

The Graphical User Interface for CAPRI

version 2008

Wolfgang Britz
Institute for food and resource economics
University of Bonn

Bonn, November 2008

Acknowledgments

Many people have over the years contributed to the development, maintenance and application of the CAPRI modelling system, and after now ten years since a first prototype was constructed, it is almost impossible to list them all and name their specific contributions. The author opted for this rather technical paper to refrain from citing the different (working) papers which shed more light on methodological questions, but rather refers in general to the CAPRI documentation.

Nevertheless, it is only fair to mention Hans-Josef Greuel and Andrea Zintl, who both long before CAPRI was born have already developed software concepts and code which underlined to a large extent until 2006 the DBMS of CAPRI, and in parts, its Graphical User Interface. Finally, Alexander Gocht over last years contributed to the Java code underlying the GUI.

The work described in here would have been impossible without the funds by different donors, mainly the EU Commission.

All errors in text and code remain with the author.

The author

Dr. Wolfgang Britz is a senior researcher and lecturer with the Institute for Food and Resource Economics at the University of Bonn, and has co-ordinated since several years the activities based on the CAPRI modelling system. Its responsibilities further on include the methodological concept of CAPRI and , to a larger extent, its software implementation.

Contact:

Dr. Wolfgang Britz

Institute for Food and Resource Economics, University Bonn

Nussallee 21

D-53115 Bonn

Tel.: ++49-(0)-228-732502

Wolfgang.Britz@ilr.uni-bonn.de

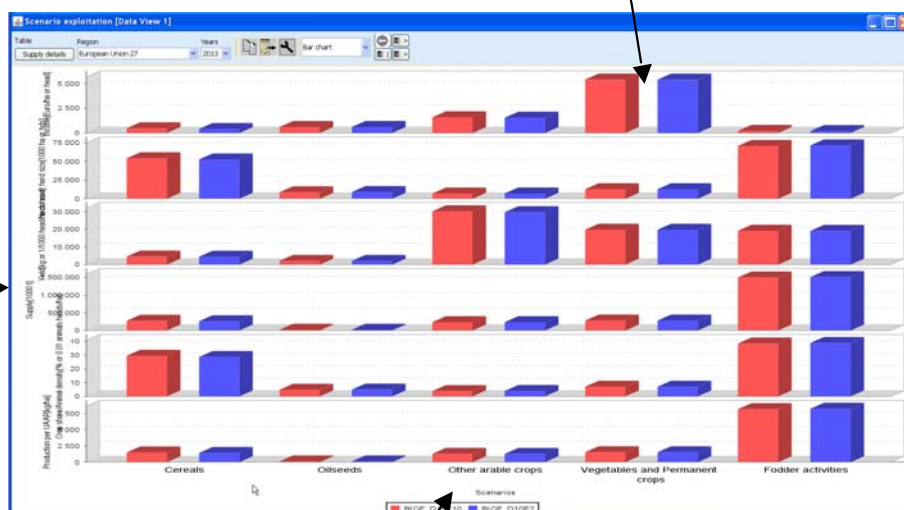
Content

Background	7
The software behind the mapping viewer and the CAPRI exploitation tools	7
CAPRI tasks as business model	9
Execution of tasks via a GamsStarter and GamsThread.....	10
Refactoring the mapping part	11
Views as the basic concept.....	12
Data model	13
Client based solution	14
The geometry model.....	15
Initialization	15
Logging in	16
Linking the GUI to the local CAPRI installation	17
GAMS settings	18
Getting help.....	18
Basic layout of the GUI.....	19
The different work steps.....	20
Build database	22
The work step “Generate baseline”	23
The task run simulation	25
The task: Exploit scenario results.....	27
The task: Collect meta information	27
Batch execution	29
Task Generate GAMS documentation.....	31
Exploitation of.gdx-files.....	31

An example: loading data from the spatial downscaling.....	33
Interaction with GAMS	35
Exploitation tools	35
The structure of the GAMS generated.gdx files.....	35
Loading the data from.gdx files	36
Multi-dimensional viewer with pivoting and exporting possibilities	37
Pre-defined views	38
Working with tables	39
The toolbar	39
View selection	39
Tooltips for column and row headers	40
Drill-down	40
Navigating in the outer dimensions of the viewport.....	40
Column and row selection	40
Predefined selection groups.....	42
Clipboard export.....	43
Export to file.....	43
Changing view options.....	43
Manually changing the pivot	44
Sorting	45
Numerical filtering based on cell content.....	46
Changing the row height and column width with the mouse	46
Changing the view type.....	46
Working with graphics	47
General handling of graphs	47
“Walking” through the data.....	49

Scenario exploitation (Data View 1)

	Scenario: BI_GFP_D10				Scenario: BI_GFP_D10E2							
	Income (Euro/ha or head)	Revenue or herd size (10000 ha or head)	Yield (kg or 5000 animals or head)	Supply (1000 t)	Crop share-Animal density (% or 0.01 animals/head/ha)	Production per 10000 (kg/ha)	Income (Euro/ha or head)	Revenue or herd size (10000 ha or head)	Yield (kg or 5000 animals or head)	Supply (1000 t)	Crop share-Animal density (% or 0.01 animals/head/ha)	Production per 10000 (kg/ha)
Cereals	533.81	6088.51	514.94	24784.88	32.71	5473.57	611.44	6878.78	4889.11	20956.19	24.71	1884.57
Oilseeds	468.78	8887.77	2386.65	22453.34	5.34	437.88	474.88	8886.66	2378.88	24678.57	5.38	138.78
Other arable crops	4758.88	7888.88	32193.87	28888.77	6.29	1474.29	1714.88	7878.38	32881.87	28878.57	6.29	1888.78
Vegetables and permanent crops	5887.14	4788.21	21878.38	21388.19	7.28	1787.18	1878.18	14711.28	21818.87	21384.29	7.28	1787.18
Fodder activities	238.29	18784.71	21248.88	188888.17	42.83	8887.88	218.81	28121.87	21171.84	187188.38	42.83	9171.88
Tot arable and follow herd	16.34	13884.41	164.21	1884.88	7.28	11.24	42.88	13887.88	168.88	2214.48	7.38	11.17
All cattle activities	227.81	94388.88	81.88	9183.88	11.38	64.17	277.88	93881.88	86.88	8813.31	18.88	43.81
Beef meat activities	8.81	38888.88	388.81	18878.88	18.77	18.11	1.78	38887.87	388.28	18878.11	18.88	18.11



Line and point charts 50

Pie charts 51

Spiderweb plots 52

Working with the mapping viewer 54

Changing the classification and the legend 54

Controlling the classification..... 55

Shrinking polygons according to UAA share..... 56

Area weighted classification 57

Excluding zeros from classification and removing small and large values..... 57

Classification method 57

Integration distribution information in the map window..... 60

Color table	60
Copying the map to the clipboard or saving to disk.....	66
Changing the title of the map	66
Zooming in and out and navigating in the map.....	66
Getting data for specific polygons.....	67
Highlighting specific regions in the map.....	68
Updating the map	72
Storing and re-loading your settings.....	72
Exporting the data underlying the map	73
What are the HSMUs and what information is available?	85
How to visualize the HSMU information.....	86
Loading data based on the GUI presets	87
Examples	89
Drawing a map showing the nitrate surplus for EU27 at regional level in the base year.....	90
Drawing a map of the High Nature Value Farmland indicator for Belgium for the base year.....	92
Drawing a map showing changes between the base year and the ex-ante results from the baseline ..	95
Drawing a map with the base year results next to one showing changes from the base year to the baseline results	98
Index.....	100

Background

The use of the CAPRI modelling system is increasing, and the user group becomes more diversified. Whereas in the first years, almost all users had directly contributed to model development and were familiar with the underlying GAMS code, more and more users now get to know about the system during training sessions, and have only a limited knowledge of GAMS and the CAPRI GAMS code. Already a few years back, a Graphical User Interface (GUI) was developed in order to support users to apply CAPRI for simulations and exploit results. For reasons laid down further down in a short chapter, this GUI needs to be now revised in major parts. That paper both explains the usage of the new GUI as well as the underlying software concept. It is structured as follows. The first chapter gives a short overview over the different work steps necessary to finally allow simulations with CAPRI.

The new GUI is as the old one realized in Java. The small technical background paper "Refactoring of the CAPRI GUI and integration of new functionalities" informs about the underlying software design.

The software behind the mapping viewer and the CAPRI exploitation tools

Reading the following chapter is not necessary to work with the GUI, but more directed towards a technically interested reader.

The original software implementation of CAPRI was based on software available at ILR at that time and comprised a DBMS realized in FORTRAN with C/C++ code for the GUI. The very first maps in CAPRI (in 1998) were produced with an MS-EXCEL mapping plug-in which was at that time a cost-free add-on. However, moving the data to EXCEL and then loading them in the viewer was not a real option for the daily debugging work on data base and model. Therefore, shortly before the first CAPRI project ended in 1999, a JAVA applet was programmed by W. Britz which was able to draw simple maps from CSV-Files automatically produced by the CAPMOD GAMS code. That code with slight modification remained active for quite a while, and some of the features are still to be found in the current mapping viewer. Then for a while, the exploitation tools were based on XML/XSLT+SVG and a mapping viewer in SVG was realized. However, the XML solution had the big disadvantage of requiring a large amount of single ASCII input files, and was not really performant when complex pivoting was used.

Therefore, the next evolution step was a pure Java GUI with direct access to GDX files which is the current state of the art. GDX files are an internal file format used by GAMS which allows a rather efficient I/O for large sparse tables. An API library allows to access GDX files from other applications. That design has the obvious advantage to be firstly based onto the portable JAVA language. Secondly, as

no external DBMS is used, it is possible to use CAPRI by solely executing GAMS programs. It is hence possible to use CAPRI on any systems where GAMS can be run, without the need to install additional software.

The GUI consists of three rather independent components. Firstly, a GUI to control the different work steps of CAPRI. The code deals mostly with defining GUI controls (button, scroll-down lists etc.) to manipulate properties of CAPRI tasks, and then to start them as GAMS processes. That part has been thoroughly refactored with the revision 2008. A second important part is the CAPRI exploitation tool, which are basically generic enough to be used for other modeling system as well. The current refactoring left most of the code untouched compared to the code based developed since 2006, with the exemption of the graphics which is now based on the JFreeChart library. However, as discussed below, in 2007, the mapping viewer was refactored in larger part to host the 1x1 km grid solution developed in the CAPRI-Dynaspat project. The exploitation tool is a rather unique solution to exploit result sets from economic models based on the definitions of views which are defined in XML tables. It combines features from DBMS reporting, data mining, spreadsheet functionalities and GIS into one package. And thirdly, there are some specialized pieces as the HTML based GAMS documentation generator which are linked into the GUI.

CAPRI tasks as business model

A core concept in the new layout is a business object called *AgpTask*. Technically defined as an interface, such an object represents a work task in the overall CAPRI system such a run of CAPREG to build the regional data base. The interface requires getters and setters for properties such as *baseYear*, *simYear* or *MemberStates*. The setters can be accessed either by a GUI interface or by the batch execution facility, formally by class implementing the interface *AgpTaskHandler*.

Most tasks are GAMS executable tasks according to their *isGams* property. These tasks also provide access to the name of the related GAMS program via *getGamsProgramName*. Each of these tasks has also a method called *generateIncludeFile* which generates the specific GAMS include file for that task.

The objects also know about the main GDX file they are generating via *getGdxResultFiles*. Related to that, they allow setting the logical names of the data dimension in the result data set via *setDimNames* and *setXMLTablesDims*.

Once the properties of a task had been defined, their logical consistency can be checked by invoking the method *checkSettings*. Check settings returns a string with a description of the first error encountered.

That layout eases dramatically the update process of CAPRI. Definition of new tasks or changes to existing ones will generally not require changes in the GUI, but simply either implementing a new object with the required methods or updating an existing one.

Execution of tasks via a GamsStarter and GamsThread

Execution of tasks with the property *isGams* is handled by a *GamsStarter* object. An instance of *GamsStarter* let the task write out the necessary include file(s) to generate a specific instance of the specific task (a simulation run for a specific scenario, simulation year, with the market model switched on or off ...). A *GamsStarter* also knows about the working directory or other specific GAMS settings as the scratch directory. It may generate a pipe for the GAMS output to the console to show it in a GUI.

An *AgpTask* can be executed by a *GamsStarter* who will create then a *GamsThread*. A *GamsThread* extends the *SwingWorker* interface of Java so that it may communicate with the normal event queue of JVM. A *GamsThread* can be gracefully terminated by sending a SIGINT signal to the GAMS process. That will let the GAMS execution stop at specific point determined by the GAMS engine itself and start the finalisation handling of GAMS as removal of intermediate files and directories.

Refactoring the mapping part

When the 1x1 km grid layer was added to CAPRI during the CAPRI-Dynaspat project it became obvious that the existing JAVA code to produce maps needed some revision, especially regarding the way the geometry was stored. In that context the question of using an existing GIS independently from CAPRI or the use of existing GIS classes plugged-into the CAPRI GUI was raised again; and some tests with open-source products were undertaken. A stand-alone GIS as the sole option was certainly the less appealing solution. Firstly, it would have required producing rather large intermediate files, and would have left the user with the time consuming and often error prone task of exporting and importing the data. Secondly, the user would need to switch between two different programs and GUI standards. And thirdly, all the usual problems with installing and maintaining additional software on a work station would occur. However, as indicated later, the GUI naturally allows passing data over to external application and does hence not prevent the user from using a full-fledged GIS solution.

The main points taken into account during the search of a map viewing solution for CAPRI were: (1) possibility to import efficiently data from the CAPRI GUI, (2) user-friendliness, (3) performance and (4), in the case of plug-in libraries, expected realization and maintenance resource need, and naturally (5) license costs. It turned quickly that an ideal product was not available. Some of the products were not able to allow for the necessary link between newly imported tables with region codes and an existing geo-referenced geometry. Others had very complex user interfaces or produced run-time errors, took ages to draw the HSMU maps or very quite expensive. From the different option tested, the gvSIG (<http://www.gvsig.com/index.php?idioma=en>) freeware GIS seemed to be the only allowing the user to import data from a CSV – which must however be semi-colon delimited – and join one of the columns to a shapefile. At least the version installed at that time was however running not very stable.

In the end, it was decided to build on the existing code base and let Wolfgang Britz write the additional code “on demand”. The main advantage of that approach is the fact that the mapping view is transparently integrated in the CAPRI GUI, it is sufficient to switch from “Table” to “Map” in a drop-down list to produce a colored map, and that user demands regarding additional functionality may be and had been added, taking into account the specific needs of the CAPRI network.

Compared to ArcGIS, where the EU27 HSMU geometry plus codes and centroids requires about 340 Mbytes, the CAPRI version requires about 27Mbytes solely. Reading in the CAPRI GUI is somewhat slower compared to ArcGIS due to unzip on demand. The actual drawing operation takes about the same time as in ArcGIS (about 11 second for the full data set), classification in Java is typically faster.

Views as the basic concept

The concept of the CAPRI exploitation tools is centred on the idea of a view. Content wise, that may be understood a certain angle to look at European agriculture as example its integration into international markets, its impact on the environmental, farm management, costs of the Common Agricultural Policy. Each view thus extracts a certain collection of numerical values, labels them so that they carry information to the user (long texts, units), chooses a matching presentation – as a table, as a map or a graphic, and arranges them in a suitable way on screen. The views can be linked to each others, allowing a WEB like navigation through the data cube. The user may have open several views in parallel, and he may change the views according to its needs, e.g. switch from a map to a tabular presentation, or change the pivot of the table, sort the rows etc.

Internally, each view is stored in a XML schema. A view could be also understood as a combination of a pre-defined selection query, along with reporting information. The XML schema allows to attach long texts, units and tooltips to the items of a table, and thus to show meta-data information to the user. The XML schema does hence replaces look up tables in a DBMS. It may equally store information regarding the pivoting, the view type (table, map, different graphic types), and for maps, classification, color ramp and number of classes. The views can be grouped into logical entities, and are shown as a popup menu to the user.

Tabular views may feature column and row groups. Empty columns and rows can be hidden, tables can be sorted by column, with multiple sort columns supported. Numerical filter can be applied to columns.

Data model

The underlying data model is very simple and straightforward. All data are kept in one large multi-dimensional data cube, and all values must either be float values or strings. Currently, only read-only is supported. Each data dimension is linked to a vector a string keys. Those keys are the base for the filter definitions. Currently, data cubes up to six dimensions are used (regions – activities – items – trading partners – years – policy scenarios). The data storage model is equally optimised to the specific needs. As only float values or strings are supported, all data can be stored as one primitive array of either floats or strings. To allow fast and efficient indexing, a linear index is constructed from the multi-dimensional data cube, and the non-zero data and their indices are stored in a hash table. That renders data retrieval very fast. All data are at initialisation time loaded in memory, for moderately long linear indices, about 10 Bytes are required to store a non-zero float and its index as an int. If the maximal linear index is very large, the index is store as a long and the storage needs goes up to about 16 Bytes. For moderately sized data cubes, 20 Million numbers can hence be hosted in about 200 Mbytes.

The data are read from a generic file format generated by GAMS (General Algebraic Modelling System, a commonly used software package in economic modelling) called GDX, the software package on which CAPRI is based. Access to GDX is handled via an API provided by GAMS.

Client based solution

Technically, the exploitation tool is completely client based. That reflects the specific user profile of the CAPRI modelling system where the exploitation tool is integrated with an economic model and tools building up its data base. The main aim of the tool is to support forward looking policy analysis, and user will create their own scenarios and in some cases even own variants of the export data, processes requiring considerable processing and storage resources. A client-server solution where the production process and data storage would need to be hosted on a web server is therefore not a preferred solution, also as users will often develop variants of the modelling system by code modification in GAMS, and contribute to its development. The structure of the data driver would however very easily support linkage to network or WEB based data bases. It should however be noted that the data base and GAMS code are managed via a Software versioning system, which is a kind of client-server environment.

The geometry model

The mapping viewer of CAPRI is based on very simple and straightforward concepts. First of all, it supports solely polygon geometries not comprising holes. The storage model is optimised to host rectangles, and is especially efficient if the polygons vertexes are all points in a raster. The topology is not read from a shapefile, but stored in a generic rather simple format. The vertices are stored in x,y coordinates already projected in a rectangular coordinate system, and the viewer does not deal with geographic coordinate system, but simply scale the rectangular coordinates in the viewport. The viewer in its current version solely supports one layer of quantities. Those restrictions naturally allow reducing memory needs, and, thanks to the rather simple data structures, also allow rather performing drawing operations. It should be noted that the JIT compiler of JAVA is indeed rather fast.

The biggest topology currently handled simultaneously covers an intersection of Corinne Land Cover, slope classes and Soil Morphological Units and comprises for EU27 around 2.7 Million polygons. As the majority of the polygons are rectangles, not more than 6-7 Million points needed to be stored. The topology handler and the drawing routines separate rectangles, for which only the two outer points are stored, from polygons, for which the vertices and centroids are stored.

The viewer is written in Java. There are two variants. One is a stand-alone version of the viewer realised as an applet. It reads from an internal portable binary data format, and java classes, data and geometry can be packed into one jar file, e.g. to ship it to a client. The second version is transparently integrated in the GUI of the CAPRI modelling system.

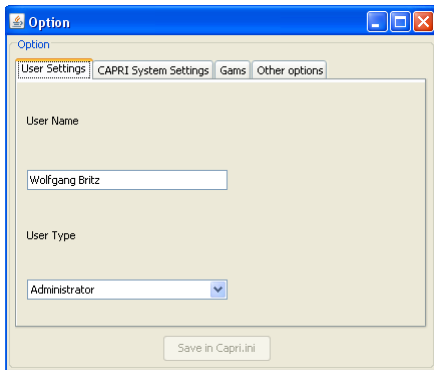
Swing is used for the GUI. In order to profit to the most of the simple implementation, the viewer has been written completely anew, and is not based on existing GIS libraries. Even certain standard JAVA classes as e.g. for hash tables, have been replaced by own implementations, to reduce implementation overhead. Some care was given to support flexibility in classification, given that only quantities are supported, so that the tool covers natural breaks, quantiles, equal spread, mean standard and nested means. Area weighting is supported as well.

In order to export data to other applications, the tools supports first of all tab delimited clipboard export, allowing import e.g. into EXCEL. Maps can be exported as JPEGs over the clipboard. Alternatively, the user may export to external file, in CSV format, DBF, to MS Access or to GAMS. DBF export will generate a second file comprising meta data.

Initialization

Logging in

The first step when the CAPRI GUI is opened for the first time is to set the user name and level, which is done by selecting the user menu from the menu bar. As long as no user name is entered, the user cannot change its type and will only have exploitation rights. The user type “runner” has additionally the right to run scenarios. A user of type “administrator” can perform all operations, including generation of a new data base and calibration of the modelling system. In order to access the user settings, choose

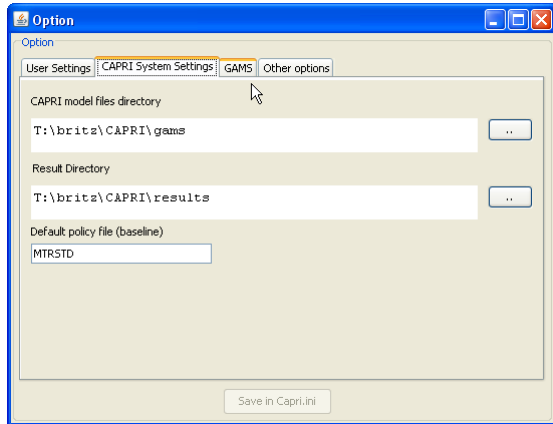


The user and user types can also be seen in the bottom panel of the GUI:



Linking the GUI to the local CAPRI installation

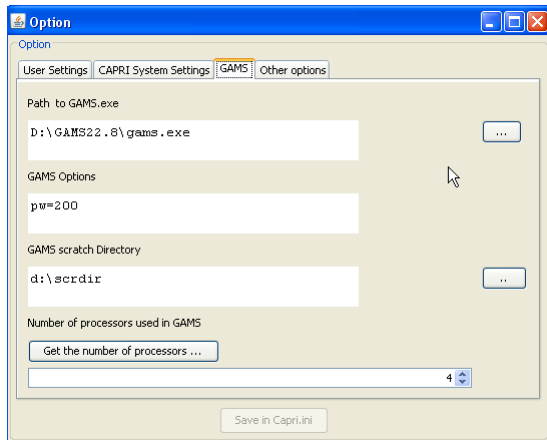
Next, the GUI needs to know where your CAPRI system is installed.



The “CAPRI model files directory” points to the location of the GAMS sources for CAPRI whereas the “Result directory” points to the location where results from CAPRI tasks will be read from and written from. Changing these settings allows switching between different installations for advanced users, e.g. when different branches from the CAPRI software versioning system are installed.

GAMS settings

In order to generate results, a GAMS installation and license are required. The relevant settings are found on the “GAMS” tab:

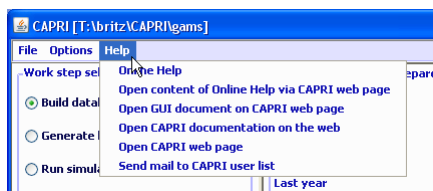


The “Path to Gams.exe” points to the actual GAMS engine to use. Currently, versions 22.3 and higher are supported. It is recommended to use GAMS 22.6 and above to benefit from the grid solve facility. The grid solve facility will use up to “Number of processor used in Gams” to solve model instances in parallel. The button “get the number of processors ...” will retrieve the number of available processors in the computer.

The “Scratch Directory” will be passed to GAMS and determines where GAMS stores temporary files. A directory on a local disk (not one on a file server) should be chosen.

The “GAMS options” field allows the user to send its own settings to GAMS, e.g. as shown above the page width use in listings.

Getting help



The “Help menu” allows opening the online help system, which can be invoked by pressing “F1”. A copy of the content is also stored on the CAPRI web page and can be accessed via the second menu item. “Open GUI document on CAPRI web page” will open the current document.

