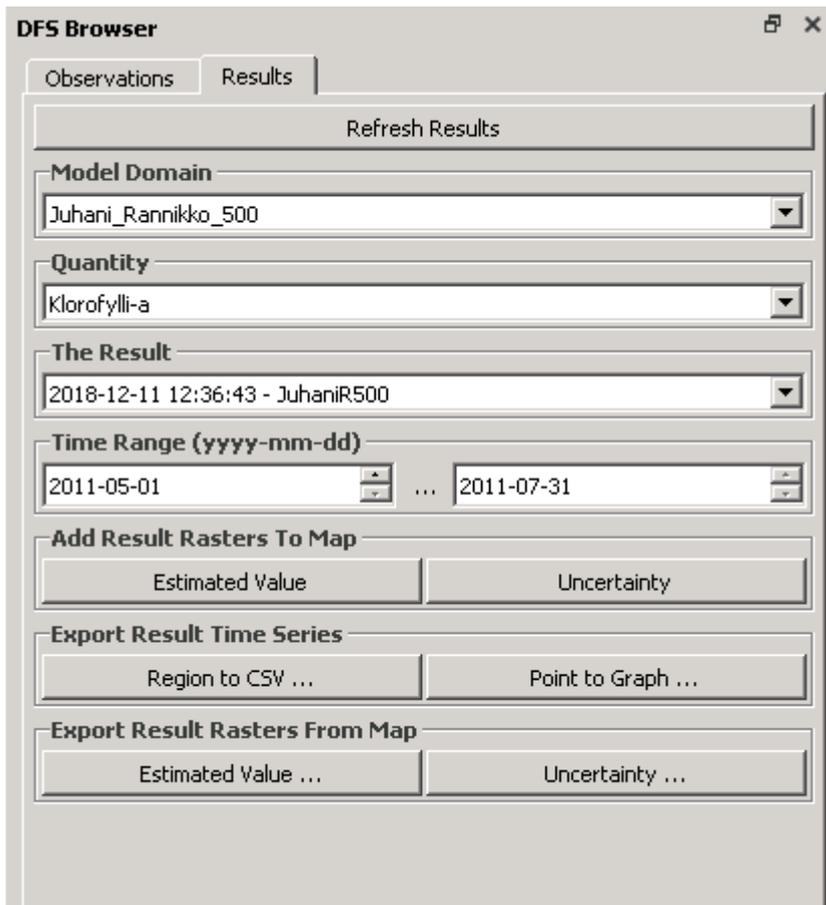- Quantity
- External source
- Time range of the observations

and press "Add Rasters To Map" to load the data on the map. The selected raster observations are listed in the Layers list and the day raster map of the first date on the date range is shown on the map.

You can change the day raster by selecting the date from the Leyers list. The map shows only one day raster/layer on the time i.e. when you select a new day raster, the previous selection will be switched off.

8.3   Viewing and exporting results datasets

To view and export results datasets select Results tab in DFS Browser. Press "Resfresh Results" -button to read the list of the available result datasets in the database.