Basic layout of the GUI

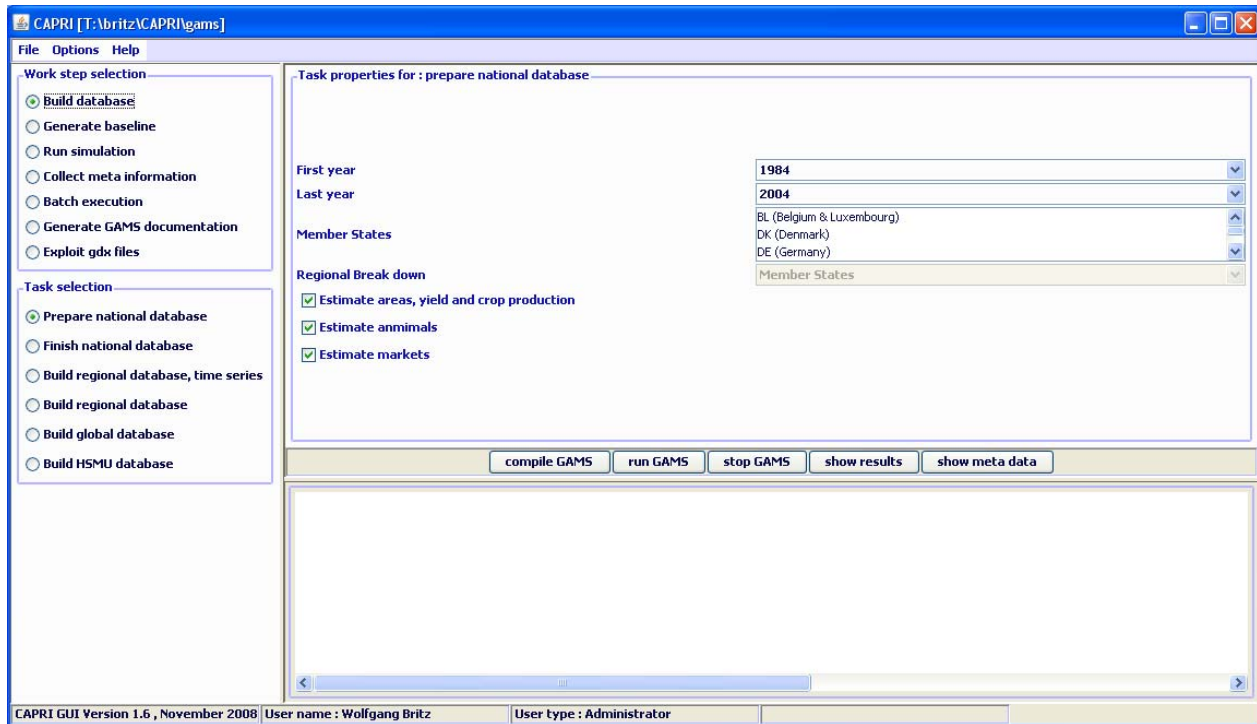
The GUI is generally structured as seen below. The left upper hand panel allows the selection of the different CAPRI work steps. The left lower hand panel lists the tasks belonging to the work steps. In both cases, only one selection is allowed. The right hand side offers controls depending on the properties of the task. There are buttons allowing starting the task, and typically a window which collects information at runtime. The footer lists the user name and type, and comprises progress bar.

For tasks linked to a GAMS program, the buttons as shown below will be active:

- “compile GAMS”: starts the GAMS compiler, but does not execute the program. A listing file will be generated. Used to test if a program compiles without errors.
- “run GAMS”: tries to execute the GAMS program. A listing file will be generate where possible compilation or run-time errors are reported.
- “stop GAMS”: sends a “signal interrupt” to the GAMS engine. It may take a while until GAMS reacts and stops with an error message after running its finalization routines.
- “show results”: loads the results form the task in the CAPRI exploitation tools
- “show meta data”: loads the meta data for the results from the task in the CAPRI exploitation tools.

Note: for exploiters, the three buttons referring to GAMS will be not visible. The same holds for runners and the work steps “Build data base” and “Generate baseline”.

Graph: Basic layout of the GUI



The different work steps

Each work step may comprise different tasks. No task will require starting more than one GAMS program, but some tasks will start the very same GAMS program with different settings. Some tasks will not start GAMS, but other tools inside the GUI.

The different work steps are shown in a panel in the lower left corner of the GUI, and are presented by so-called radio-buttons, which means, that only one button can be selected at any time.

Graph: the work step panel

Work step selection

- Build database**
- Generate baseline**
- Run simulation**
- Exploit scenario results**
- Collect meta information**
- Batch execution**
- Generate GAMS documentation**
- Exploit.gdx files**

Each work step may comprise several tasks, which are shown in second panel below the work step panel. The content of the panel hence changes when the user selects a different work step. Again, the different task panels comprise radio buttons for selections purposes.

Build database

Graph: the task panel for “build database”

Task selection

- Prepare national database**
- Finish national database**
- Build regional database, time series**
- Build regional database**
- Build global database**
- Build HSMU database**

Building up the data base is the logical starting point in sequences of work steps. A new data base for the model needs to be constructed either after updates of the underlying statistical raw data, or after methodological changes in the code affecting content and structure of the data base. Controlling if the different tasks necessary when updating the model yielded satisfactory results is a time demanding task which requires in-depth knowledge about the quality of the different in-going data and the logical relations between the different elements of the data base. Users interested in ex-ante policy analysis are usually better off by taking the data base as given, and consequently, the work step is disabled for users which have no “administrator” status.

The work step consists of six different tasks:

1. ***Prepare national data base:*** *Generation of complete and consistent time series at national level, mainly based on Eurostat data (CoCo, from Complete & Consistent). CoCo runs per Member State simultaneously for all years, if data from other Member States are used to derive fallbacks as EU average, only the raw statistical data are used. The user can only choose which countries to run, and which years to cover.*
2. ***Finish national data base:*** *Completion of the CoCo data by time series on consumer prices and certain feeding stuffs. In both cases, it turned out that only the complete and consistent time series for all Member States from 1. provide a good basis for that step. The step is hence run simultaneously for all Member States and years, based on the results of the CoCo task. Here, only the years to cover can be chosen by the user.*
3. ***Build regional data base, time series,*** *Generation of time series at regional level (CAPREG). The treatment of years in CAPREG is not identical. For all years, activity levels, output coefficients and input coefficients (excluding feed inputs) are generated. However, only for the base period, a*

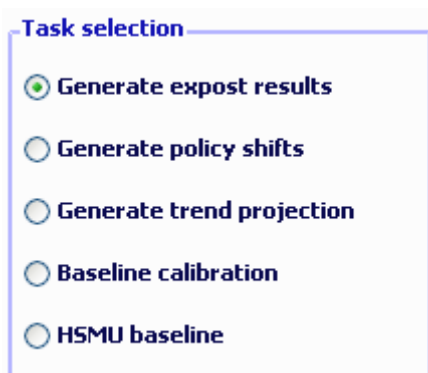
4. **Build regional data base (CAPREG).** Currently the same as three, only that the base year data will be loaded instead of time series.
5. **Build global data base (GLOBAL): Building up the international data base.** The step includes aggregation of Supply Utilization Accounts and bilateral trade flow matrices from FAO to the product and country definitions of CAPRI, aggregation of the supply and demand elasticities from the World Food Model to the product and country, estimation of bi-lateral transport costs and conversion of the FAPRI baseline to the product and regional aggregation of CAPRI.
6. **Build HSMU data base (CAPDIS_GRID):** spatial downscaling of regional results for the base year to 1x1 km grid cells.

The underlying methodology for the different work steps is described in detail in the CAPRI model documentation. The sequence of the tasks as described above follows the work flows. It should be mentioned that certain preparatory steps, as downloading updated data from EuroStat, and converting these data into GAMS tables read by CoCo and CAPREG are no yet integrated in the GUI.

The actual controls available will depend on task. Please use the “F1” button to open the online help to get detailed information on settings for the tasks.

The work step “Generate baseline”

Graph: the task panel for “Generate baseline”



For manifold reasons discussed in methodological papers, economic models as CAPRI are not suited for projections, but as tools for counterfactual analysis against an existing comparison point or an existing set

of ex-ante time series. The point in time or these time series are called “base line” or “reference run”. CAPRI “runners” which use the model for ex-ante policy simulation do not need to construct their own baseline, but are typically better off by sticking to the baseline provided on a yearly basis along with the latest version of the GAMS code, data base and software. Accordingly, the step and the included tasks are only for user type “administrator”. According to current planning, the baseline will be updated in close co-operation with DG-AGRI twice a year in early summer and early winter, following the release of a new “medium term market outlook” by DG-AGRI.

The CAPRI baseline is a mix of trends, expert knowledge and automated checks for logical consistency, and is constructed by a sequences of tasks:

1. *Generation of ex-post results.* Albeit not strictly necessary for the base line, the ex –post results often prove quite helpful when analysing the reference run. The ex-post results are model run for the base at base year policy and other exogenous parameters, inflated to the chosen simulation year.
2. *Generation of policy shifts.* In order to capture in the later trend projection the effect of policy changes between the base and the simulation year, so-called policy shifts are calculated by applying the ex-ante policy to the state-of-world (technical progress, management practises, population numbers, per capita income, consumption patterns etc.) in the base year. The policy shifts are defined as the relative changes against the base year of implementing the ex-ante policy.
3. *Generation of the trend projection.* The trend projection task is rather time consuming, and may run several days when the farm types are included. It consists of several sub-tasks. Firstly, independent trend lines for many different variables and all regions are estimated, and for each of these trends lines, statistics as R^2 , variance of the error terms etc. are calculated. These results, together with the base period data and the policy shifts, are used to define so-called supports, i.e. the most probable values for the final projection. These sub-tasks are relatively fast. The final consistency sub-task is broken down in two iterations. In the first iteration, only the Member States consistency problems are solved. For the different projection years, the problem will look for minimal deviation from the supports – which may be interpreted as a priori information in a Bayesian interpretation – such that different necessary logical relations between the data are not violated – the data information in a Bayesian estimator. These relation define e.g. production as the product of yield and activity level or force close market balances. The details can be found in the methodological documentation. Once that step is done, the Member state are added up to the EU level, and new support are defined which take given expert projection into account, currently mainly a baseline provided by DG-AGRI. In the second round, the Member State problems are

solved again, and then, problems for all NUTS II regions in each Member State, and, for all farm types inside of each NUTS II region.

4. *Baseline calibration.* In the final task, the results from the trend projection serve as the major input to generate the baseline.

The task run simulation

At the core of CAPRI stands its simulation engine, which iteratively links different types of economic models: aggregate programming models at regional or farm type level, with an explicit representation of agricultural production technology, aggregated versions of these models at Member States model linked together to derive market clearing prices for young animals, and finally, a global spatial multi-commodity model for main agricultural products and selected secondary processed products.

Differences in results between simulations may be rooted in three different blocks:

1. Differences in the in-going base year data and baseline. CAPRI allows several base years and calibration points to co-exist, and user may choose the base and baseline year.
2. Difference in what economic models are linked together and in the regionalisation level as the user may switch the market modules on or off, may run the model at Member State NUTS II and farm type level or in comparative static or recursive dynamic mode.
3. And finally, the most common, differences in the exogenous assumptions including the policy definition.

Graph: The interface in simulation mode

Task properties for : run simulation

Base year	2002
Simulation year	2013
Member States	BL (Belgium & Luxembourg) DK (Denmark) DE (Germany)
Regional Break down	Member States
Number of iterations	15
Scenario file name	01GWPBUR.GMS
<input type="checkbox"/> Global market model	<input type="checkbox"/> Young animal market model
<input checked="" type="checkbox"/> Aggregates for activities and commodities	<input checked="" type="checkbox"/> Environmental indicators
<input type="checkbox"/> Life-cycle assessment for energy	<input type="checkbox"/> Multi-functionality indicators

The following discussed the settings:

- **Base year:** determines the three year average underlying the regional (see *Build regional data base*) and global data base (see *Build regional data base*) and the trends (see *Generate trend projection*).
- **Simulation year:** the year for which results are generated and trends are loaded.
- **Member States:** if the global market model is switched off, the user may run a simulation for selected Member States, only.
- **Regional break down:** the level of regional dis-aggregation in the supply part. It is not longer recommended to user the “Member State” level for production runs.
- **Number of iterations:** with market models switched on, CAPRI sequentially calibrates the market models to supply model results which are solved at prices from the market models. Usually, fifteen iterations are recommended.
- **Scenario file name:** the GAMS file which comprises the settings for policy and further exogenous variables for a simulation. The files are stored in “gams\pol_input” and must be valid GAMS code. Use a text editor as e.g. the GAMS GUI to manipulate the files and generate new ones.
- **Global market model:** Switch the spatial global market model for agricultural products on and off. If switched off, outputs prices will be fixed to the baseline results. If switched on, the supply model will work with prices provided by the global market model, and the global market model will be iteratively calibrated to the results of the supply models aggregated to Member State level.

- *Young animal market model*: switch endogenous prices and market clearing for young animals on and off.
- The last four settings relate to post model reporting. Currently, the reporting part is still embedded in the GAMS code and cannot be started separately. That means that the full runs needs to be repeated if results from one of the reports are later needed.

The task: Exploit scenario results

The task allows loading simultaneously results from different scenarios into the exploitation tools.

Graph: The interface in exploit scenario result mode

Task properties for : exploit scenario results

Base year	2002
Simulation year	2013
Member States	BL (Belgium & Luxembourg) DK (Denmark) DE (Germany)
Regional Break down	Member States
Scenario 1	
Scenario 2	
Scenario 3	
Scenario 4	
Scenario 5	

show results show meta data

The task: Collect meta information

The task scans the GDX files from the different tasks and collects Meta information (user, date of execution etc.) from there and shows it in a table. Besides the result directory stored in the settings, the task performs a “passive crawl” such that it will collect information on all files in the result directory which could technically be generated by the tasks. Settings as base year or simulation year in the interface are not taking into account. The processing depends on availability of the META symbol in the GDX file. Using the new GUI with files based on older version of CAPRI could lead to situations where files are not included in the reporting system.

The following information is provided:

- **Member State:** the Member State for which Meta information is provided. Depending on the task, the program may have run for that Member State separately (e.g. CAPREG) or along with other Member State (e.g. CAPMOD)
- **Work step:** the task run as reported in the GDX file.
- **#:** logical order of the work steps. Steps with a higher order build on results of steps with a lower one. Consequently, results from steps from steps with a higher order should be younger than those from steps with a lower order.
- **Item:** the different types of Meta data provided.
- **Content:** actual Meta data for the item.
- **File name and SVN status:** the GDX file from which the meta information is loaded, plus information about its status in the SVN versioning system. “Modified” means that the file in the local copy was modified since the last update and was not committed afterward. “Out of date” means that a newer version is available on the server. “Conflicting” means that an updated version of the file is available to the local copy, but that the file was modified. “Not under version control” means that the file is not handled by the version system.

It allows checking the internal consistency, i.e. if results entering certain work steps are outdated. In that case, the line with the date stamp is shown in red.

The scroll down boxes above the table can be used to select in the table. The first entry (.*) selects all items. Clicking on the table columns allows sorting.

Graph: Table with meta information

Meta data information

Member state	WorkStep	#	Item	Content	File name and SWN status
AL000000	BASELINE CALIBRATION	4	TITLE OF DATA SET	Baseline calibration MTRSTD	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	WORKSTEP	Generate baseline	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	KEY	MTRSTD	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	NAME OF PROCESSOR ORGANISATION	Wolfgang Britz	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	TEMPORAL COVERAGE	2013	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	DATE OF VERSION	2008-09-26 15:53:45	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	BASEYEAR	2002	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	SIMYEAR	2013	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	MODEL_SWITCHES	REC_DYN OFF MARKET_M OFF YANJ_M OFF	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	MEMBER_STATES	BL DK DE EL ES FR IR IT NL AT PT FI SE UK CZ EE HU LT LV PL SI SK RO BG CY MT NO AL MK CS BA HR MO KO	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	LANGUAGE WITHIN THE DATA SET	ENGLISH	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	NAME OF EXCHANGE FORMAT	GDX	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	NAME OF OWNER ORGANISATION	CAPRI network	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	NAME OF ORIGINATOR ORGANISATION	CAPRI network	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	DESCRIPTION OF PROCESS STEP	Generate baseline, Baseline calibration	res_2_0213MTRSTD.gdx : Modified
AL000000	BASELINE CALIBRATION	4	GEOGRAPHIC COVERAGE BY NAME	BL DK DE EL ES FR IR IT NL AT PT FI SE UK CZ EE HU LT LV PL SI SK RO BG CY MT NO AL MK CS BA HR MO KO	res_2_0213MTRSTD.gdx : Modified
AL000000	GENERATE EXPOST RESULTS	2	TITLE OF DATA SET	Generate expost results EXPOST	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	WORKSTEP	Generate baseline	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	KEY	EXPOST	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	NAME OF PROCESSOR ORGANISATION	Alexander Godt	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	TEMPORAL COVERAGE	2002	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	DATE OF VERSION	2008-09-01 11:25:55	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	BASEYEAR	2002	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	SIMYEAR	2013	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	MODEL_SWITCHES	REC_DYN ON MARKET_M ON YANJ_M ON	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	MEMBER_STATES	BL DK DE EL ES FR IR IT NL AT PT FI SE UK CZ EE HU LT LV PL SI SK RO BG CY MT NO AL MK CS BA HR MO KO	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	LANGUAGE WITHIN THE DATA SET	ENGLISH	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	NAME OF EXCHANGE FORMAT	GDX	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	NAME OF OWNER ORGANISATION	CAPRI network	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	NAME OF ORIGINATOR ORGANISATION	CAPRI network	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	DESCRIPTION OF PROCESS STEP	Generate baseline, Generate expost results	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE EXPOST RESULTS	2	GEOGRAPHIC COVERAGE BY NAME	BL DK DE EL ES FR IR IT NL AT PT FI SE UK CZ EE HU LT LV PL SI SK RO BG CY MT NO AL MK CS BA HR MO KO	res_2_0202mtrstd.gdx : Normal
AL000000	GENERATE TREND PROJECTION	3	TITLE OF DATA SET	Generate trend projection CAPTRD	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	DATE OF VERSION	2008-09-24 22:53:35	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	TEMPORAL COVERAGE	1984 - 2030	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	LANGUAGE WITHIN THE DATA SET	ENGLISH	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	NAME OF EXCHANGE FORMAT	GDX	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	GEOGRAPHIC COVERAGE BY NAME	BL DK DE EL ES FR IR IT NL AT PT FI SE UK CZ EE HU LT LV PL SI SK RO BG CY MT NO AL MK CS BA HR MO KO	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	NAME OF ORIGINATOR ORGANISATION	CAPRI network	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	NAME OF OWNER ORGANISATION	CAPRI network	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	NAME OF PROCESSOR ORGANISATION	Wolfgang Britz	results_0213.gdx : Modified
AL000000	GENERATE TREND PROJECTION	3	DESCRIPTION OF PROCESS STEP	Generate baseline, Generate trend projection	results_0213.gdx : Modified

Batch execution

Graph: Batch execution panel

Batch execution

Batch file to execute:

Directory for exp/ref files:

Generate EXP and REF files for HTML documentation Only compile the GAMS programs

Start batch execution End batch execution after next finalised GAMS step End batch execution immediately

The batch execution allows starting a file defining settings and tasks from the different CAPRI work steps and executing them without user intervention. Once started, the batch processor may be stopped so that the currently running GAMS programs ends on its own (“end batch execution after next finalised GAMS step”) or by sending a “CTRL-C” to the GAMS program. It will continue to run until the GAMS processor notices the CTRL-C – which may take a while – and then end with an error code. However, the GAMS processor will run some finalisation tasks as removing temporary files and directories.

The batch execution facility is a tool which:

- Allows executing many different CAPRI task after each other without requiring user input.
- Reports the settings used, any errors and GAMS result codes in a HTML page from which they may queried at a later time.
- Ensures that each new run generates its own listing file, which can be opened from the HTML page.
- Allows storing the output of the different runs in a separate directory, while reading input from unchanged result directories.

The purpose of the batch execution facility is therefore at least twofold. At the one hand, it allows to set up test suits for the CAPRI GAMS code such as checking for compilation without errors for all tasks and different settings such as with and without market parts etc.. Secondly, production runs of e.g. different scenarios can be started automatically. It is planned to add timer facilities to the batch execution so that the GUI will start a suite of runs at a pre-scheduled time. Along with the planned functionalities to compare in a more or less automated way differences in results between versions, the batch facility is one important step towards quality control.

For details on the batch execution facility see the technical document “Batch execution of CAPRI tasks” to be found on the Capri web page under technical documents.

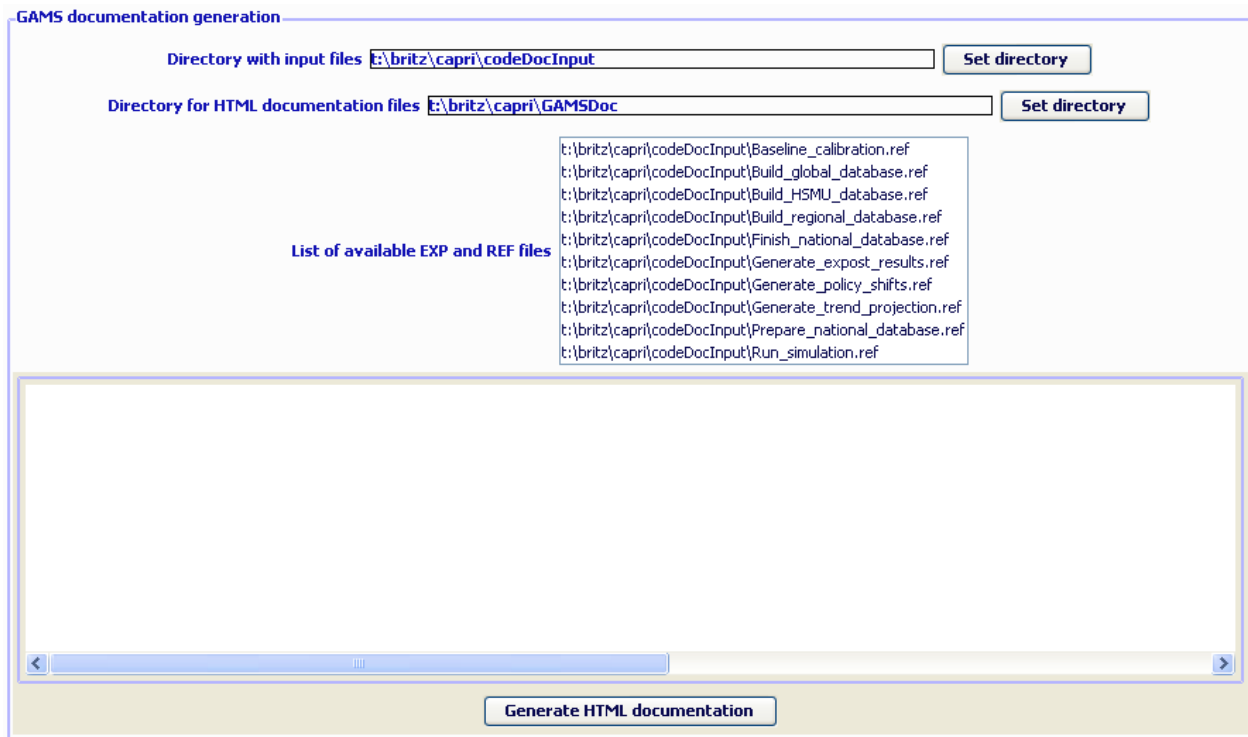
If the suite of tasks comprises “execute” statements, those can be downgraded to “compile” with “Only compile GAMS programs” check box.

The check box “Generate EXP/REF files for HTML documentation” adds settings to the GAMS calls which generate two specific reference files by the GAMS compiler which comprise information of files and symbols used by GAMS. For details on the code documentation facility see the technical document “Javadoc like technical documentation for CAPRI” to be found on the Capri web page under technical documents. The “directory for exp/ref files” defines where those files will be stored.

The batch language allows definition of a timer, i.e. to start the execution at a specified time.

Task Generate GAMS documentation

Graph: Panel to steer GAMS documentation generation



The GUI comprises a tool to generate for each GAMS file and each symbol used HTML pages which are interlinked. For details on the code documentation facility see the technical document “Javadoc like technical documentation for CAPRI” to be found on the Capri web page under technical documents.

The controls on top allow the user:

- To define in which directory the “EXP”, “REF” and “GDX” files are stored which serve as input into the documentation generator.
- To chose the directory where the HTML files will be generated.
- To select the tasks covered by the documentation generator.

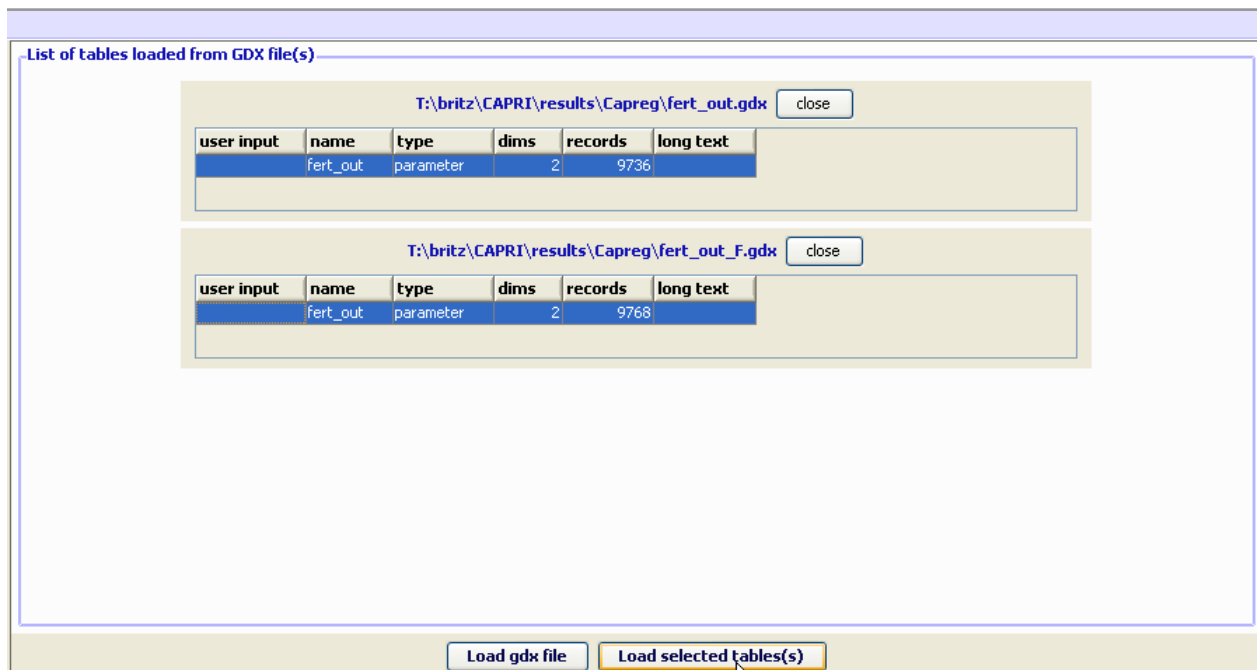
Exploitation of gdx-files

GDX-files are generated by GAMS and typically serve either an exchange format between different GAMS applications, or for exploitation purposes as the GAMS-IDE comprises a view for GDX-files. Further tools for GDX-files are available from GAMS company and are described in different documents.

In opposite to listings generated by GAMS programs, the GDX files store the data in full numerical precision in an internal format.

The new CAPRI version passes information from one task to the next with the help of GDX files, so generates CoCo a.gdx files with the time series at national level, which is read by CAPREG. And the regional time series generated by CAPREG are inputted by the trend projection tool CAPTRD. These.gdx files are accessed when the different tasks of “Data base exploitation” are chosen. The user has on top the possibility to load one or several tables from one or several freely chosen.gdx files.

Graph: Panel to GDX file exploitation



When the task “exploit.gdx files” is selected by pressing the related button, two buttons are shown in the task panel. The first one, labelled “load.gdx files” will open a file selection menu when pressed. When the ok button of the dialogue is operated, the content of the.gdx file is partially loaded, and a table is added to the right upper window of the application showing the parameters and sets comprised in the.gdx files, along with their number of dimensions and records. When the close button next to the table is pressed, the table is deleted. Pressing the “load.gdx file” again will add more tables.

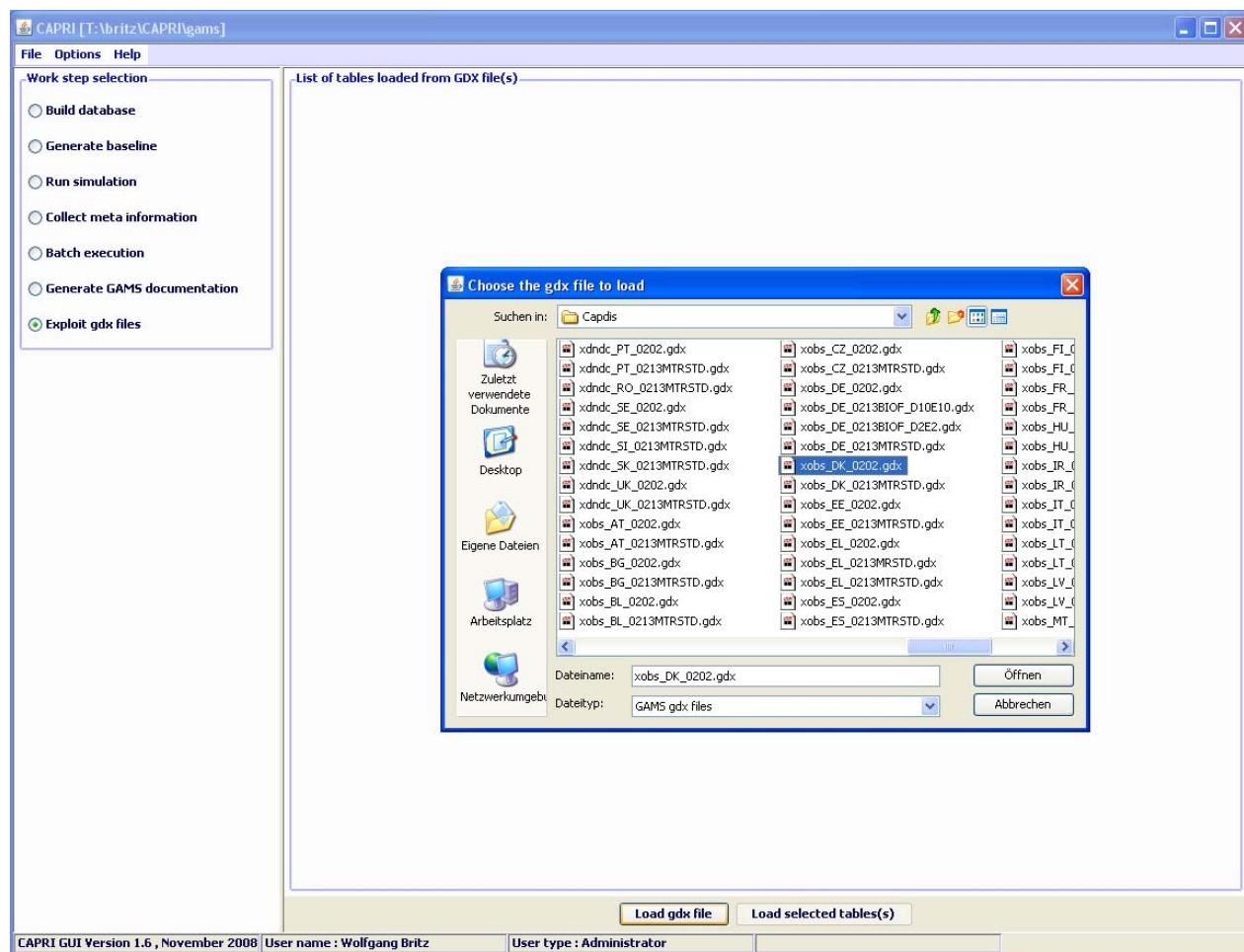
One parameter from each table may be selected (pressing the “ctrl” key when clicking with the mouse de-selects). If several parameters from one file needs to be loaded, the user may open the same file several time.

The content of the different parameters is merged together, and the parameters themselves span an additional data dimension. If the user does not provide input in the first column of the tables labelled “user

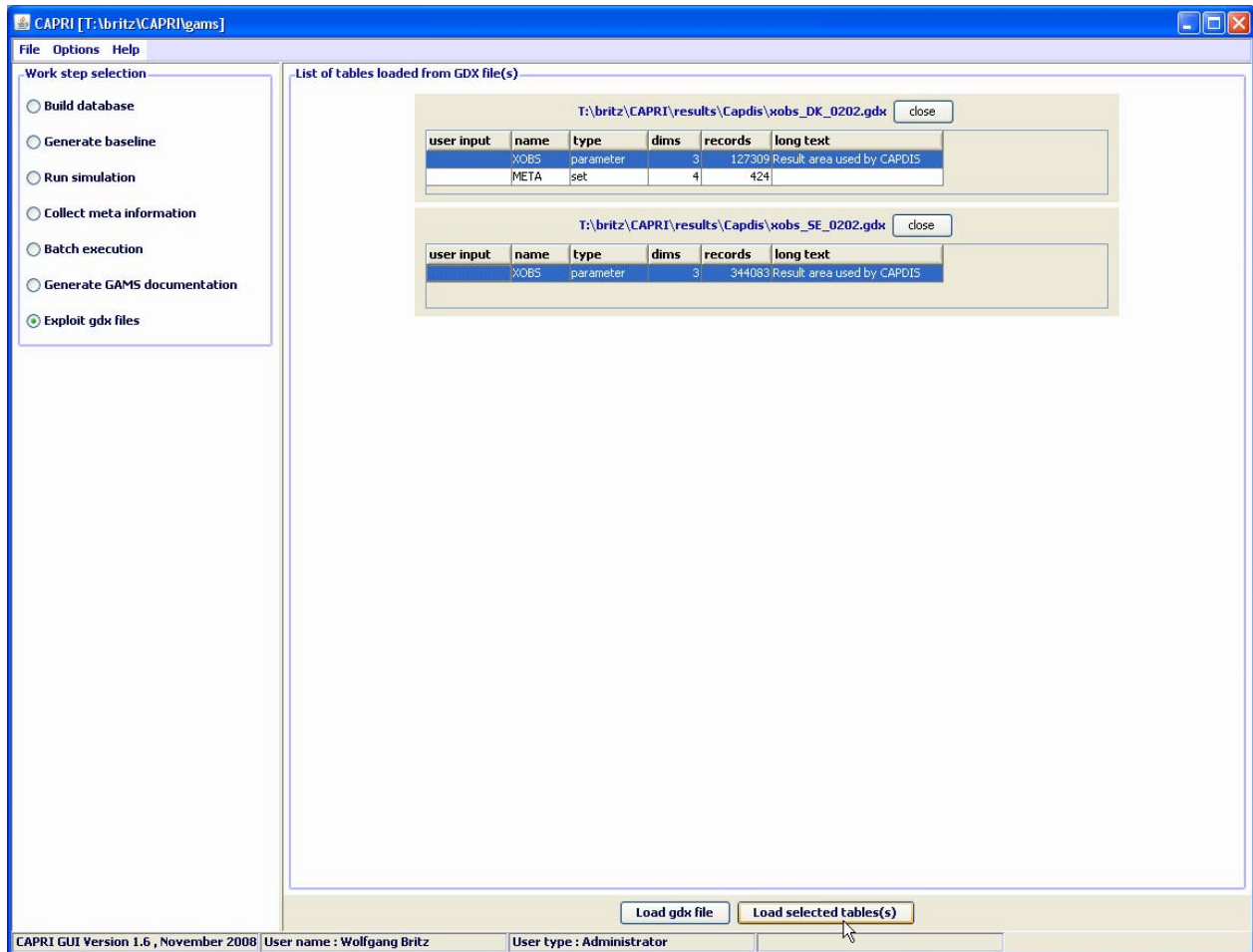
input”, the program will generate automatically names. The data loaded are shown in the table tool described above.

An example: loading data from the spatial downscaling

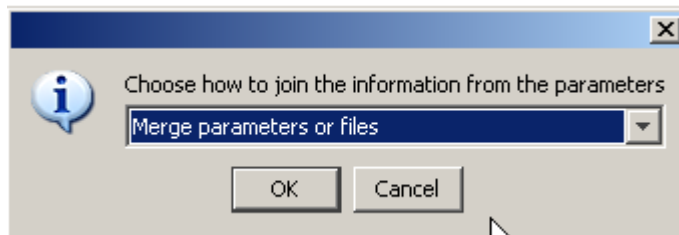
The option described here is introduced for completeness. The names of the file generated by the disaggregation programs start with “XOBS_” followed with the two character code of the Member state, then an underscore followed by the base year and the simulation year, and if applicable, the code for the simulation which is identical to the name of the GAMS files used from “pol_input” which was used to run the scenario.



If one wishes to see the information for several Member States simultaneously, one may click on “Load.gdx file” again and add additional files. Afterwards, the lines with the parameters “XOBS” should be selected.



Pressing load selected tables will open the following dialogue, choose



As otherwise, the program will introduce a new dimension for the data loaded from the different files, and you will not be able to see the information for “DK” and “SE” together in one map. Afterwards, the selected records will be loaded from the files.

Depending on the amounts of records, that may take several seconds. Before turning to the mapping view, only one column should be selected. Press on the button left of “SWHE” in the table headers (“Open selection dialog for table column”), and select with the mouse one of the codes, then press o.k. The table should now comprise only one column. Afterwards use the drop-down list with the viewing options and

choose map as shown below. Choose “HSMU.” to select the geometry for the HSMUs. The program will now load the geometry for the HSMU and draw the map which takes several seconds.

Interaction with GAMS

The interaction with GAMS consists of three parts:

- Generating GAMS code based on user input
- Starting GAMS
- Controlling the GAMS run

There are two types of input files generated based on user input. The first one are so-called scenario files and define the exogenous drivers for a CAPRI run as population growth, macro-economic environment or policy definitions. Here, the final aim is to integrate the scenario editor from SEAMLESS into the CAPRI user interface. The scenario files are typically stored for longer period on disk, both to provide templates for other scenarios as well as for documentation purposes. The name of the file to load is passed to GAMS either as an argument or stored in a input file with a fixed name. The second types are rather small files with a fixed name which typically comprise the information for which years and regions to run the GAMS program along with a small number of methodological switches. These files are overwritten with each start of the related GAMS code.

GAMS is started as a sub-process in an own thread. The output from GAMS which is typically shown in command processor window is redirected into a pipe and its content read from there and shown in a window on the CAPRI user interface, so that the user can check GAMS execution at run time. The code allows filtering out specific statements generated by GAMS to be shown in the windows title bar to give an indication about program progress.

There are two final control mechanisms. Firstly, the return code by GAMS which indicates if the GAMS program was correctly compiled and then executed. Typical execution time errors are math errors as division by zeros or read/write errors on external files. Secondly, the user can apply different type of exploitation tools to check the logical content of the results.

Exploitation tools

The structure of the GAMS generated.gdx files

The exploitation tools load directly the.gdx-files generated by the GAMS processes linked to the tasks described above. The.gdx-files only store non-zero numerical values. The main content of a.gdx file are

two types of records. The first type provides a list of all labels used to identify the numerical data in the.gdx file as GAMS does not support numerical indices, but requires character labels. The list does not distinguish for which data dimensions the labels are used. They are hence typically a mix of product, activity, region and further labels. The second type of records belongs to GAMS parameters (scalars, vectors, or multi-dimensional tables). Each non-zero numerical item in each parameter has its own record. Each of these records provides the numerical data in double precision (depending on the parameter type there may be different data stored in one record, as for a variables its upper and lower bound, current level and marginal value etc.), and a vector of indices pointing in the list of codes described above.

Loading the data from.gdx files

The data matrices generated by the different tasks as described above and stored in.gdx-files are typically rather sparse, so that it seemed appropriate to load the data from the.gdx-file into hash tables for exploitation purposes. That is done in a two step procedure. In the first step, all records from the.gdx file are read and vectors of all found indices are stored. The length of each data dimension is only known when all data records are read, and is equal to the number of unique indices for each dimension. Once all records are read, the final length of these index vectors then defines a linear index room for the multi-dimensional table. In a second step, the records are read again, and the index vectors for each record now allow to define a linear index in the total table. A hash code is derived from that linear index to store the numerical values into a hash table. As the number of items to store in the hash table is known beforehand, a rather simple hash table implementation can be used. If necessary, step one can be run over several parameters which may be hosted in several.gdx files, so that results can from different runs can be merged into one hash table.

As the.gdx-files provide a lists of all labels used in any parameters stored in that.gdx-file, the index vectors allow to build lists of labels linked for each index in a data dimension. There exists an additional storage type in the.gdx-files to retrieve long-texts to the labels as defined in GAMS set definitions. However, one label may occur in different sets with different long texts, and the.gdx-file do not store a possibly user defined relation between a data dimension of a parameter and a specific set, an option termed domain checking in GAMS. In order to link hence long-texts to the labels used for a specific data dimension, two options are possible. Firstly, the user may interactively at run time re-establish based on its knowledge the link between data dimensions and specific sets, and thus add long-texts to the labels used on that data dimension. Or the relation may be hard coded in the JAVA code.

Multi-dimensional viewer with pivoting and exporting possibilities

The multi-dimensional table is then loaded in a spreadsheet like viewer with pivot-possibilities. The user may switch between a tabular view of the data, or different type of graphs (line, bar, pie, spider) or maps. Scroll-down boxes allow the user to rotate through data dimension not shown in the view port columns and rows. Several data dimensions may be merged into one view port dimension. The user can use column and rows groups, and may apply selection to columns and rows as well as to columns and column groups. Rows carrying zero values only may be hidden. Rows may be sorted by size of the numerical values in one or several columns. The current table may be loaded into the clipboard. Alternatively, all or a selection of tables may be exported to a external file, in different format (HTML, CSV, tab-separated, GAMS, fixed width tables). There are further possibilities as changing fonts or the number of decimals.

Pre-defined views

A XML file links pre-defined views to the result content of the tasks. Each view defines selections in the different data dimensions, the view type (table, graph or map) and the pivot, plus some other information.

Graph: A pre-defined view



The screenshot shows a software window titled "Exploit scenario results". At the top, there is a toolbar with icons for "Table", "Years", and "Table". Below the toolbar, there is a "Scenario information" dropdown menu set to "2013". The main content area is a table with two columns: "BIOF_D2E2" and "BIOF_D10E10". The table contains the following data:

	BIOF_D2E2	BIOF_D10E10
Time and date	2008/02/26	2008/02/18
Recursive dynamic mode	OFF	OFF
Expost simulation mode	OFF	OFF
Policy shift mode	OFF	OFF
Baseline calibration mode	OFF	OFF
Market model for agricultural outputs	ON	ON
Market model for young animals	ON	ON
Regional break down (NUTS-level)	2	2
Base year	2002	2002
Number of Member States included	34	34
Number of regional units included	252	252
Number of inner iterations per year	15	15

Working with tables

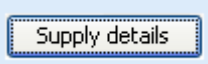
Tool bar Controls for column and row selection

BIOF_D2E2						BIOF_D10E10				
	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Yield [kg or 1/1000 head/ha or head]	Supply [1000 t or 1000 animals]	Crop share/Animal density [% or 0.01 animals heads/ha]	Production per UAAR [kg/ha]	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Yield [kg or 1/1000 head/ha or head]	Supply [1000 t or 1000 animals]
Cereals	438.65	59217.85	4958.80	293649.34	32.24	1598.78	533.21	60473.37	5142.97	311012.5
Oilseeds	454.55	8335.37	2271.59	18934.54	4.54	103.09	649.48	9607.46	2421.03	23259.8
Other arable crops	1693.24	7938.25	32743.16	259923.47	4.32	1415.16	1768.59	7761.67	33752.76	261977.5
Vegetables and Permanent	5954.93	14307.07	21918.83	313594.12	7.79	1707.37	5985.65	14297.90	21930.87	313565.4

The toolbar

View selection

The *currently selected view* is shown as description of the pushbutton in the left corner of the toolbar

under the label *View*.  Pressing the button opens a pop-up menu to *select another view*. The available views will depend on the results you have loaded. The views are logically grouped under heading, and moving on the heading will show the single views. Some views will be opened as graphics (see chapter) or maps (see chapter).



Tooltips for column and row headers

For predefined tables, tooltips may be stored which give additional information on the columns and rows. They will appear when the mouse is moved over the respective column or row header.

Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Yield [kg or 1/1000 head/ha or head]	Supply [1000 t or 1000 animals]	Crop share [% or 0.01 %]
344.87	4484.26	6228.03	8244.00	
Premiums + Revenues - variable costs according to the definition of Economic Accounts for Agriculture, income available to farmers to pay for land, labour, capital and profits				
398.37	63.48	2945.76	187.02	
1171.26	140.57	39192.45	5509.33	
30713.87	14.42	285446.28	4116.05	

Drill-down

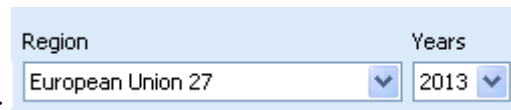
Some views comprise a hyper-links to other tables. Numbers with hyperlinks are shown in blue



, and a tooltip will appear when the mouse is moved over them. Clicking in the cell will jump to the connected table.

Navigating in the outer dimensions of the viewport

In many views, some data dimensions will not be shown in the columns and rows, but as drop-down boxes in the toolbar. Use the mouse to select in the boxes. You can use the keyboard to search items. In the



above example, these controls are:

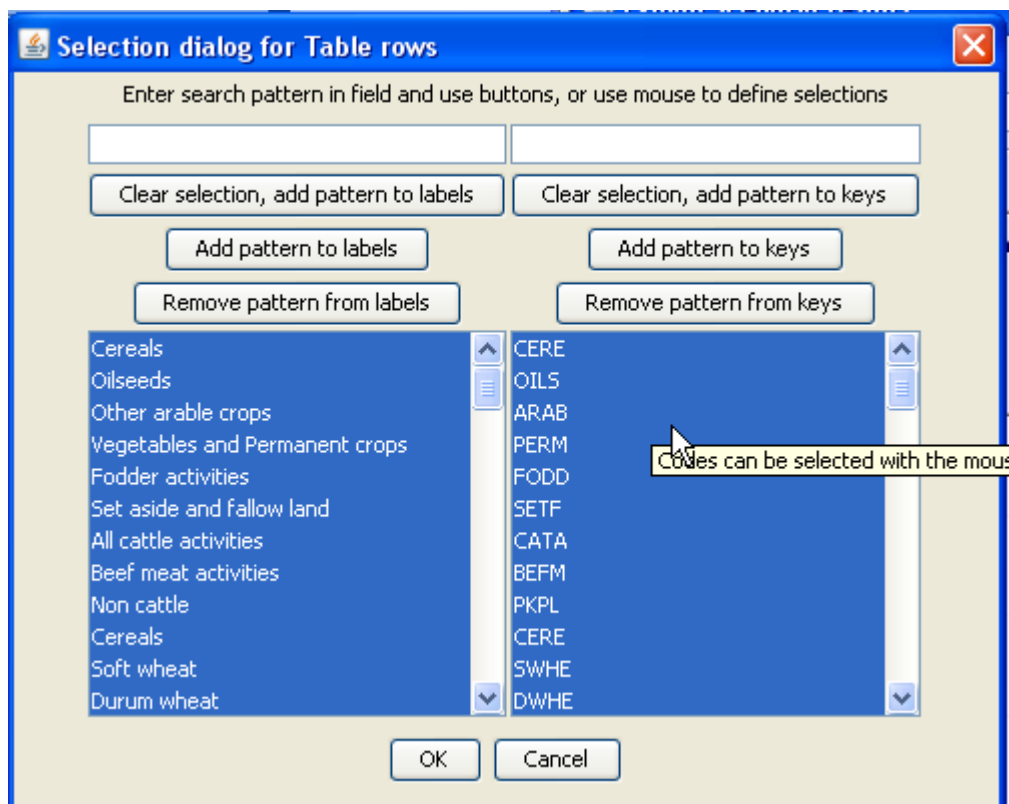
Column and row selection

Columns and rows can be hidden and included in the current view by using the buttons shown below.

The screenshot shows a table titled "BIOF_D2E2" with three columns: "Income [Euro/ha or head]", "Hectares or herd size [1000 ha or hds]", and "Yield [kg or or hea]". The rows are categorized by crop type: Cereals, Oilseeds, Other arable crops, Vegetables and Permanent crops, and Fodder activities. Callouts indicate selection options: "Selection for column groups" points to the first two columns, "Selection for columns" points to the third column, and "Selection for rows" points to the "Cereals" row.