After that select from pulldown menus the

- Model domain
- Quantity
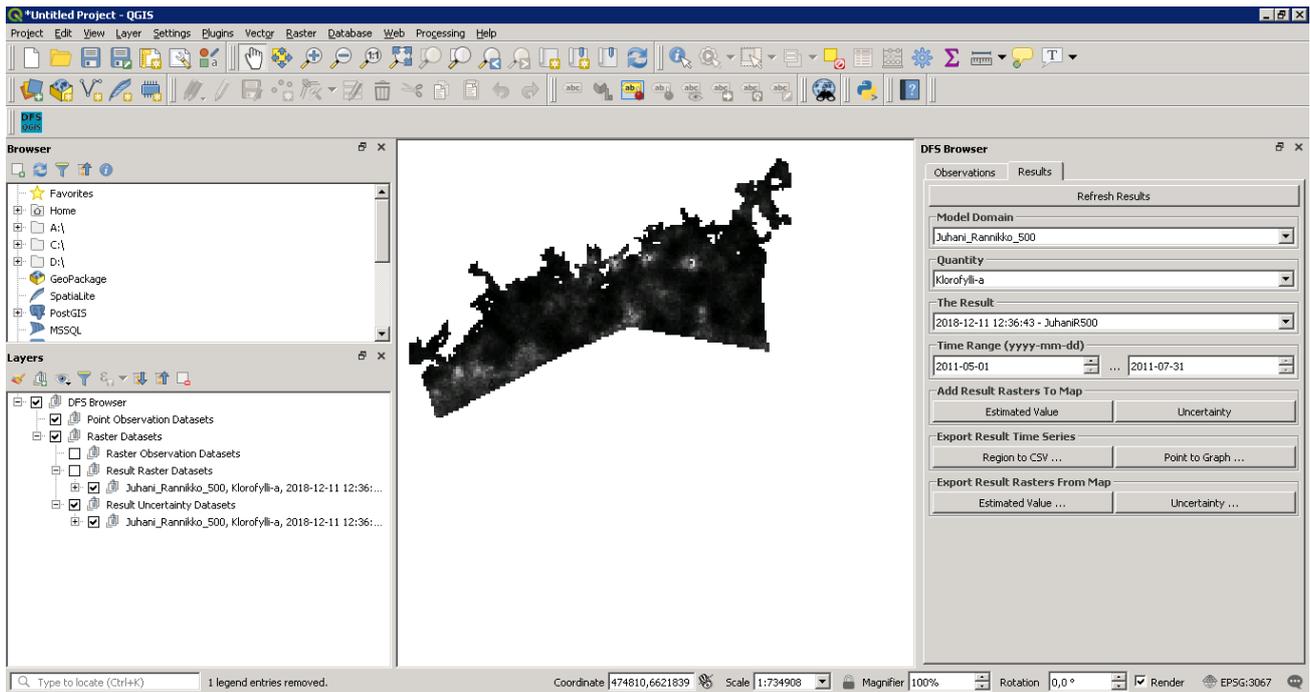- The Result
- Time range of the results data set

8.3.1 Adding result rasters to map

8.3.1.1 Estimated value

After selecting the results data set, you can add estimated values to the map by pressing the "Estimated Value" button. The results will be added into the Result datasets layer on the map.

8.3.1.2 Uncertainty

After selecting the results data set, you can add uncertainties of the results to the map by pressing the "Uncertainty" button. The uncertainty rasters will be added into the Uncertainty layer on the map.

8.3.2 Exporting results time series

8.3.2.1 Region to CSV file

Ater you have selected the results, you can export results data to SCV file as following:

1. Press region to CSV -button, Buton will stay selected and the application shows the ares of the Results data sets (green rectangle)
2. Draw the area to the map where you want to export the data (will be shown red)
3. Wait until Estimaged values export -dialog opens and give thefile name of the CSV -file to be saved. Note that depending on how large time interval and how large area you have selected it can take long time after step 2 before the export dialog is open.
4. Press "Save"
5.
   a. If you want to export another are to CSV file go to step 2
   b. If you are finished with exporting data, press region to CSV -button to finish (the button returns to unselected state)

8.3.2.2 Point to graph

You can create a graph from a selected cell on the results raster on the map as following:

1. Press the "Point to Graph" button
2. Select a cell fromt the map

3. Application opens the graph
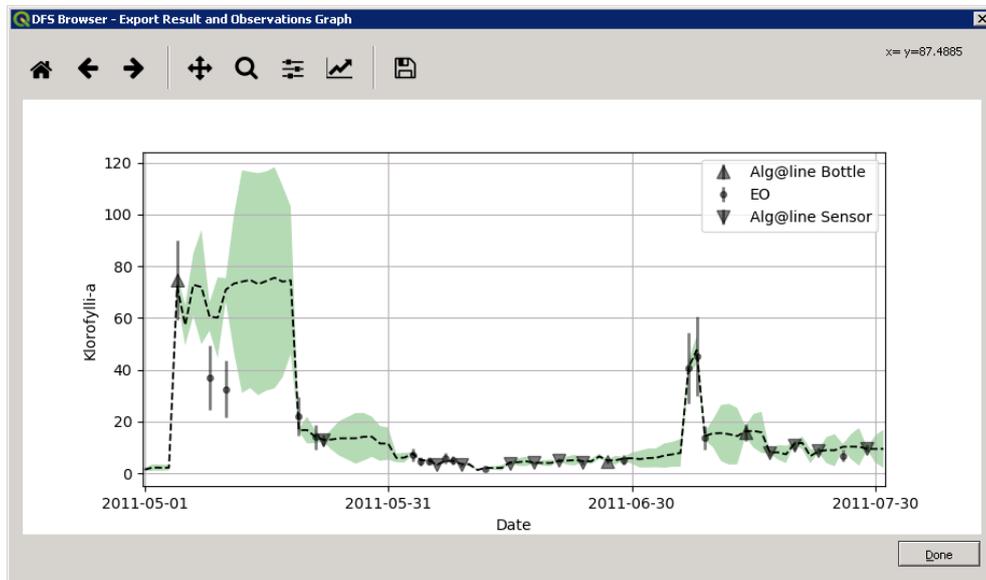
4. Close (done) the graph and
5.
    a. Select a new cell for the graph
    b. Exit from the Point to Graph tool mode by pressing the "Point to Graph" button again

The Graph window has a legend which shows the observation data values, for example:

and a toolbar which has the following functions:

 Reset original View

 Back to previous view

 Forward to next view

 Pan axes with left mouse button and zoom with right mouse button

 Zoom to rectangle

 Configure subplots

 Edit axis, curve and image parameters

 Save graph to file

8.3.3 Exporting results raster from the map

After you have added the results dataset to the map you can export the value and uncertainty raster maps to the file.

8.3.3.1 *Estimated value*

To export estimated value raster maps to a file:

1. Press the "Estimated value…" button
2. Select a folder where the raster images are to be saved and press select folder
3. Application creates the estimated value raster files.

8.3.3.2 *Uncertainty*

To export Uncertainty raster maps to a file:

1. In Export result rasters from Map section, press the "Uncertainty…" button
2. Select a folder where the raster images are to be saved and press select folder
3. Application creates the uncertainty raster files.

8.3.4 Removing results

Beginning with version 1.6, it is possible to remove data fusion results from within the DFS Browser addin. To remove a result:

1. Select the model domain, quantity and result on the "Results" page of the DFS Browser component
2. Press the "Delete Result" button
3. You will be prompted to confirm deleting the result
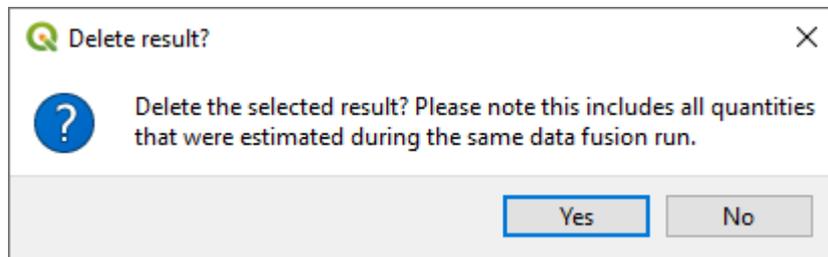
4. Click Yes to confirm or No to deny the operation
5. The DFS Browser UI elements are updated automatically after successful removal of a result

Data fusion result can be removed even if there are QGIS map layers utilizing the results data, the affected map layers will simply no longer be drawn.

Note: Please note that the "Delete Result" button removes the entire result record, including **all quantities** that have been estimated during the corresponding data fusion run (regardless of which quantity is selected in the DFS Browser UI).