	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Yield [kg or or hea]
Cereals	438.53	59241.46	
Oilseeds	442.66		
Other arable crops	1699.08		
Vegetables and Permanent crops	5953.01	14307.25	
Fodder activities	213.30	80038.12	



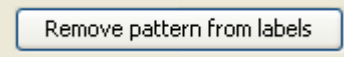
Pressing the button will open a selection dialogue:



The selections can be done with mouse, following the convention of the operation systems. Additionally, a selection string can be entered in the field above, with the following possibilities:

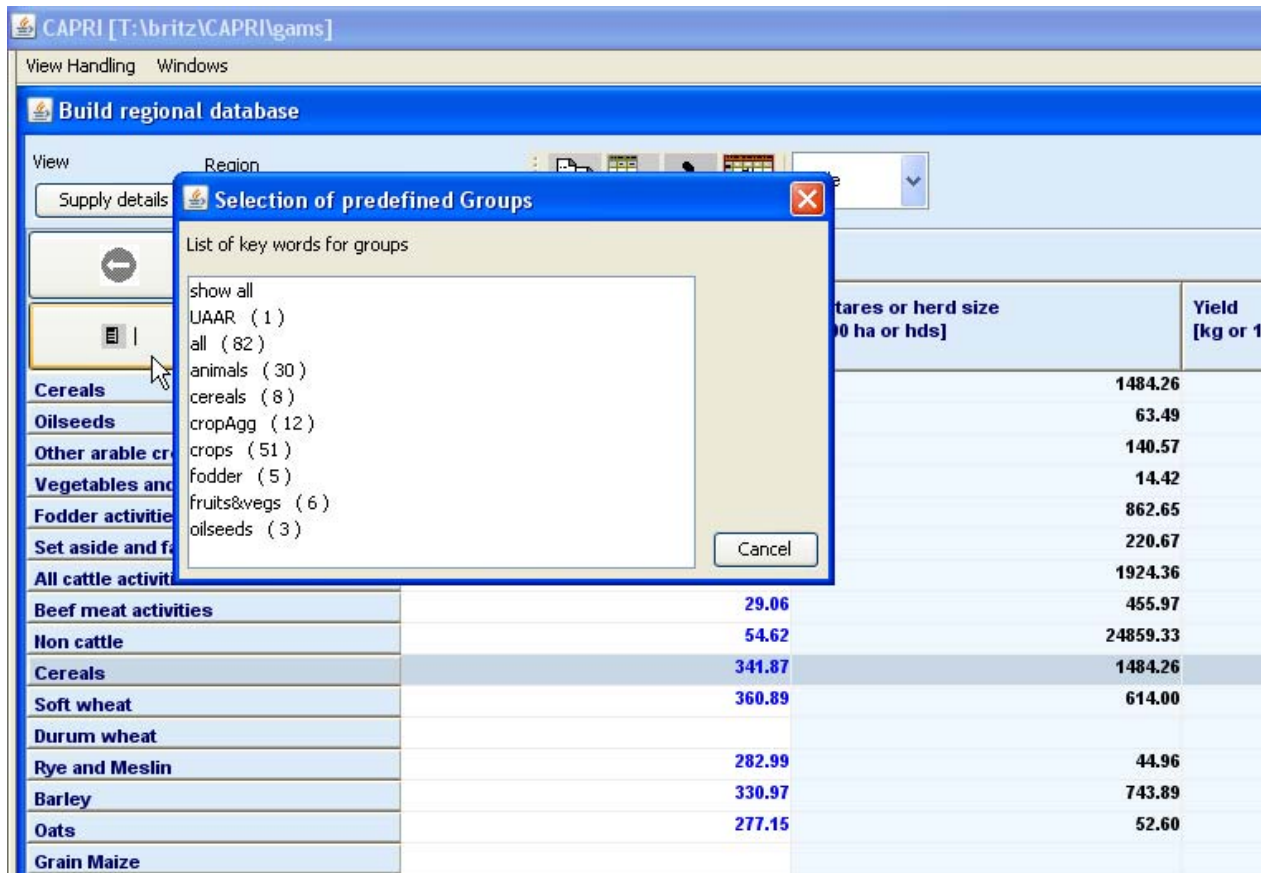
- "*" select all

- “C*” select all items starting with “C”, “C???” will select string starting with C followed by any 3 characters.

The  button will remove any selection and select only those items which match the entered pattern. Pressing  will keep the selection and add the matching items, where  will remove matching items from the selection.

Predefined selection groups

For some tables, pre-defined selection groups for columns or rows are stored. When the mouse is moved over the selection button and rests there for some time and such groups exists, a dialogue will show as below where the groups can be selected.



The screenshot shows the CAPRI software interface. A dialog box titled "Selection of predefined Groups" is open, displaying a list of key words for groups. The background shows a table with columns for "ares or herd size [0 ha or hds]" and "Yield [kg or 1]".


	ares or herd size [0 ha or hds]	Yield [kg or 1]
Cereals		1484.26
Oilseeds		63.49
Other arable cr		140.57
Vegetables and		14.42
Fodder activitie		862.65
Set aside and f		220.67
All cattle activit		1924.36
Beef meat activities	29.06	455.97
Non cattle	54.62	24859.33
Cereals	341.87	1484.26
Soft wheat	360.89	614.00
Durum wheat		
Rye and Meslin	282.99	44.96
Barley	330.97	743.89
Oats	277.15	52.60
Grain Maize		

The dialog box "Selection of predefined Groups" contains the following list of key words for groups:

- show all
- UAAR (1)
- all (82)
- animals (30)
- cereals (8)
- cropAgg (12)
- crops (51)
- fodder (5)
- fruits&vegs (6)
- oilseeds (3)


Clipboard export

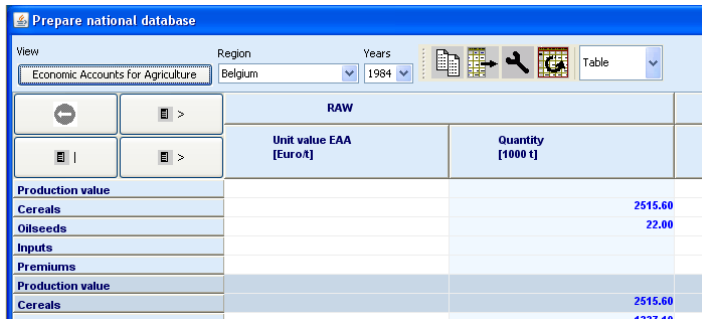


The content of the currently shown view can be copied to the clipboard by pressing the  button. Tables are placed as tab delimited text in the clipboard, so that they can be pasted into spreadsheets. Graphics and maps are placed as graphics in the clipboard and can be copied into word processing.

Export to file




A dialog opens when pressing the  button to export the full dataset of the view to file. The action provoked by the button depends on the view type. In tabular view, in opposite to the clipboard export, the export file will scroll through the outer dimensions and will copy all stacked tables after each other into a file. Take the table below as an example. Clipboard export will export the data for Belgium and 1984. File export will export data for *all* regions and for *all* years, if the user does not apply filters in the export dialog. An example is discussed on page 73.

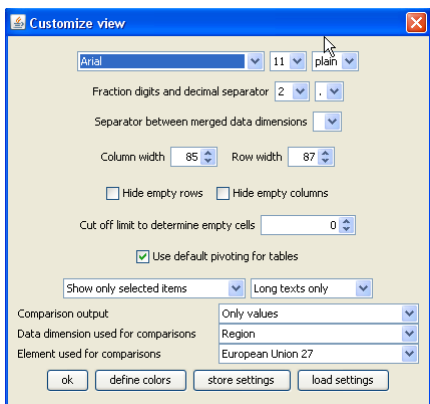


RAW		
	Unit value EAA [Euro t]	Quantity [1000 t]
Production value		
Cereals		2515.60
Oilseeds		22.00
Inputs		
Premiums		
Production value		2515.60
Cereals		4337.40

Changing view options



A dialog opens when pressing the  button to change various options of the current view:



Customize view

Font: Arial, Size: 11, Style: plain

Fraction digits and decimal separator: 2, .

Separator between merged data dimensions: [dropdown]

Column width: 85, Row width: 87

Hide empty rows Hide empty columns

Cut off limit to determine empty cells: 0

Use default pivoting for tables

Show only selected items: [dropdown] Long texts only: [dropdown]

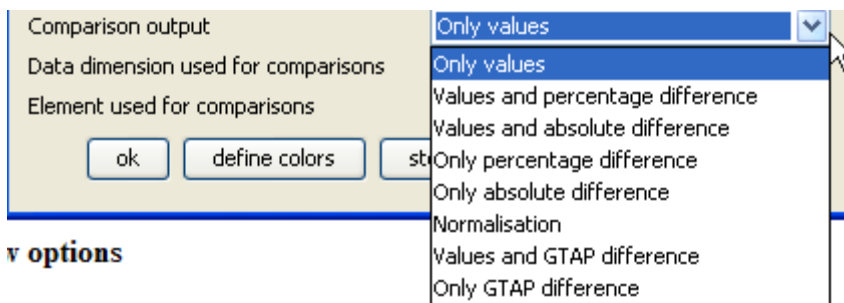
Comparison output: Only values

Data dimension used for comparisons: Region

Element used for comparisons: European Union 27

Buttons: ok, define colors, store settings, load settings

- Fonts: set font family, size and style
- Number formatting: chose number of digits and define the decimal separator. The tool supports rounding numbers before the decimal point by allowing for negative fraction digits. Choosing e.g. -1 will round all numbers to tens.
- Hide empty rows and hide empty columns will suppress in currently seen view any columns and rows which would show only blank cells.
- Use of short code and/or long texts
- Comparison output: the exploitation tools can add different types of



Comparison options

comparison output. The

“and” options will show in table two lines, one with the observations, and one with the

BIOF_D2E2				
	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Yield [kg or t/1000 head/ha or head]	Supply [1000 t or 1000 animals]
Cereals	438.65 -17.73%	59217.85 -2.08%	4958.80 -3.58%	293649.34 -5.58%
Oilseeds	454.55 -30.01%	8335.37 -13.24%	2271.59 -6.17%	18934.54 -18.80%
Other arable crops	1693.24 -4.26%	7938.25 2.27%	32743.16 -2.09%	259923.47 -0.78%

comparison output, as seen below.

The “Data dimension

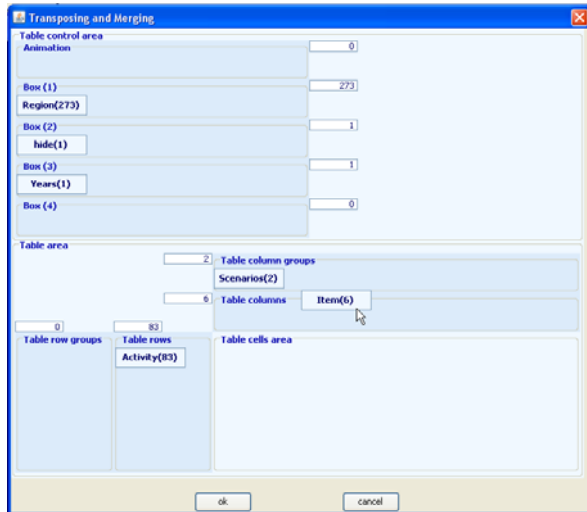
used for comparisons” offers a drop-down list” with in connection with drop-down list to select

the “Element used for comparisons” defined the comparison point.

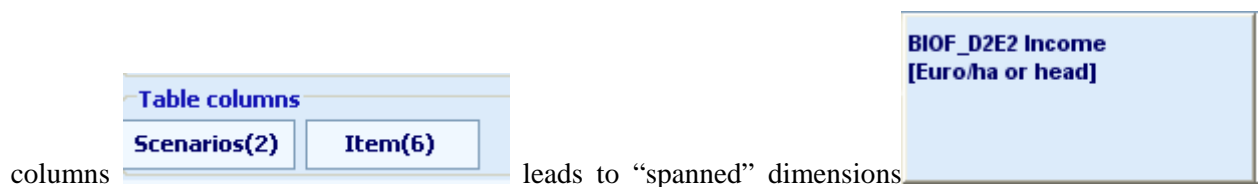
Manually changing the pivot



A dialog opens when pressing the button to pivot the currently shown or selected part of the view:



The boxes **Scenarios(2)** show the data dimension and their lengths. They can be dragged to the different viewport dimension as shown in the screen shot above. Assigning several dimensions to the



columns leads to “spanned” dimensions

		Cereals	Oilseeds	Other
BIOF_D2E2	Income [Euro/ha or head]	438.65	454.55	
		-17.73%	-20.01%	
	Hectares or herd size [1000 ha or hds]	59217.85	8335.27	
		-2.08%	-13.24%	
	Yield [kg or 1/1000 head/ha or head]	4958.89	7271.59	
		0.85%	4.37%	
BIOF_D18E1	Income [Euro/ha or head]	533.21	648.48	
		0.00%	0.00%	
	Hectares or herd size [1000 ha or hds]	68473.37	9687.46	
		0.00%	0.00%	
	Yield [kg or 1/1000 head/ha or head]	5142.97	2421.83	
		0.00%	0.00%	

Alternatively, columns and rows can have row block: In combination with the selections for columns and rows, and column and row blocks, the view can be adjusted to the need of the user, e.g. to export the data in a specific ordering to external file.

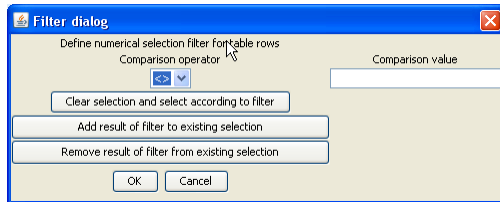
Sorting

The rows can be sorted by one or several columns by clicking with the left mouse button in the column headers. A sorting symbol will show sort direction, and its size will show the sorting order.

Cereals	Oilseeds
68473.37	9687.46
0.00%	0.00%
59217.85	8335.27
-2.08%	-13.24%
5142.97	2421.83
0.00%	0.00%

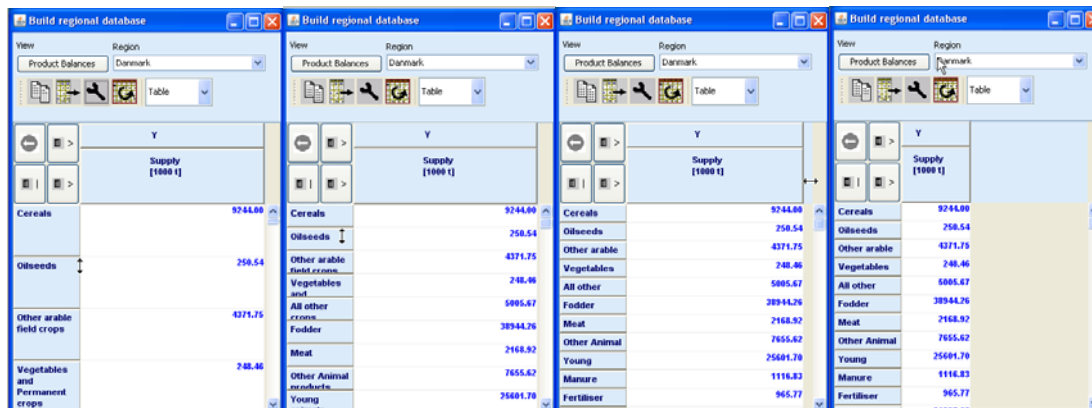
Numerical filtering based on cell content

Clicking with the right mouse button in one with the columns will open the “filter dialog” which can be used to apply numerical filters to remove rows not matching the filter from the view.



Changing the row height and column width with the mouse

When moving the mouse over the bottom of the first row header cell (Cereals in the example below), the mouse pointer will change as shown below and the cell height can then be dragged with the mouse. The column width can be changed in a similar way by dragging the right border of the column header. Alternatively, the size can be set in the “Changing view options” dialogue.



Changing the view type

The drop down box at the end of the tool let you switch between:

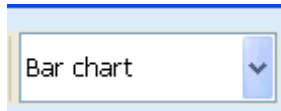
- Table view
- Different type of graphics
- Maps

Working with graphics

The exploitation tools allow showing the current content of a tabular view as a graphic.

General handling of graphs

The selection of graphs is based on the drop-down box in the tool bar:

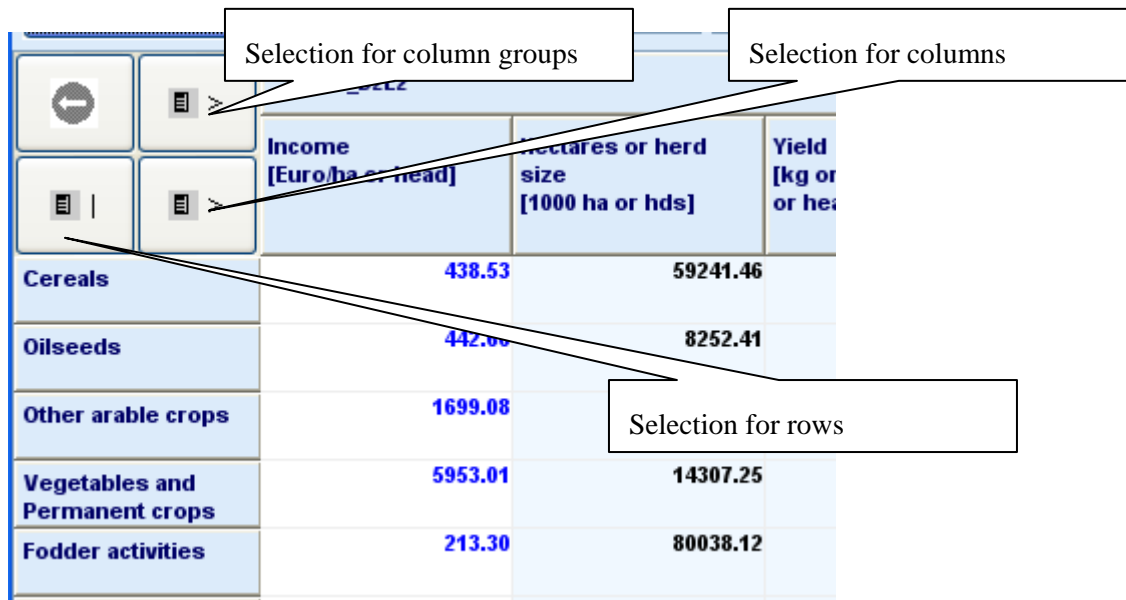


The following *graphic types* are currently supported:

- Bar charts
- Line charts
- Point charts
- Spiderweb charts
- Pie charts

The *selection* of rows and columns shown in the graph can be set in three different way, for all type of graphics:

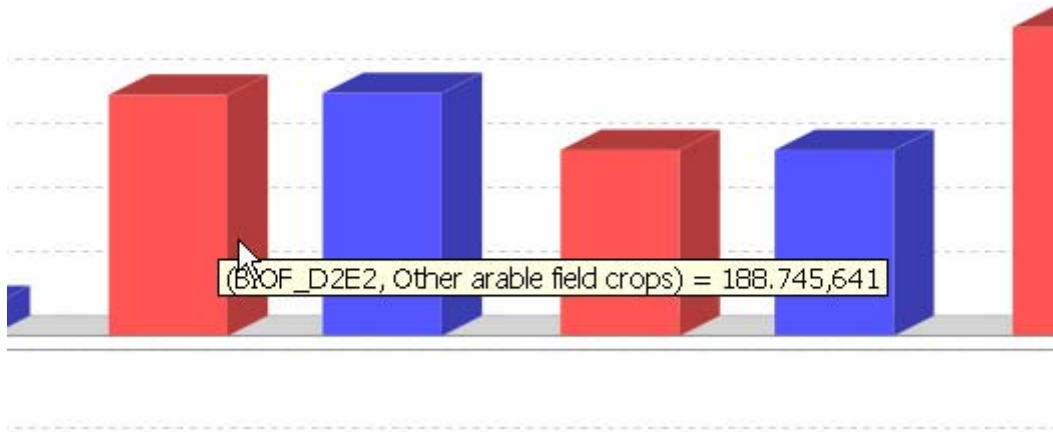
- Using the selection dialog (upper left corner of the table, or the buttons next to the graphic type selection drop down box: double click)

A screenshot of a table with annotations for selecting rows and columns. The table has columns for 'Income [Euro/ha or head]', 'Hectares or herd size [1000 ha or hds]', and 'Yield [kg or or head]'. The rows are 'Cereals', 'Oilseeds', 'Other arable crops', 'Vegetables and Permanent crops', and 'Fodder activities'. Annotations include 'Selection for column groups' pointing to the column headers, 'Selection for columns' pointing to the column selection buttons, and 'Selection for rows' pointing to the row selection buttons.

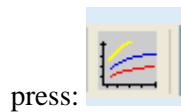
	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Yield [kg or or head]
Cereals	438.53	59241.46	
Oilseeds	442.00	8252.41	
Other arable crops	1699.08		
Vegetables and Permanent crops	5953.01	14307.25	
Fodder activities	213.30	80038.12	

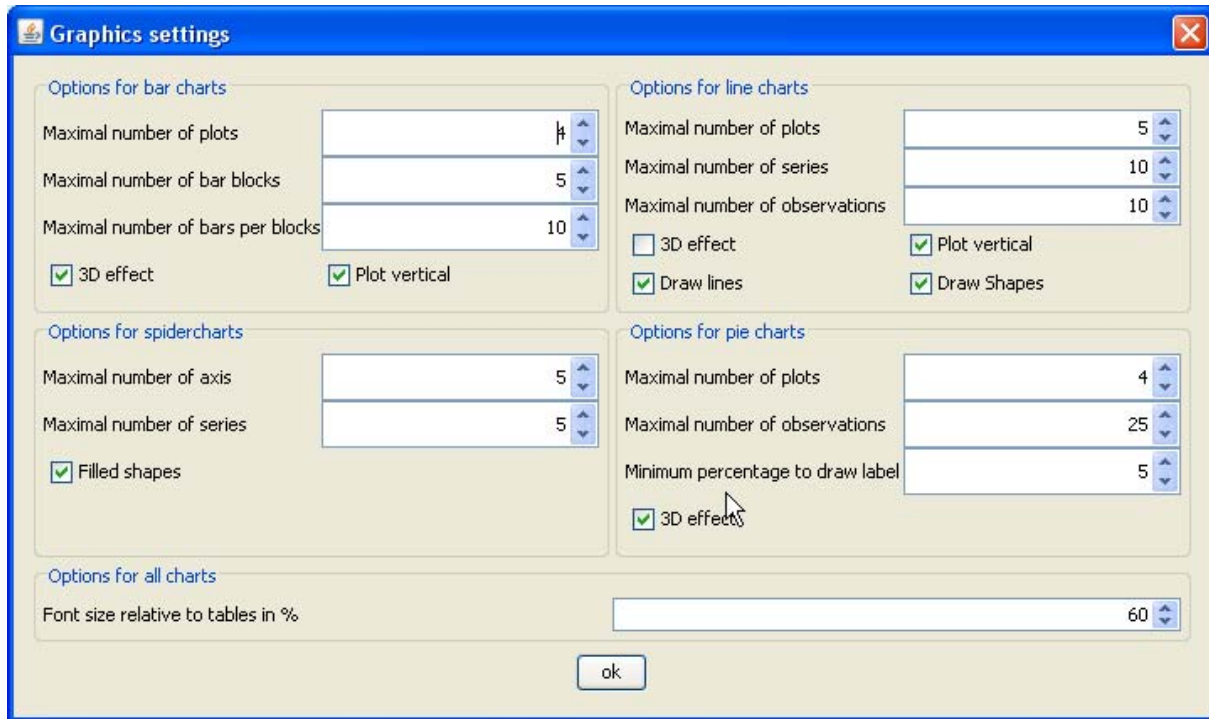
- Using those buttons in graphic mode: single clicks with the left mouse button will scroll down in the list, right mouse single clicks will scroll up.
- Scrolling the table with the scroll bar to a specific position. The column/row in the upper left corner of the table will define the starting point for the graphic.

All type of graphics support *tooltips* to query the numerical values underlying the graphic. The tooltips appear when moving the mouse on a graphic element linked to the value as e.g. a bar.



A perhaps unexpected feature is the *zooming in and out* with the mouse. The graphs support *saving* to disk via a popup menu and *printing*. The popup menu allows also changing for the current graph temporarily certain properties. Some settings which will pertain can be edited opening the graphics option dialogue,

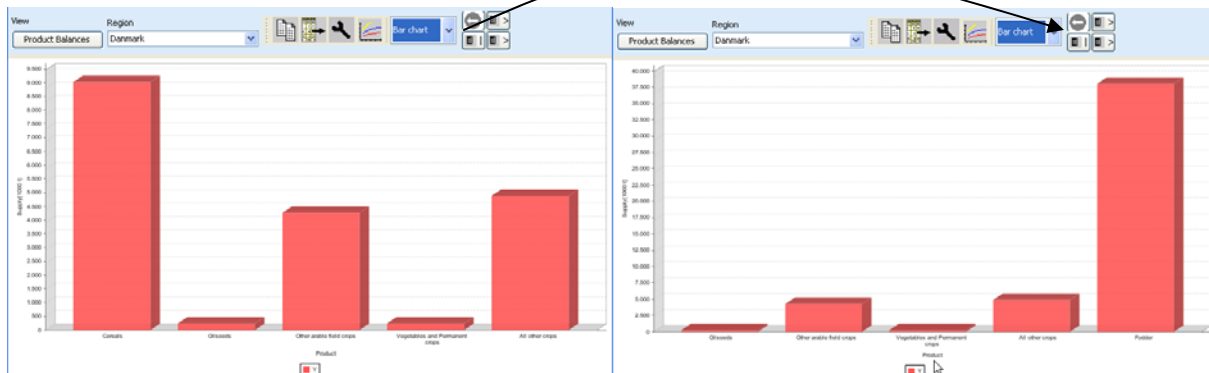




Hopefully, most of the options are self-explaining at least after reading the rest of that document.

“Walking” through the data

As not all columns and/or rows will be typically shown in a graph as the maximal numbers of elements shown is restricted (see above), the user has basically two possibilities to change the visible columns or rows. Firstly, columns and rows can be selected by the selection dialogues, see Column and row selection. Secondly, the user can click with the right or left mouse button on the buttons for table dimensions to mode one row or column up or down.



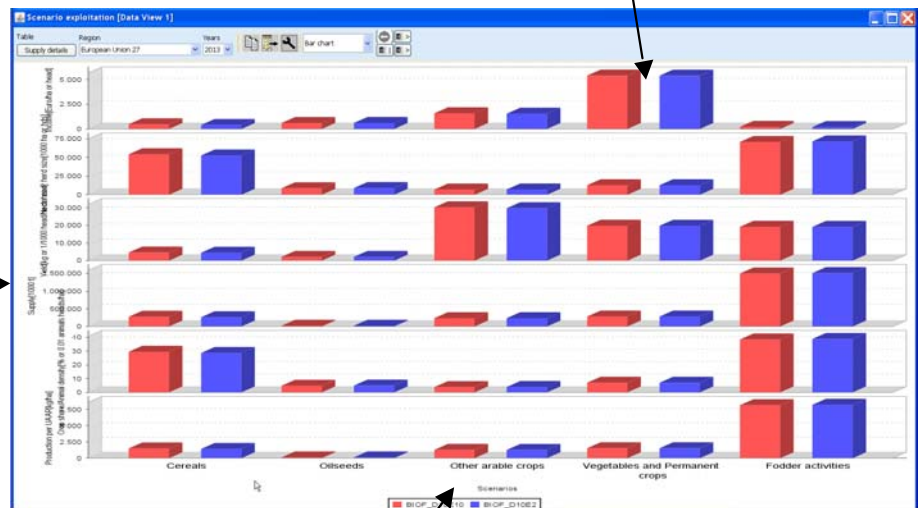
Bar charts

Bar chart treat the columns – typically the table items – as having different units and consequently assign an own plot with a value axis to each of them. The observations are taken from the table rows and define

the domain, the horizontal axis. Each column groups are present – typically the scenarios –receives its own colour. An example is given below.

Currently, there is a fixed maximum of 5 plots. It is planned to let the user change those and similar limits in the near future.

	Scenario BOP_2007					Scenario BOP_2012						
	Income (€ per ha or head)	Structures on farm size (1000 ha or head)	Yield (kg or 1000 head)	Supply (1000 t)	Crop share/Animal density (% or 0.01 animals/head/ha)	Production per 1000 (kg/ha)	Income (€ per ha or head)	Structures on farm size (1000 ha or head)	Yield (kg or 1000 head)	Supply (1000 t)	Crop share/Animal density (% or 0.01 animals/head/ha)	Production per 1000 (kg/ha)
Cereals	133.67	6888.15	8114.91	27281.54	32.71	1873.57	481.04	5870.78	4895.55	2988.19	31.75	1584.57
Oilseeds	448.79	8887.71	2388.65	25871.34	5.34	477.85	474.94	18885.86	2378.85	7491.51	5.88	138.75
Other arable crops	1718.55	7888.88	3393.87	26188.73	4.78	1474.31	174.16	7875.28	3288.81	28778.53	4.31	1487.79
Vegetables and permanent crops	5887.14	4884.21	21878.88	31888.18	7.78	1477.18	1878.18	14211.28	21881.87	21344.28	4.78	1587.83
Fodder activities	238.79	18794.75	21248.88	161888.57	41.83	1882.88	218.81	21871.87	21871.84	161728.28	41.88	1877.48
Not arable and fallow land	163.34	11284.43	154.71	288.88	7.29	11.24	43.88	13887.88	188.88	2774.88	7.88	13.77
88 cattle activities	277.88	14248.88	88.88	181.88	11.38	88.17	277.88	13882.88	88.88	181.38	11.38	43.88
88 pig activities	277.88	28888.88	288.88	18178.88	11.77	88.18	1.79	28387.87	288.29	18178.53	11.55	2.21



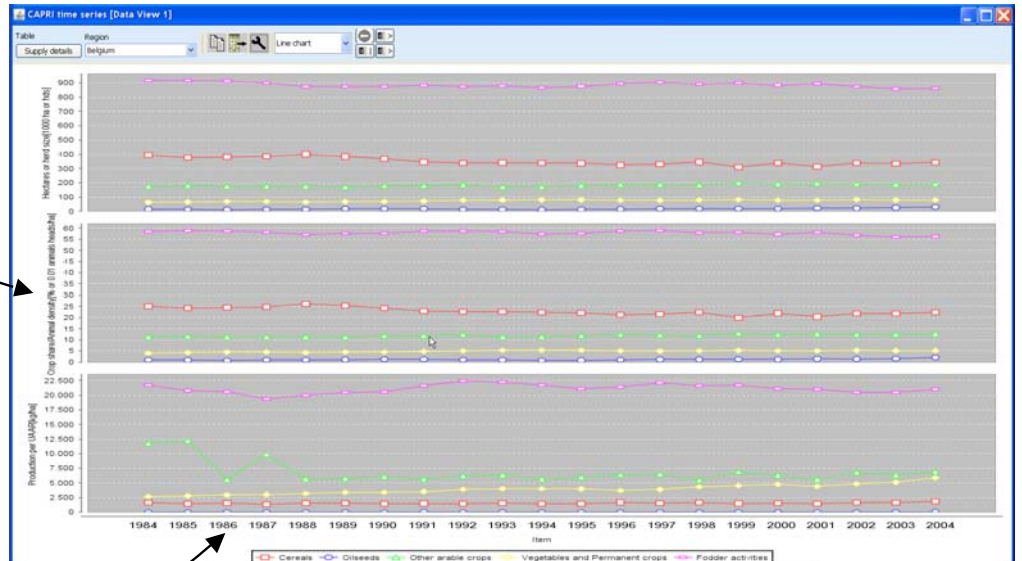
Line and point charts

Line and point charts assume that the columns of the table present some ordered sets a e.g. years or iterations. There is currently a maximum of 25 such observations set. The different series to plot are taken from the table rows. If different column groups are present, those receive their own plot with an own value axis.

CAPRI time series [Data View 1]

Table Region
Supply details Belgium

Year	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	
Cereals	15.84	15.39	14.18	14.85	15.17	15.58	15.55	15.22	15.98	15.19	15.16	15.16	15.17	15.17	15.15	15.15	15.15	15.15	15.15	15.15	15.15	15.15
Other arable crops	175.10	168.38	179.41	175.28	175.28	175.28	175.28	162.19	168.28	173.29	168.21	168.42	165.11	168.28	168.28	168.28	168.28	168.28	168.28	168.28	168.28	168.28
Vegetables and Permanent crops	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40	14.40
Fodder activities	29.48	29.17	29.42	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15	29.15
Net value added	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10
All crops	184.78	178.44	193.21	191.12	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24	191.24
Net value added	271.20	269.28	282.17	275.28	275.12	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17	282.17



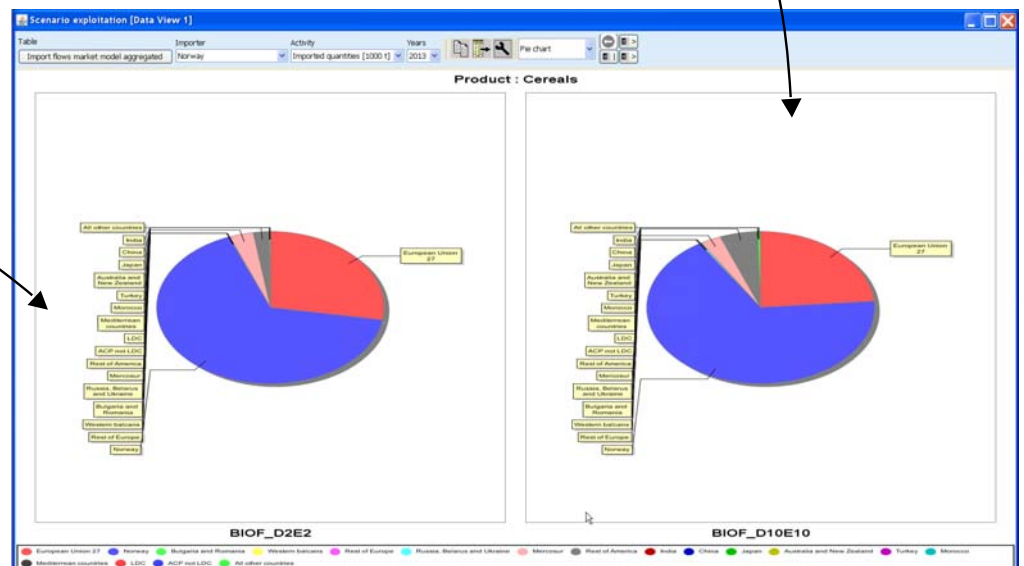
Pie charts

Pie charts are useful to show shares on total as e.g. trade flows. The shares are calculated from the columns, whereas each column group – typically scenarios – receive its own pie. Only one row is allowed.

Scenario exploitation [Data View 1]

Table Importer Activity Years
 Import flows market model aggregated Norway Imported quantities (1000 t) 2013

	European Union 27	Norway	Bulgaria and Romania	Western Balkans	Rest of Europe	Russia, Belarus and Ukraine	Morocco	Rest of America	India	China	Japan	Australia and New Zealand	Turkey	Morocco
Cereals	888.87	1187.33				2.53	56.48	38.38						
Oilseeds	9.26	24.85	5.15				413.38	8.82						
Other arable field crops	34.11	251.65												
Vegetables and Permanent crops	284.25	142.27			6.78		83.87	261.58		81.34		45.65	38.73	
Meat	2.51	298.56					8.24	8.15		8.23		8.34		
Other Animal products	8.88	42.87												



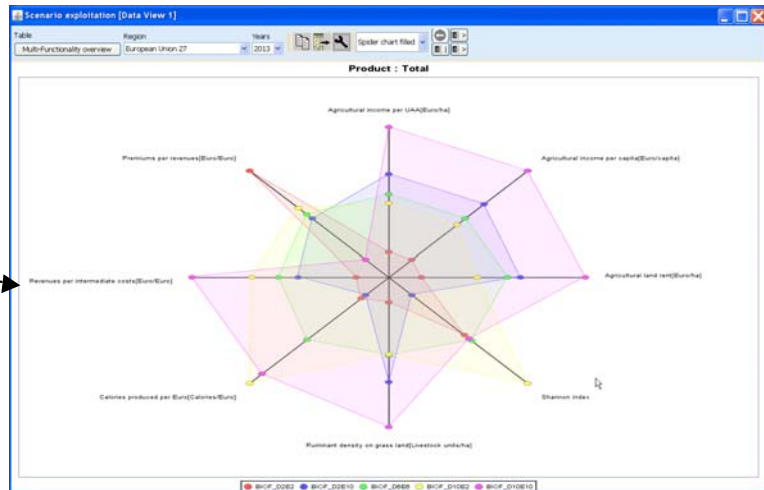
Spiderweb plots

Spiderweb charts are useful to compare several dimensions simultaneously across a range of alternatives. It is assumed that the columns show the items which each receiving its own axis, whereas the column groups are the alternatives to compare. The axis are not ticked with numerical values, instead they are always scaled to cover the minimum and maximum found in any alternative.

Scenario exploitation [Data View 1]

Table: Multi-Functionality overview Region: European Union 27 Years: 2013

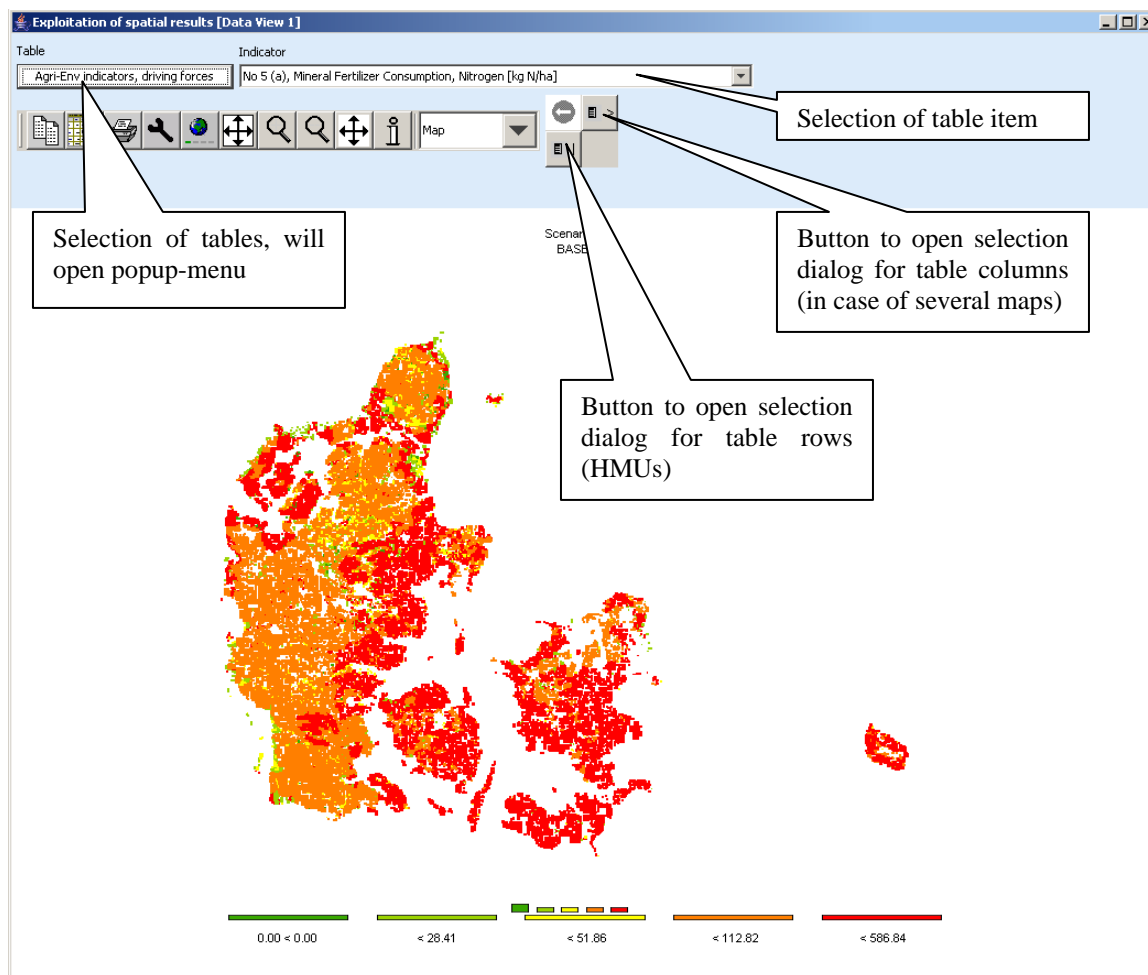
BIOF_D2E2							BIOF_D2E10					
Agricultural income per UAA [Euro/ha]	Agricultural income per capita [Euro/capita]	Agricultural land rent [Euro/ha]	Shannon index	Ruminant density on grass land [Livestock units/ha]	Calories produced per Euro [Calories/Euro]	Revenues per intermediate costs [Euro/Euro]	Premiums per revenues [Euro/Euro]	Agricultural income per UAA [Euro/ha]	Agricultural income per capita [Euro/capita]	Agricultural land rent [Euro/ha]	Shannon index	Ruminant density on grass land [Livestock units/ha]
1095.7	411.00	272.37	2.79	0.95	18.47	2.37	0.13	1138.18	427.16	282.75	2.78	0.



Working with the mapping viewer


The most obvious way to work with the results at HSMU level is the use of maps. When starting the GUI, the mapping view uses some pre-sets which can be interactively changed as described below. The following screen-shot shows the result of loading the base year results from the spatial dis-aggregation for Denmark and then switch from the tabular to the mapping view.

As with other views, the content of the map can be changed by working with the drop-down boxes, or by (de)selecting columns and rows. The map specific possibility to change class limits and colors and further features are discussed in the following.



Changing the classification and the legend

In order to change the layout of the map, click with the mouse in the area of the legend or press the map

option button . The following dialogue will open.

The screenshot shows the 'Map option dialogue' window with several callouts pointing to specific features:

- User options to change colors and classification:** Points to the 'Classification method' dropdown (set to 'Quantile') and the 'Number of classes' spinner (set to 5).
- Current class definitions (labels, limits, colors):** Points to the table below the 'Color table' section.
- Distribution graph, showing current class limits and colors:** Points to the histogram showing the distribution of data with colored bars and vertical lines for class limits and standard deviation.
- Options for output in map window:** Points to the 'Legend' and 'Standard map title' dropdowns.
- Some basic statistics:** Points to the statistics table at the bottom right.
- Options for info window:** Points to the 'Dimension shown in columns of result window for current region' and 'Dimension shown in rows of result window for current region' dropdowns.

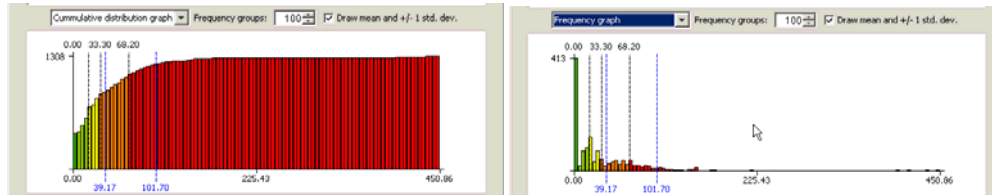
#	label	class limit	% of obs
1	0.00 < 0.00	0	30.41
2	< 28.41	28.407	17.398
3	< 51.86	51.865	17.398
4	< 112.82	112.821	17.398
5	< 586.84	586.835	17.317

n	1227
Min	0.00
Mean	55.963432
Median	30.241
Max	586.8354
Std.Dev	85.59119

It offers different options to change the way the map is drawn on screen and information supporting the classification.

Controlling the classification

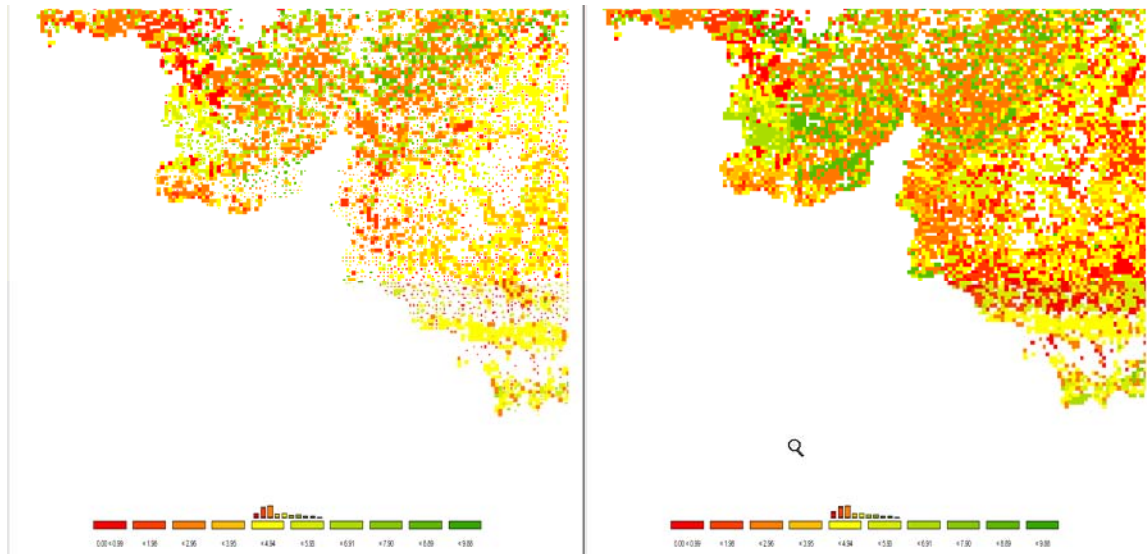
The classification can be controlled by the frequency diagram which can be either drawn as a cumulative distribution or as a frequency bar chart as shown below. The blue lines show the mean and \pm one standard deviation, the black lines the current class limits. The bars are drawn according to the current color model. The user can change the number of groups used to draw the diagrams, which does however not influence in the actual classification.



Shrinking polygons according to UAA share

The optical impression received from a map where colors are used to differentiate between values is to a large extent depending on the area covered by certain color. If the majority of the pixels is drawn in red, that will send a warning message to the user. In the case of the HSMUs and information relating to agriculture that message can be strongly biased as almost all HSMUs comprise some other land cover than agriculture, and some of the HSMUs comprise only very little agriculture, but e.g. forest, shrub lands, water bodies or artificial land cover. The HSMU geometry therefore comprises the information about the share of UAA assigned in the base year to each HSMU, and that information can be used to shrink the area of the polygons when drawn on screen accordingly. That is done by drawing all points of the polygons towards to the centroid of the polygon, by multiplying the distance between the point and the centroid with the square root of the share of the UAA. In the original HSMU geometry, such polygons had been broken down to simpler ones where the connection between a point and a centroid would cut through a segment of the polygon as in such a case, shrinking could let the new polygon to hide other ones.

The graphs below show the very same map (same input data, classification and coloring) for the High Nature Value indicator for a part of Belgium. The right hand side map draws the HSMUs in their full size, the left hand side one using shrinking. The message perceived is probably mighty different. In the unshrunk right map, one may conclude that there is a lot of highly intensive agriculture (low HNV indicator drawn in red) in the lower diagonal triangle, and some important areas of high nature farmland in the protruding area. That optical impression is different with the polygons are drawn corrected for the share of agriculture. It turns out that in the lower diagonal triangle, the density of agriculture is often low, and especially low in the intensively managed HSMUs. Equally, it turns out, that the area covered by High Nature Farmland in the protruding part is relatively small.



Area weighted classification

The classification can be generally applied treating each “region” (a NUTS II or a HSMU) as an observation with equal weight or using the areas of the underlying polygons as weights. Those weights are multiplied with the share of UAA if shrinking is used as explained above.

Excluding zeros from classification and removing small and large values

In GAMS, zeros and missing values cannot be distinguished. For certain results, zero results are therefore coded as very small numbers to allow for that distinction. Zero observation can be excluded from classification and the polygons with zero observations will not be filled. Equally, a number of regions with small and large values can be excluded from classification.

Classification method

A first important feature is called “classification method” and defines how internally the class limits are set. For all types of automatic classification methods a clean up procedure is used which removes classes with identical limits. It is generally recommended to use a number of classes which can be easily identified by the user, and to consult the frequency or cumulative distribution graphs present in the map option dialogue to check to what extent the class limits chosen represent well the data.

The following classification methods are currently supported:

Natural breaks

Natural breaks classification is a method to cluster the data into classes so that differences between the means of the classes become high while the standard deviation inside the classes becomes low (FISHER, W. D. (1958). "ON GROUPING FOR MAXIMAL HOMOGENEITY," JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION 53, 789-798. Code based on: HARTIGAN, J. A. (1975). CLUSTERING ALGORITHMS, JOHN WILEY & SONS, INC., NEW YORK. PAGES 130-142.). The algorithm does only find the approximate best solution, but gives often a rather appealing class limit definitions.

It works rather well if no extreme outliers are present in the distribution. In the latter case, classes solely comprising the outliers will be generated, and the vast majority of the values will be put in one or two classes.

The clustering algorithm is rather expensive to calculate, so that in cases of population exceeding 500 observations a somewhat simplified version is implemented in the CAPRI GUI. From the original observations, a “condensed” population is generated whose members represented means of consecutive observations of the original one. The members are set so that the number of observations from which the mean is calculated is not bigger than 1/500 of the original population size and that the spread of those observations is smaller than the minimum of 1/500 of the spread of the total population and 10% of the standard deviation. The actual calculations are then done taking the size of the resulting classes into account.

Quantile

The observations of the regions are split such that in each class fall approximately the same number of observations. Quantiles are cheap to calculate and are therefore the default setting, and often appealing as colors occupy similar areas in the maps as long as the polygons have approximately the same size.

If unique values are found at the end of a quantile, the algorithm will either exclude all observation with that unique value from the class or include all of them. The decision will be based on the fact if with or without inclusion the size of the class comes closer to the desired size. If the user has e.g. chosen five classes, the desired class size should cover 20% of the observations or area weights.

Equal interval

The differences between the current minimum and maximum value is divided into classes of equal spread. May lead to rather curious class limits when outliers are present. In those cases, it may be appropriate to

exclude some regions from the classification. See below for details how to exclude regions from the classification.

Mean standard dev

The class limits are defined according to the mean and portions of the standard deviation of the data. Works best with normally distributed data, but may result in very small classes if the distribution is skewed, e.g. long tailed. The algorithm will always introduce at least four classes, then six, eight, ten and twelve. More than twelve classes are neglected.

The algorithm takes into account the spread of the data, and sets the class limits accordingly. If all observations fall into +/-25% of a standard deviation, class limits are introduced at 25% and 10% for four classes. If the number of classes is higher, new limits are introduced at 5%, 2.5%, 1% and 0.5%. In case of +/-50%, the smallest class is dropped and +/-50% added, and so forth up to +/- 3 standard deviations.






Nested mean

The nested mean classification will only work with 2, 4 or 8 classes. The classes will be defined such that one break is found at the mean of sample. The resulting two halves of population are then again divided by their mean to get four classes, and the resulting quarters divided by their means to define eight classes. Works well with rather skewed distributions.

Manual classification

Finally, the user may set the class limits by hand. In order to do so, double click with the mouse on the appropriate row in the table with the classification results in the column "class limit". The value can now be changed with the keyboard. When done, click into another cell. The labels will be adjusted accordingly. Afterwards, when all the class limits are defined, the user may also overwrite the label (e.g. using words as "low" or "high").

Please keep in mind that currently the values will be lost if you load other data or change the classification, number of classes etc..

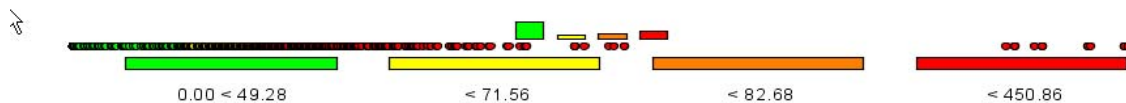
#	label	class limit	% of obs	color
1	0.00 < 0.00	0	30,488	
2	< 28.41	28.407	17,398	
3	< 51.86	51.86	17,398	
4	< 112.82	112.821	17,398	
5	< 586.84	586.835	17,317	

Integration distribution information in the map window

The GUI allows the user to enter in different ways distribution information in the map. The first possibility is to print a simple frequency diagram above the legend. That gives a rather intuitive feel on how well the class limits represent the data distribution. In our example below, it is obvious, that the majority of the values in the first class.



Less suitable for final out, but useful while playing around with classification methods and class definition are the distribution dots which can be added. They carry additionally information about the where the values of in different classes can be found.



Finally, switching to linear or logarithmic may be a way to help reading the map.



Color table

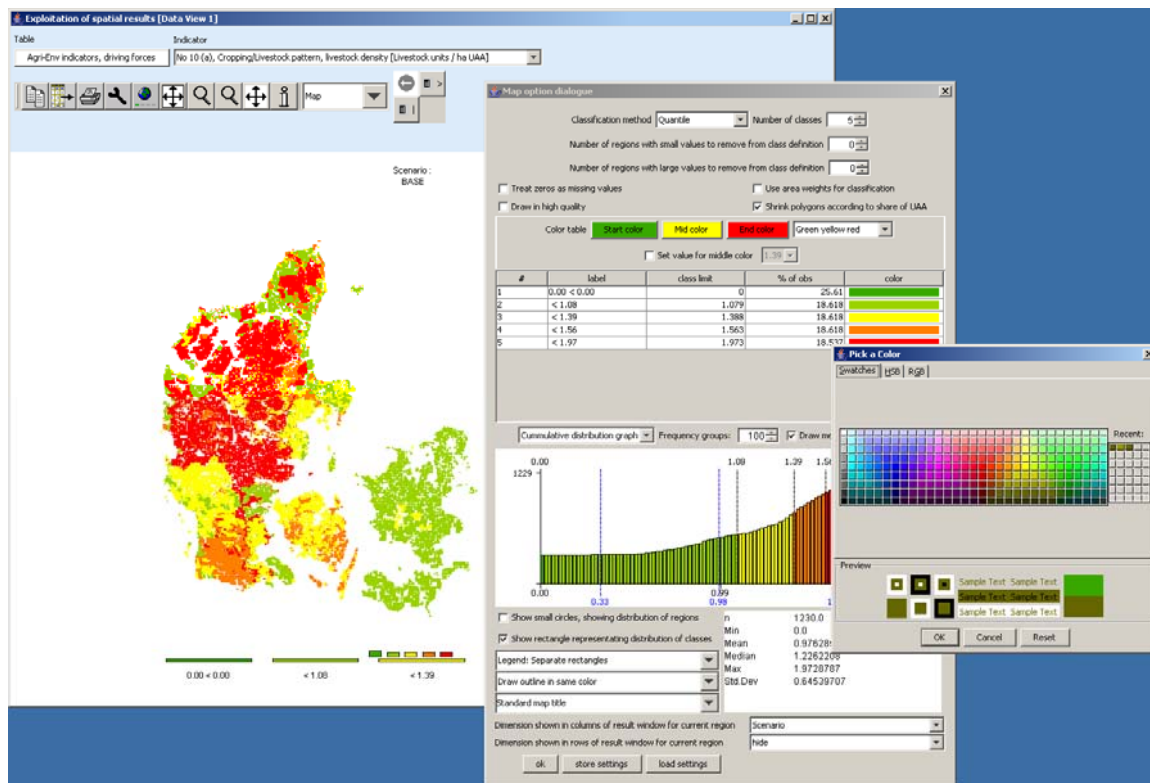
The color table defines the colors used for the classes. When choosing the color model, keep in mind that colors carry a meaning; red e.g. is generally interpreted as dangerous. Equally, it is important to think about the final medium with which the map will be published. Exporting colored maps to a black-white device will render it almost impossible to read the map. Is it best to try different color tables and different classification methods on your data. The following color models are currently available, named according to the data order from minimal to maximal value:

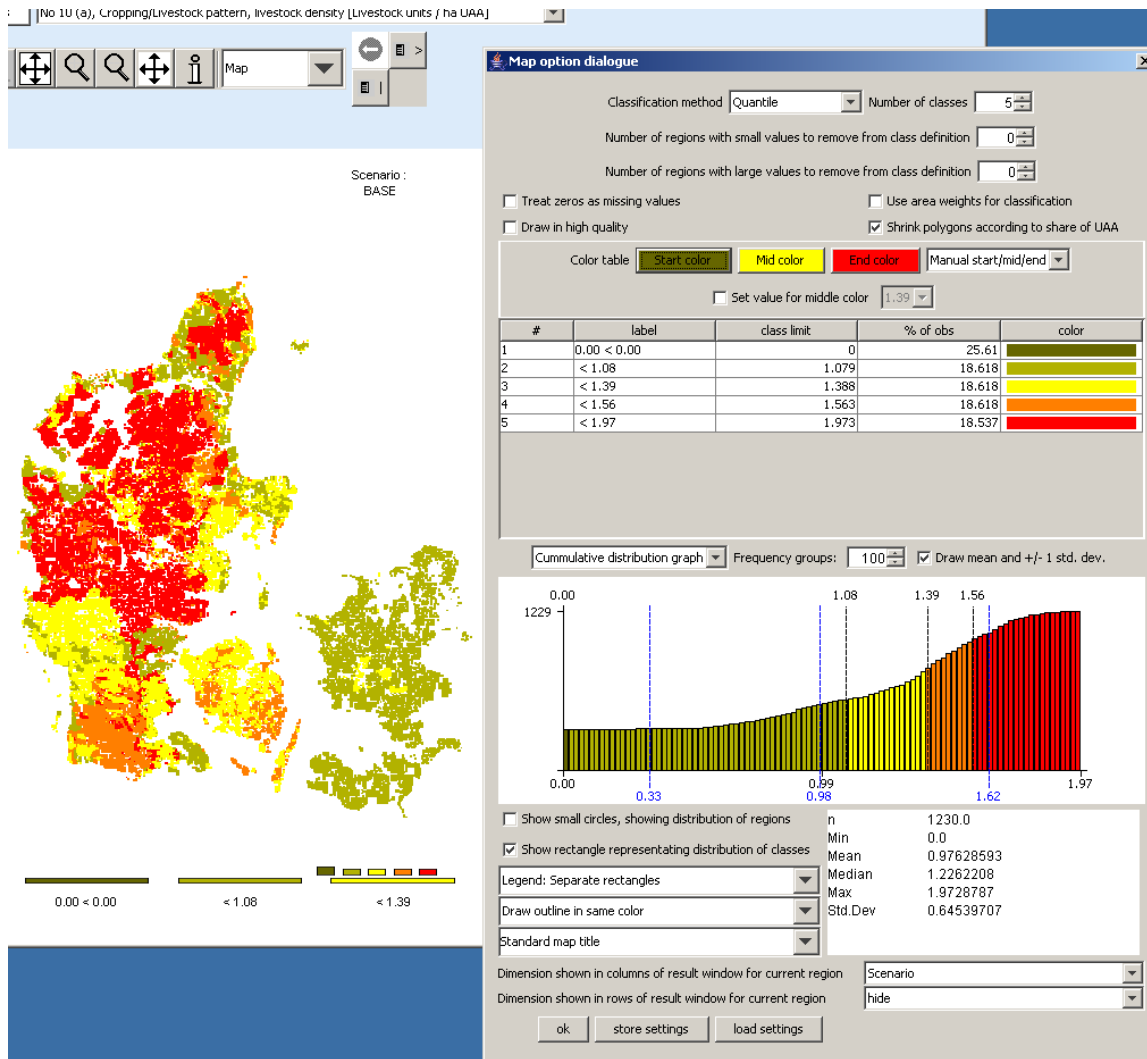
- *Green – Yellow – Red (standard)*. Normally, the middle class is drawn in yellow, smaller values in shades been yellow and green, and larger ones from green to reed. Should be applied e.g. to environmental indicators where the damage increase with the value of the indicator.
- *Red – Yellow – Green*, as above, only that high values are shown in green. Should be used e.g. for income indicators or environmental benefits.

- *Red – Gray - Green / Green – Gray – Red*, more available for historic reasons as they mimic the color tables of the original JAVA applet.
- *Blue – Gray - Green / Green – Gray - Blue*; introduced on demand of DG-AGRI. A good choice if the “good”/”bad” interpretation of the distribution is to be avoided.
- *Shades of grey*; sometimes needed for publications when color printing is available in the final hardcopy. Beware to use a limited number of classes.
- *Shades of blue*; useful where the notion or “bad” or “good” inheritably comprised in greenish and reddish colors is to be avoided.

Defining an own color model

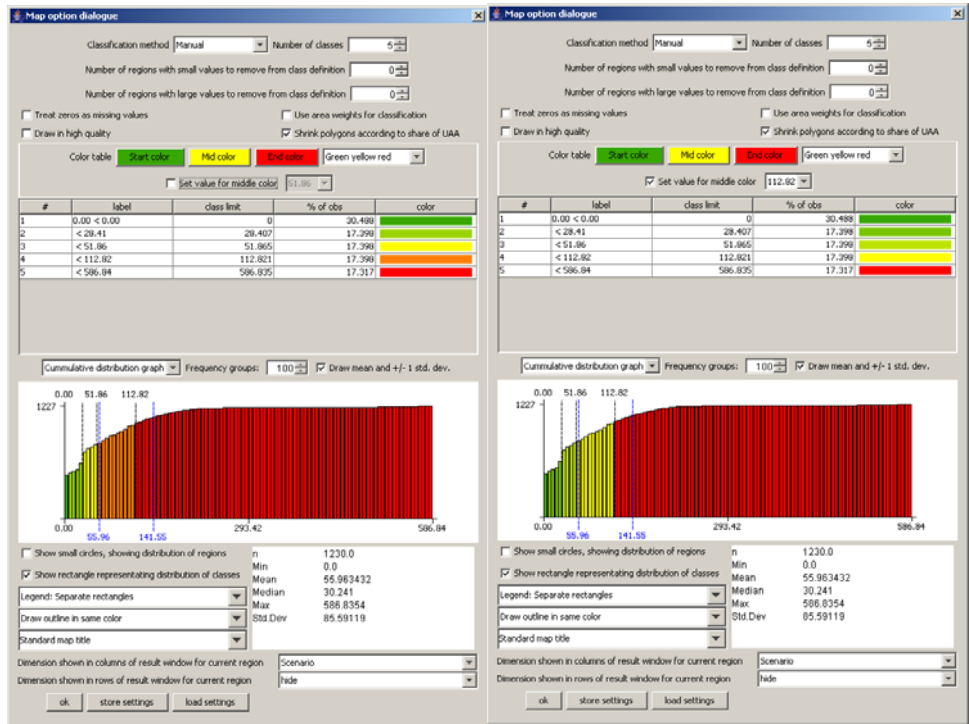
Once a color model is chosen, the user can re-define the start, middle and end color using the three buttons below the color table selection row, as shown below, given a lot of freedom to generate color ramps.





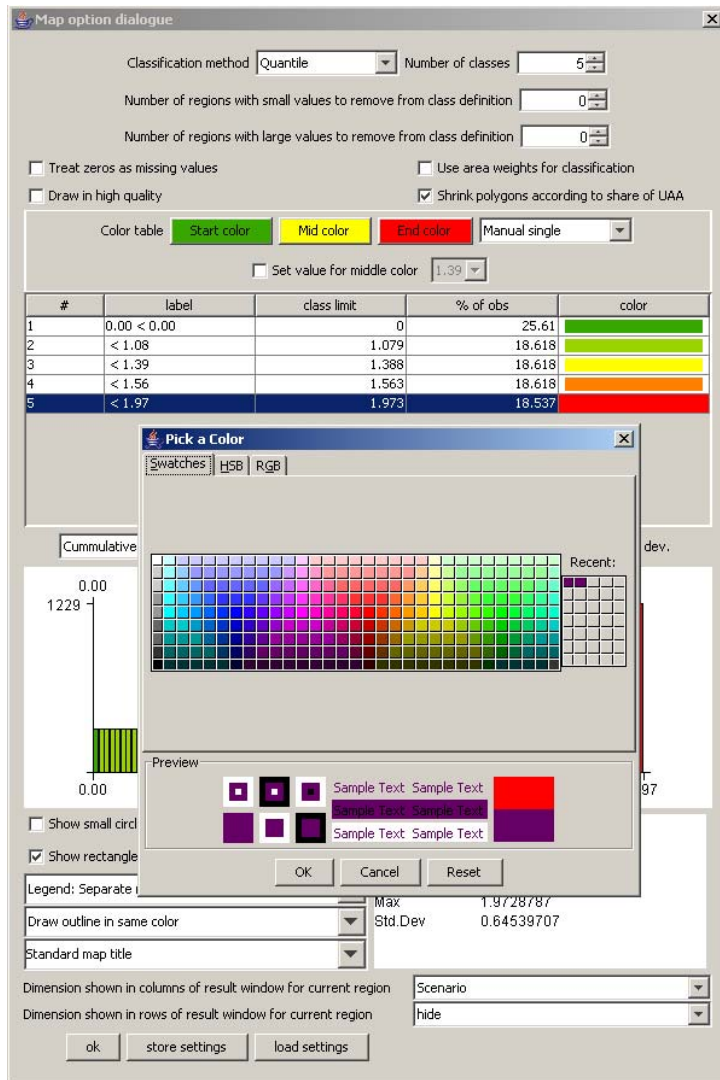
Changing the value for the medium color

Normally, the medium color (yellow or gray) is assigned to the middle class. Sometimes, the user may wish to change the class where the color switches. First, the “Set value for color change ..” must be ticked on. Next, in the now enabled drop-down box, choose the class limit for which the middle color should be used. The effect is shown below. Whereas before, values in the class below “392.70” – the middle class – was drawn in yellow. When the user now selects another class limit, the colors assigned to the classes change, here one of the shades of green is dropped and shade of red is added.



Manual set colors

Finally, the user can choose its own colors by double clicking in a color field in the legend table. That should only be done after the final definition of the class limits is set as otherwise, the manually set color will be lost.



Changing the way the legend is drawn

The map viewer puts the legend always below the map, and currently offers three options how legends are drawn:

1. Separate, equally sized rectangles which show the upper class limit with the exemption of the lowest class, which shows the lower limit.



2. A continuous linear scaling bar. That gives an optical idea about the how distribution of the class limits. Overlapping of the number of avoided.



3. A continuous logarithmic scaling bar .



In all the cases, the tool dialogue can be used to set number of digits shown, e.g. reducing the number of digits to zero leads to a linear bar as shown below:





The reader is reminded that the label can be changed manually as shown below.

#	label	class limit	color
1	low	345.891	yellow
2	high	1,392.846	red

Cumulative distribution graph: n = 249.0, Min = -418.65223, Mean = 358.84164, Median = 345.89148, Max = 1392.846, Std.Dev = 4468.262.

Dimension shown in columns of result window for current region: Years
 Dimension shown in rows of result window for current region: Activity

Copying the map to the clipboard or saving to disk


In order to export the map to other applications, the easiest way is to use the clipboard, in order to do so, press the “copy to clipboard”  button. Afterwards, the current map can be imported into other applications as e.g. MS Word. Another possibility is to store the current map in jpeg format on disk, to do so, use the “export”  button which will open a file dialog to choose the name of the file.

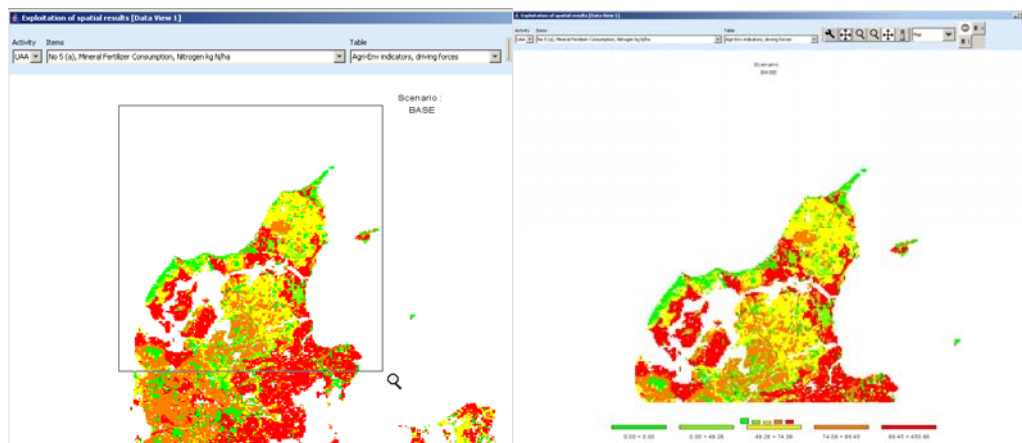
Changing the title of the map




When using output to clipboard or disk, the user may often prefer an own title or no title at all on top of the map in order to produce a caption for the map in another application. In order to refrain from drawing a title on top of the map, click into the legend part of the map, and in the dialog at the bottom, choose “none” in the row labeled “Title on top of map”. Alternatively, the user can simply write something in the box.




Zooming in and out and navigating in the map

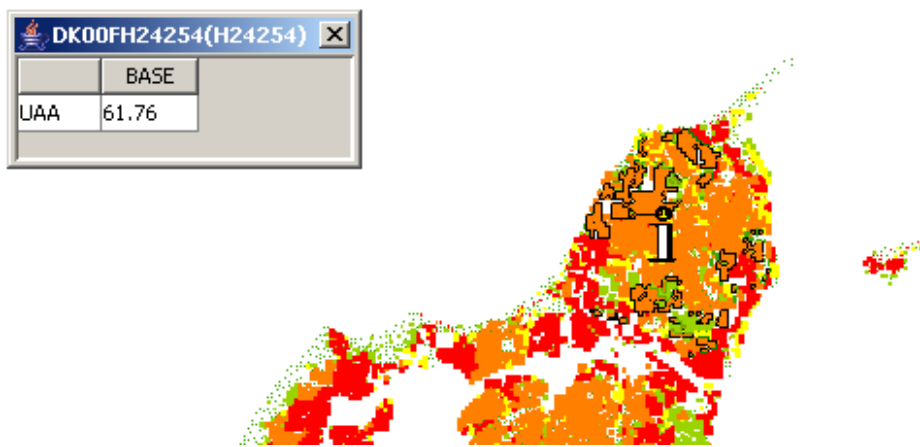
In order to zoom in part of the map, press the  button. The mouse pointer will change to a magnifying glass with a cross in it. You can then mark an area on the map by pressing the mouse button, dragging and releasing the mouse. After the mouse is released, solely the selected zone of the map will be drawn, without changing the class limits or any other setting. Clicking with the mouse while in zoom in mode will increase the map resolution step-wise by 25% and center the map at the current mouse position.



By clicking with the zoom out pointer  on a point of the map, the point becomes the new center point of the map and the map resolution is reduced stepwise by 25%. Equally, you may drag the map while keeping the current resolution by choosing the drag pointer . And finally, in order to return the original full sized map, use the “full extent” button .

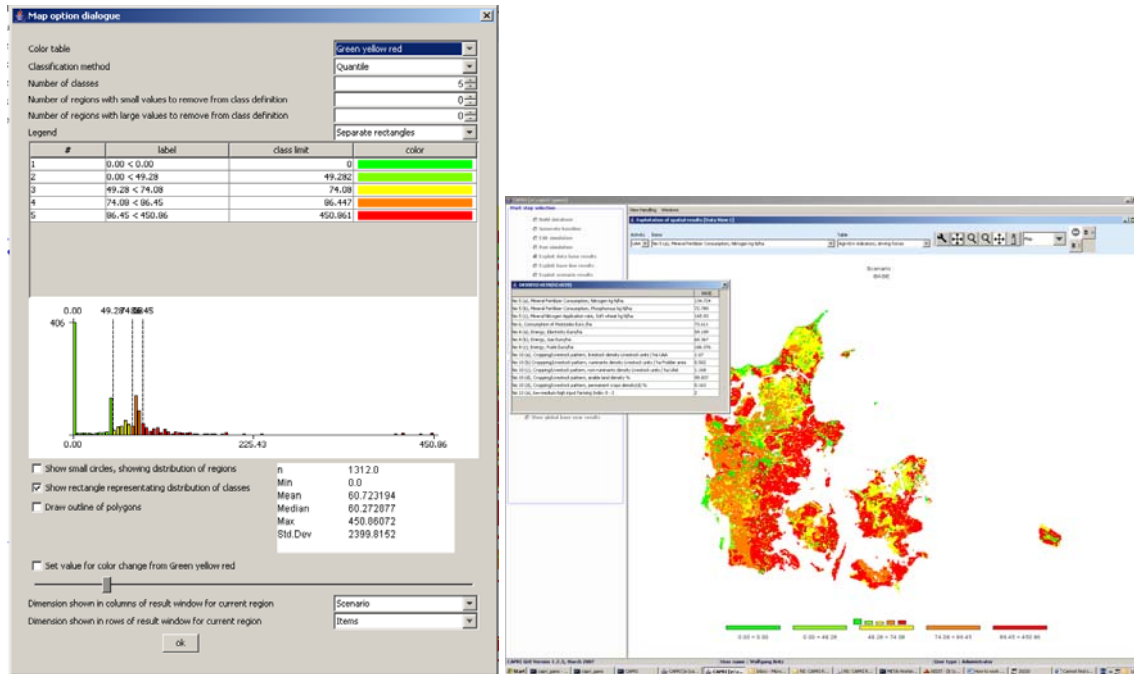
Getting data for specific polygons

The info pointer  will open an additional window as shown below which displays information on the current polygon – the circle above the “i” being the focus point. The new window shows in his title bar the code and, if available, the long text of the polygon currently pointed to with the info pointer. The content of the info window is continuously updated when the mouse is moved over the map, and all polygons belonging to the same region as the one where the mouse is over are highlighted.




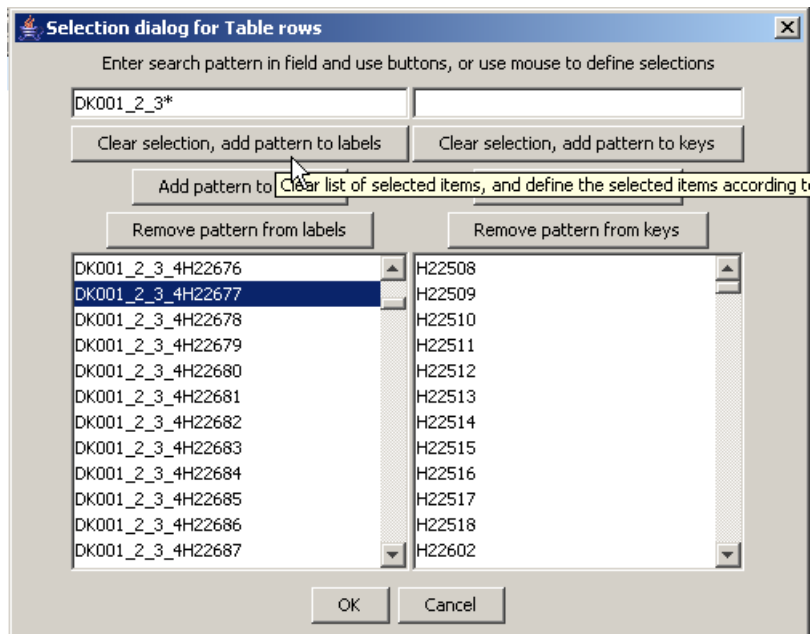
If the user opts to use one of the comparison options to be shown (percentages, differences, normalization) by clicking on “customize” button, an additional column is automatically added to the info window showing the comparison value used. That is especially helpful when the map shows only differences.

The content shown in the info window is not fixed, rather, the user can decide which data dimensions to use for the columns and rows by using the “map option dialogue” by clicking on the legend of the map. If the user e.g. switches to “items” instead of “activity”, the “info” window will look like shown below. An alternative is to use a second tabular view in addition to the map.

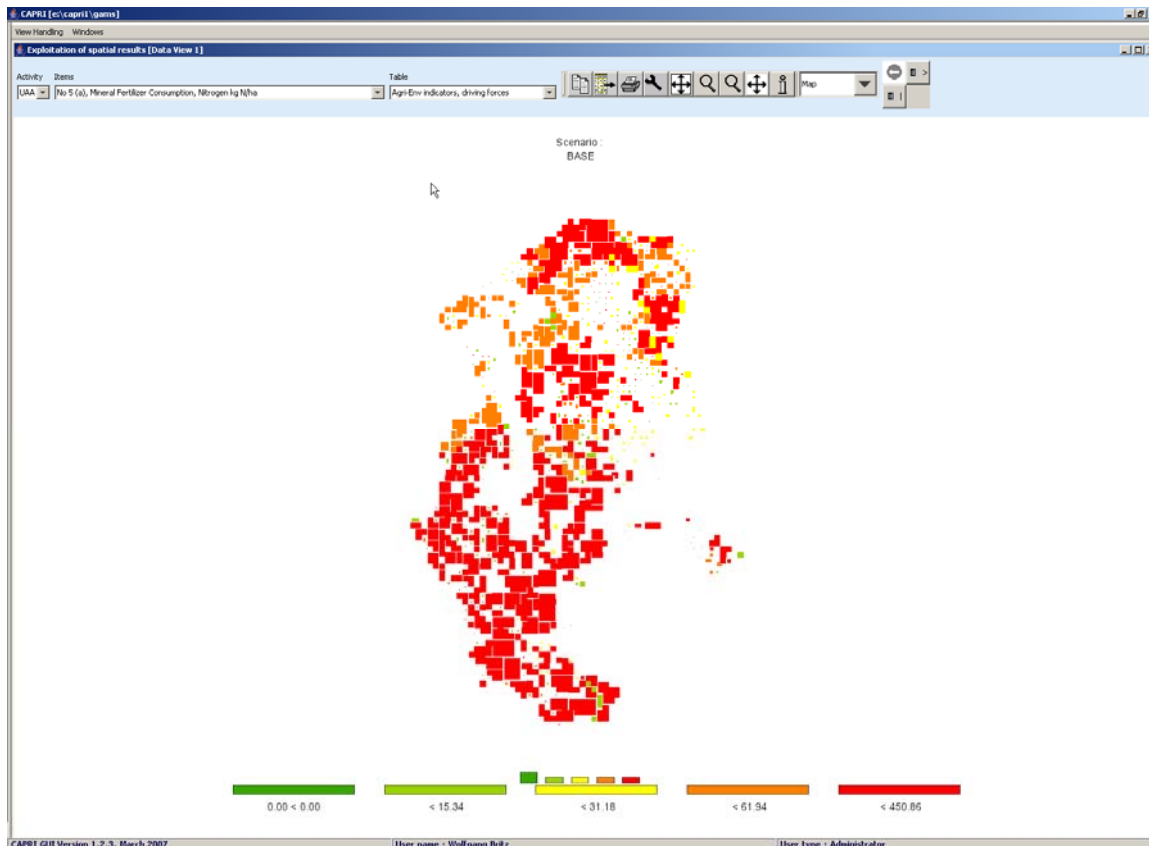


Highlighting specific regions in the map

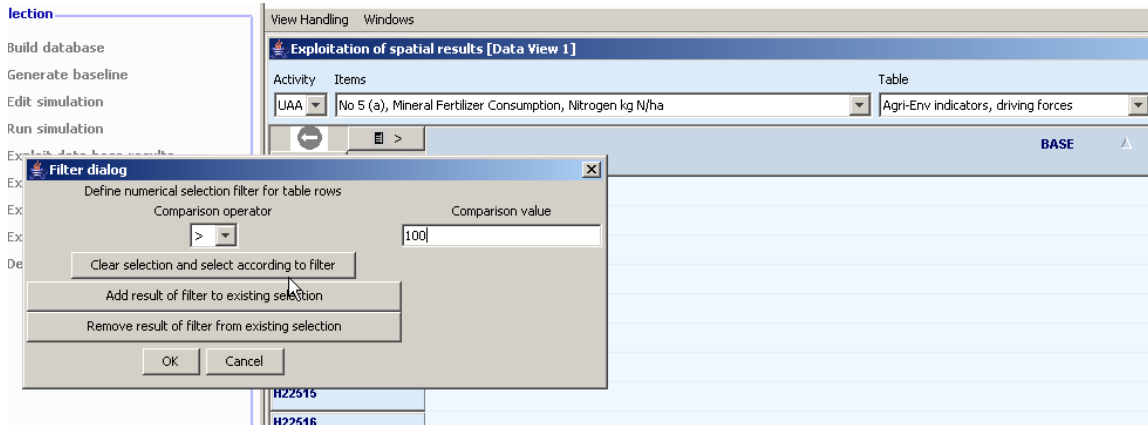
Sometimes it may be interesting to see the spatial distribution of specific data or data constellations. All views open the possibility to (de)select columns and rows, allowing e.g. to use the NUTS code in front of the numerical HSMU code to select only the HSMU belonging to specific administrative regions. That possibility is explained in short. First, press the row selection button  (“Open selection dialog for table rows”) which will open the following dialogue.



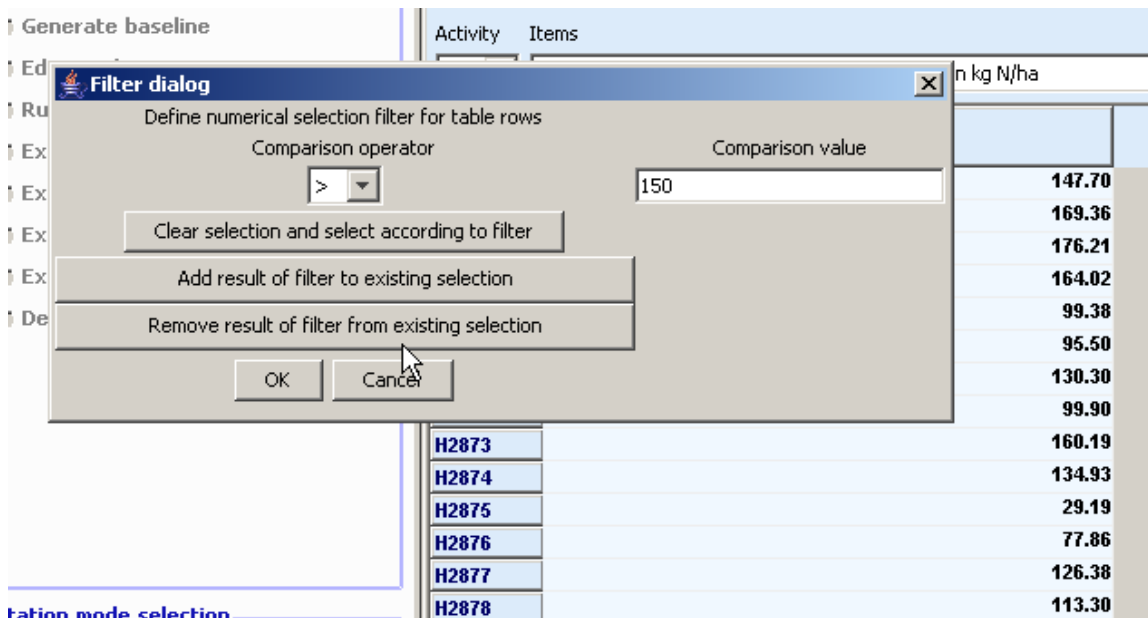
Now, we may e.g. selected only the HSMU belonging to the FSS region DK000_1_2_3 by typing “DK001_2_3” in the left input box, and then choosing “Clear selection, add pattern to labels”. Afterwards, the map will look as shown below.



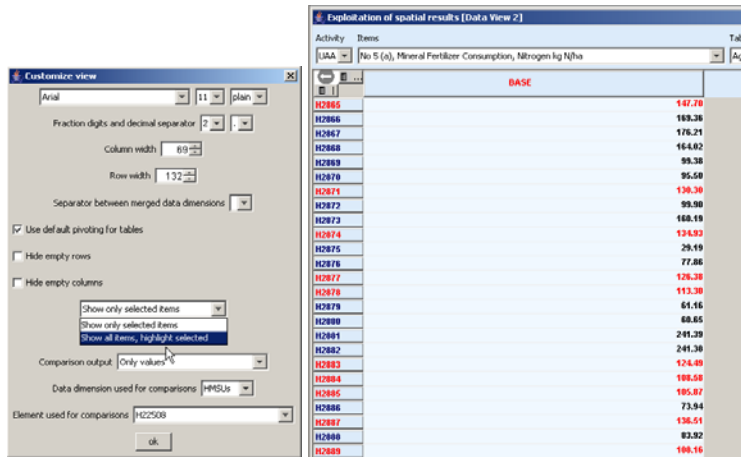
The tabular view opens up the possibility of using numeric filters, an option discussed in the following. Take as an example the task to select all regions where the Nitrogen Fertilizer Consumption is between 100 and 150 kg/ha. First, switch from map to tabular view. In the table click with the right mouse button in the column header of that column holding the values to which the filter should be applied, as shown below. We will need to apply the filter step-wise, first e.g. selecting all values greater than 100 and then removing those which are above 150.



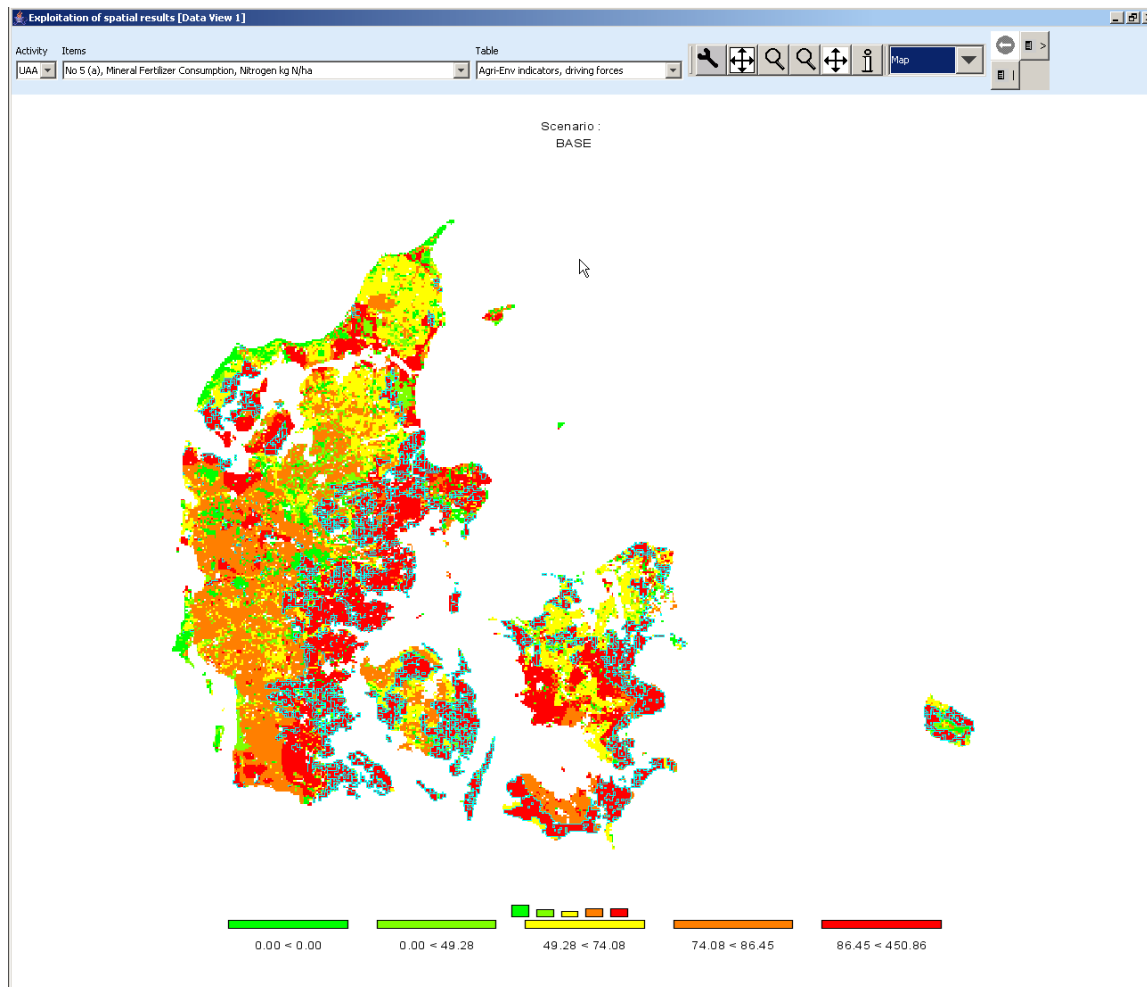
After pressing on “clear selection and select according to filter”, and then on “ok”, the table will only show such regions where the value in the column “BASE” is above 100, as shown below. Next, we must exclude the regions above 150 kg/ha. To do so, set the filter to “>” “150”m and then press “remove result of filter from existing selection”.



Now, drawing a map with just those regions is not so interesting. However, with the tool dialogue, we can highlight the selected value instead of hiding all others. The selected rows are now shown in red in the tabular view.



When we now draw the outlines of the selected polygons only (see map option dialogue), the map will draw the outline of the selected regions in cyan and thus highlight them. The row selection will be maintained when the pivot or the table is changed as long as one of the selected items can be found in the rows of the new table. The example map shown below is certainly not so interested, as changed class limits could have done basically the same job. However, we could switch e.g. to grass land shares to see if that fertilizer input is more often found on arable or grass land.




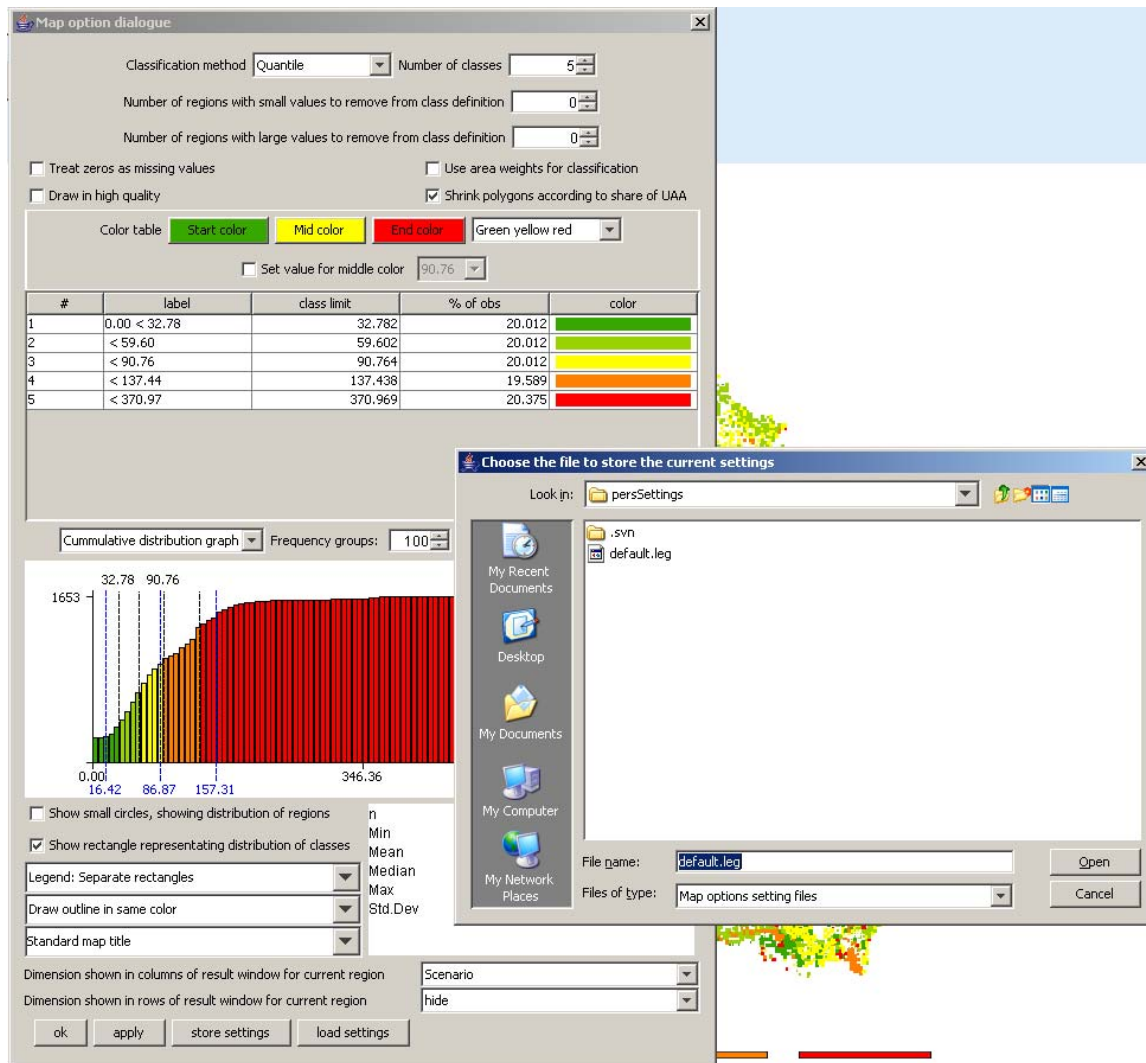
Updating the map

Generally, the map is updated automatically when the user changes an option impacting on its layout as long as long as the number of visible polygons is below 20.000. If that amount is exceeded, the classification dialogue is updated immediately, but not the underlying map. In order to apply the changes, the “apply” button must be pressed. The user is informed that the “ok” button will also update the map, so that an “apply” immediately before an “ok” is not necessary.


Storing and re-loading your settings

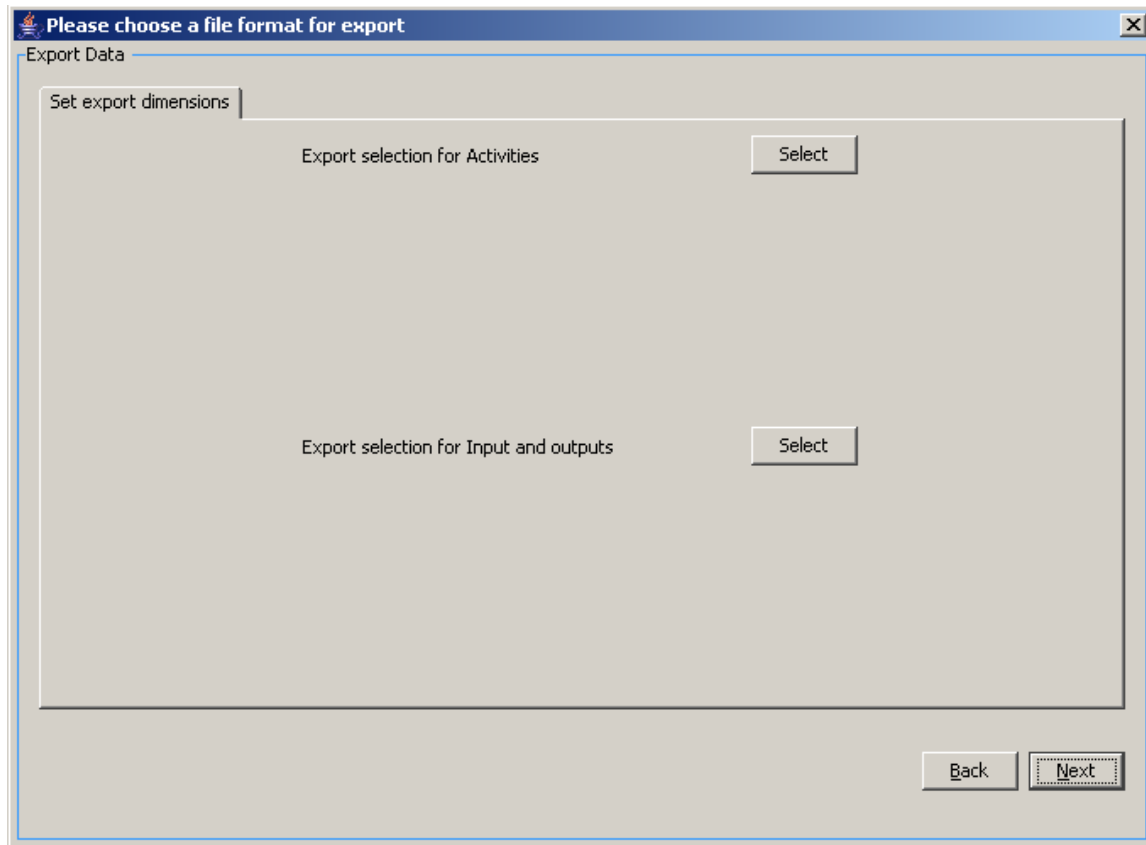


Open the map option dialogue by pressing the map option button “”. Change the settings according to you needs, and then press the “store settings” button in the lower part of the dialogue. Choose a file name and location. You may later used “load settings” to retrieve them again, and apply them to another map.

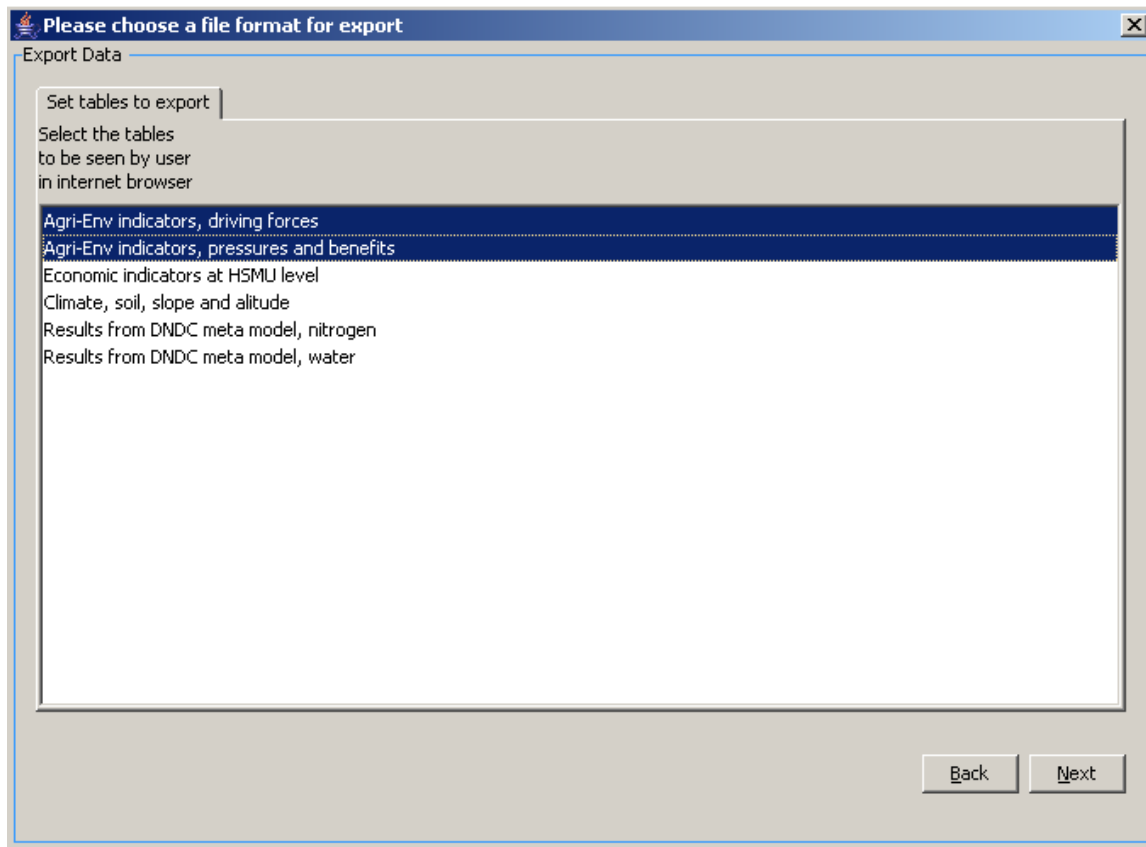



Exporting the data underlying the map

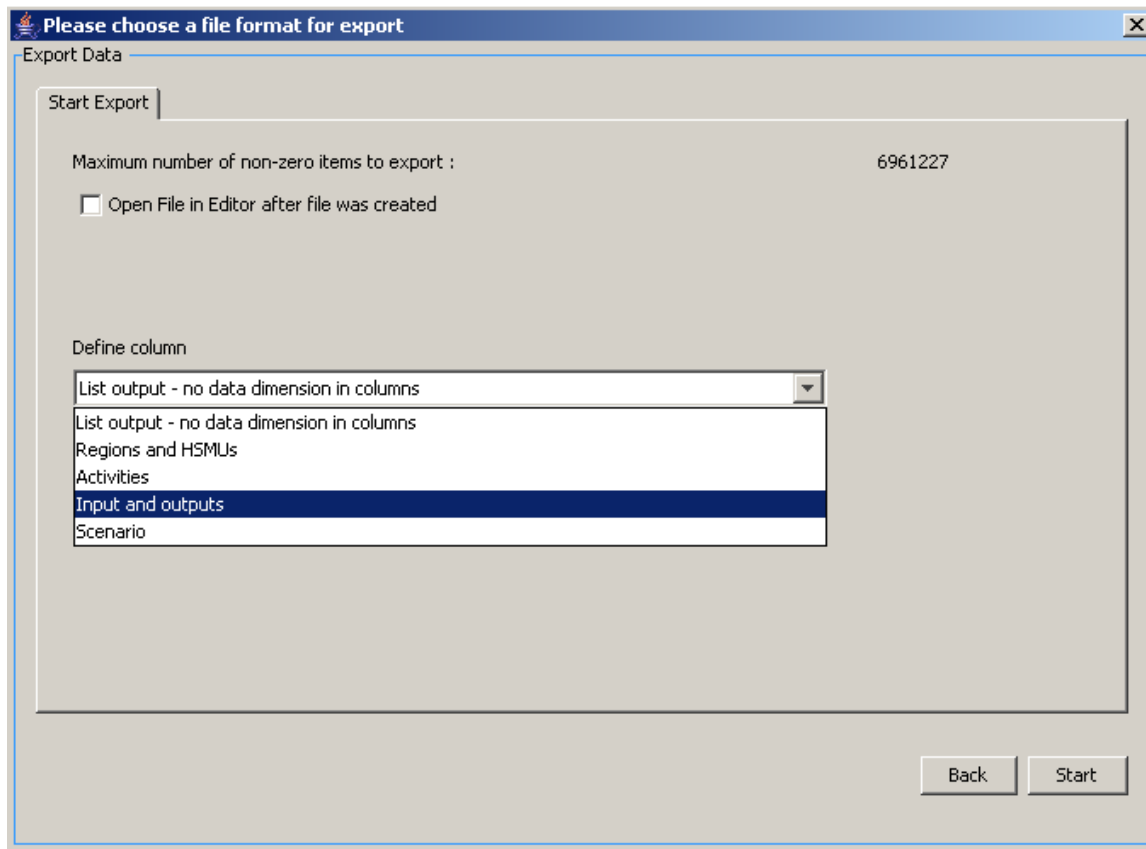
As mentioned above, the mapping viewer is part of the CAPRI exploitation tools which is in its core based on pivot tables. In order to export the data, e.g. to GIS system, the view must first be changed to tables. Afterwards, the  button which will open a file dialog as shown below. For GIS-export, e.g. to ArcGIS, DBF is the recommended format.



Next, the tables for export can be selected.



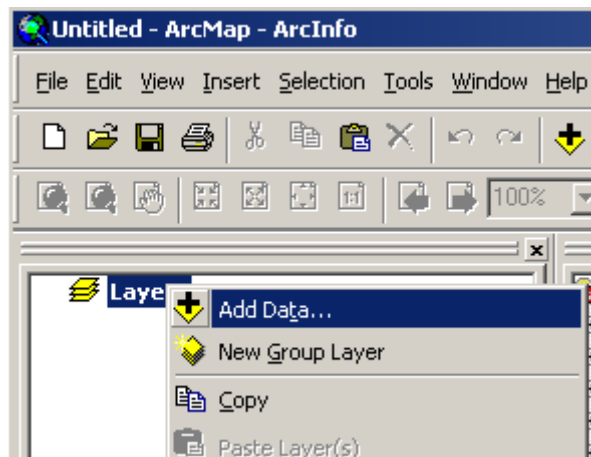
Beware: the pre-defined table structure will be lost, as will the long-texts and units attached to the tables. However, in the case of DBF-export, a second file with that information will be automatically created. If you solely want to export the table you have currently up front, use the “copy to clipboard”  button. The clipboard export will retain the pivoting and further information.



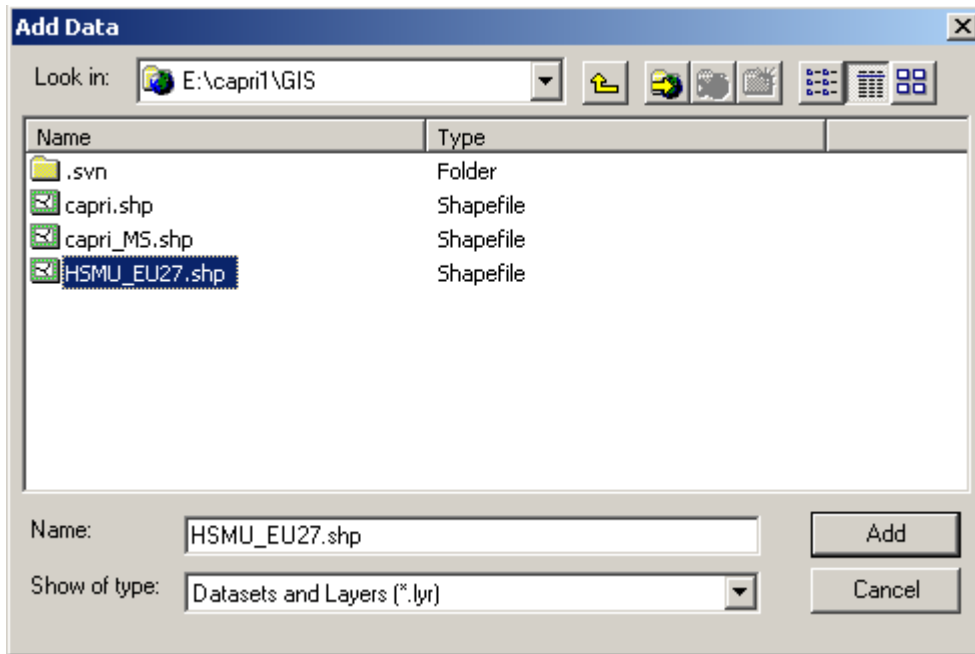
The last pane let's you decide for DBF-export if you want a list, or if you want of the data dimension spanned across the columns. For exporting the HSMU tables, it is recommended to put “Inputs and outputs” in the columns.

If everything has worked well, we should now find two files: one with the data, named as chosen in the file dialog, and a second one with “_meta” introduced before the file extension.

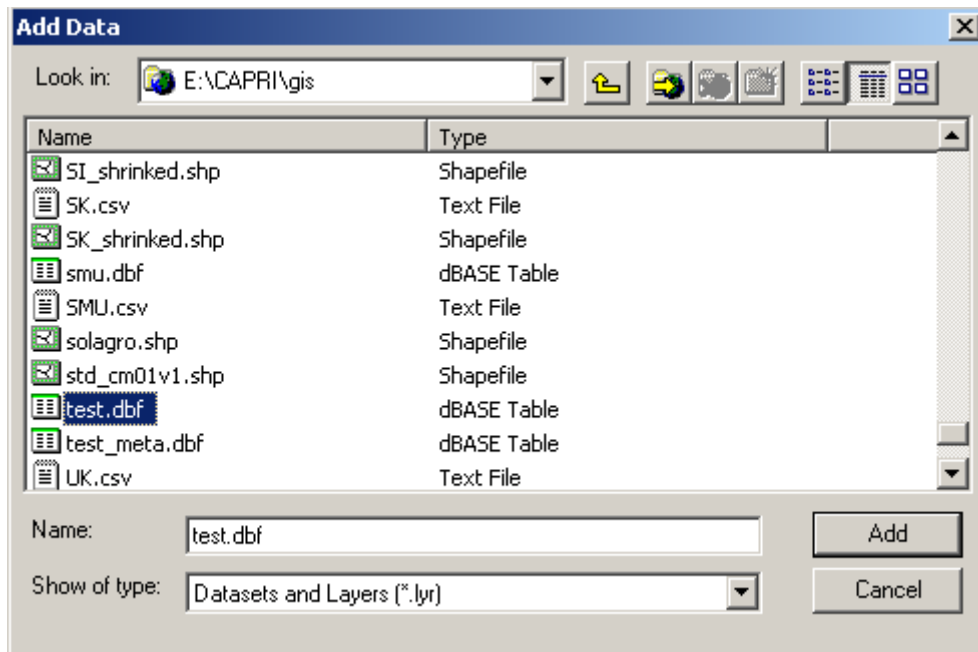
The following section will briefly explain how to the now work with the data in ArcGIS. Under Layers, choose add Data,



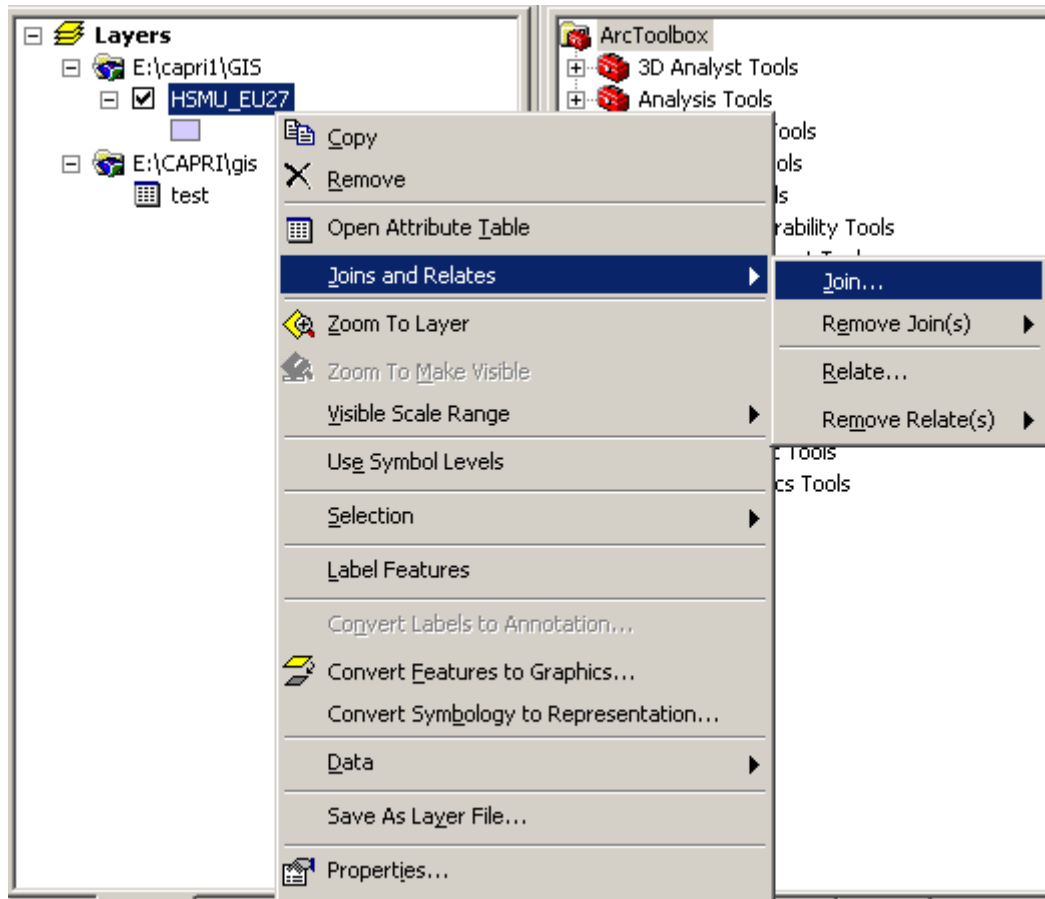
and in the case of the HSMUs, add the “HSMU_EU27.shp” shapefile.



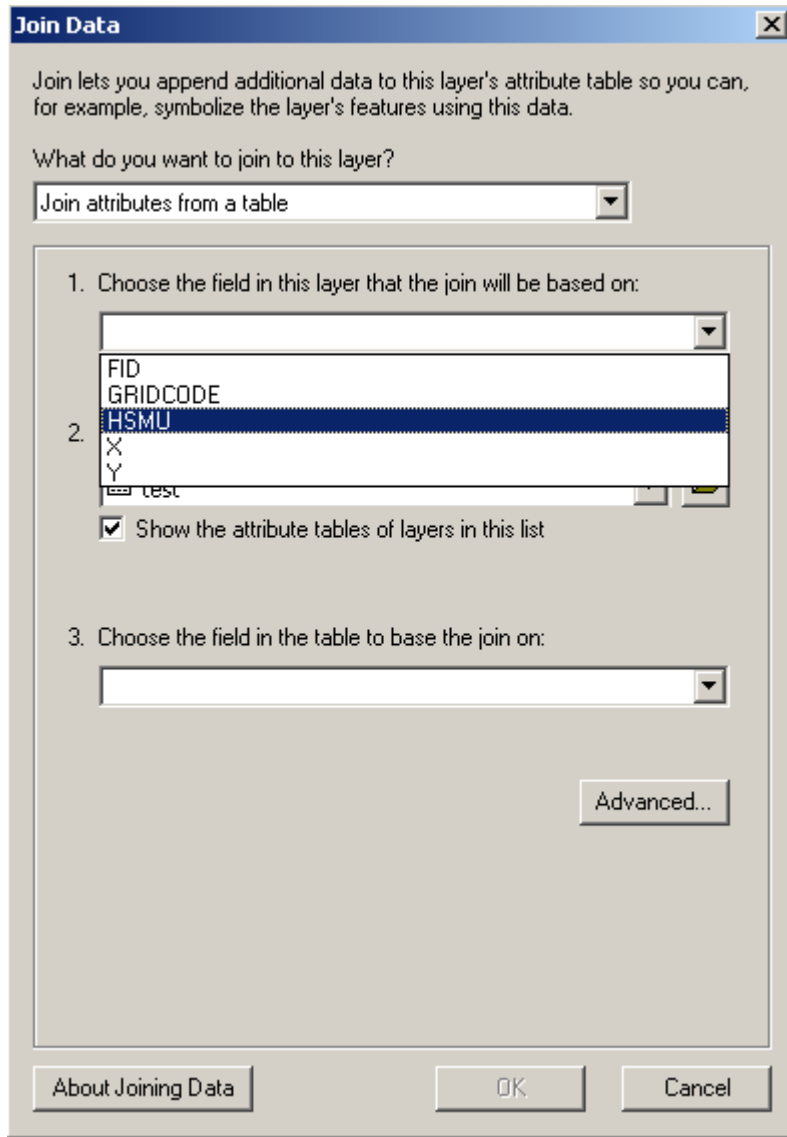
Then, choose add layers again, and add the dbf-file you generate in the step explained above. You may also add the file with the meta data.



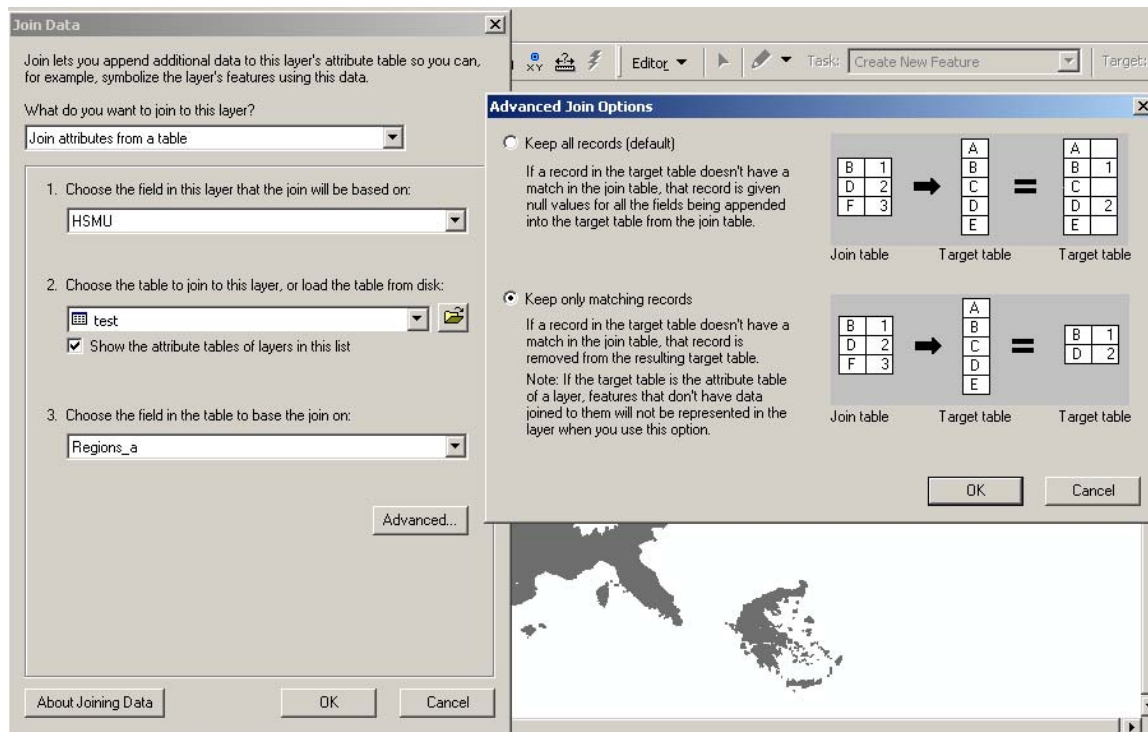
Next, we need to connect the HSMU geometry with the newly loaded data, a process called “joining” in ArcGIS. In the context of “HSMU_EU27”, choose “Join and Relates”, then “Join ...”.



That will open the join dialogue as shown below.



Make sure that “Join attributes from a table” is set in the first drop down box, and under 1., select HSMU, i.e. the field in the HSMU_27 geometry where the codes for the HSMU polygons are stored. Use the name of the exported dbf-table under 2., and select the field “Regions_a” (the field name are restricted to 10 chars) under 3. Then press the button labeled “advanced”, and choose the radiobutton “keep only matching records”. If you are asked to build index, confirm.



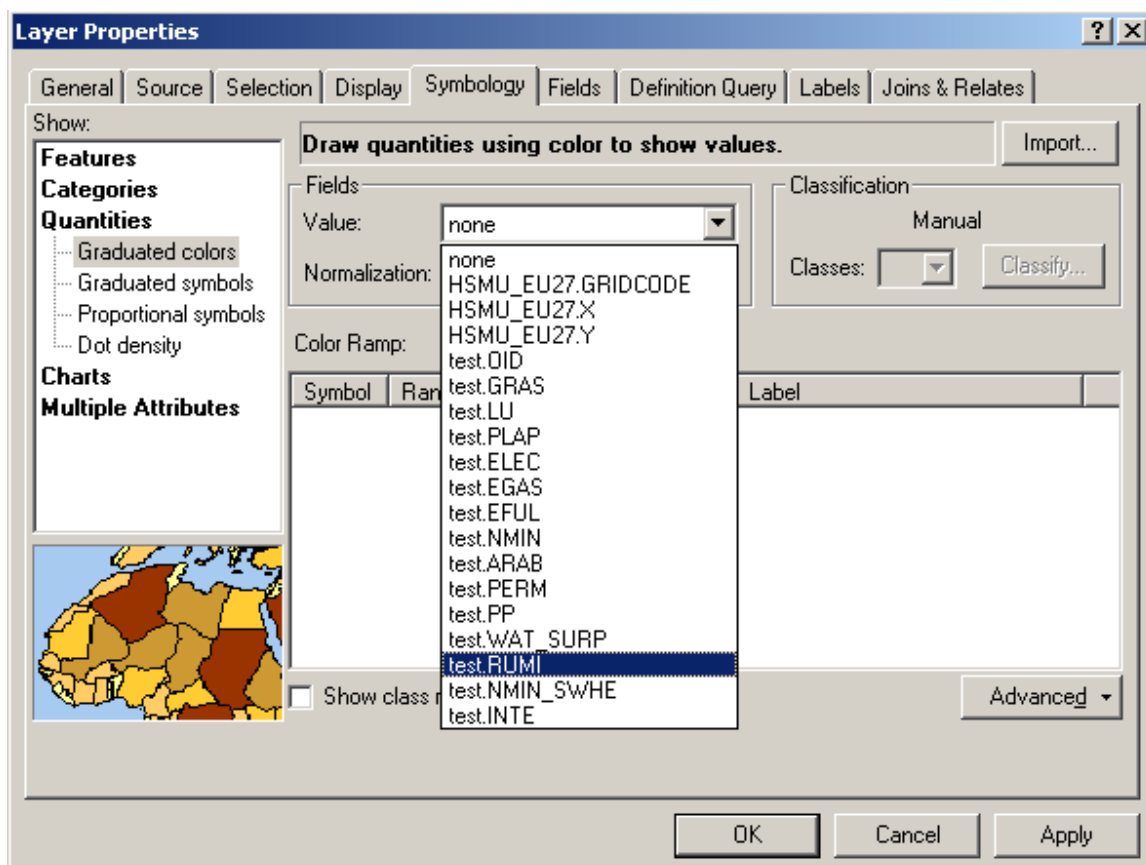
If anything has worked well, you should now see the country or countries you had in the original map.

There is a trap, though. If you export several tables, or results for several scenarios, your table will normally have several fields used as a row header (e.g. year, scenario, activity). If that is the case, the join will not work properly as several rows for the same regions will be joined to the very same polygon. Unfortunately, ArcGIS will not warn you about that. You have *first* to execute a definition query in the table, selecting those rows you are later going to draw a map from.

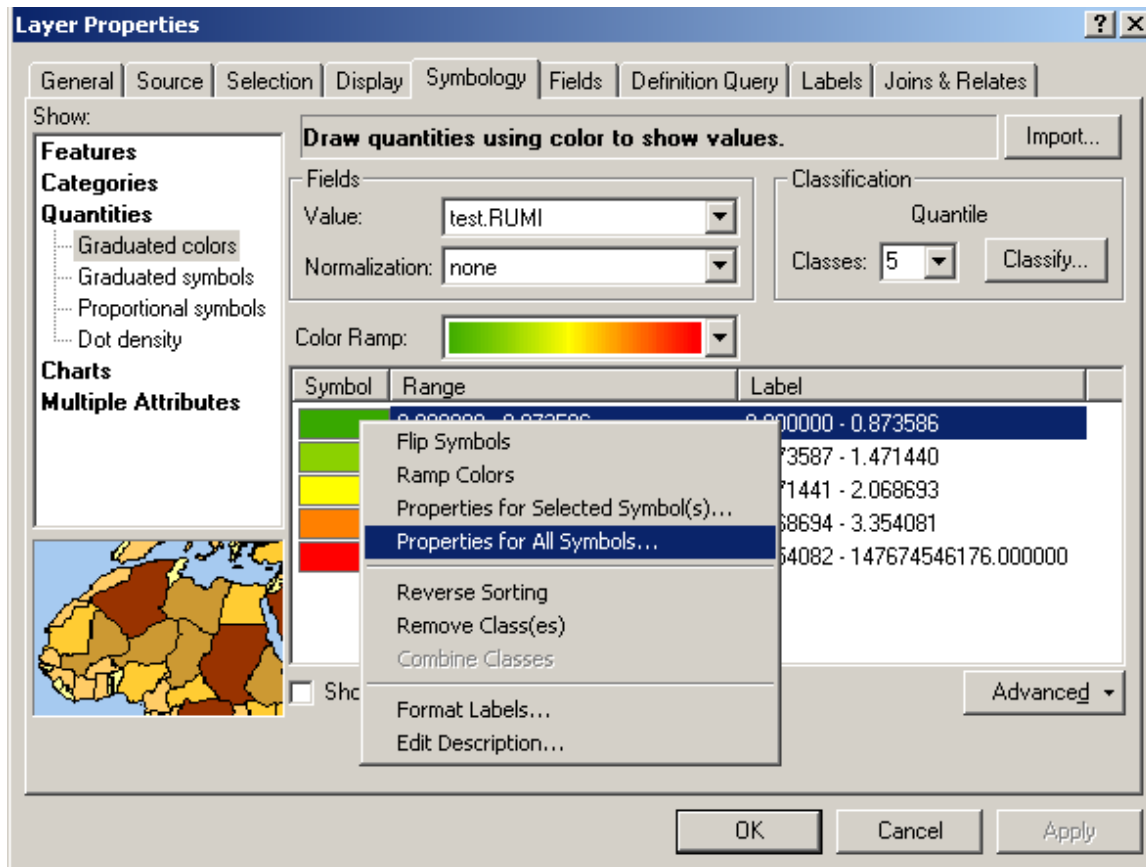
In order to draw now a thematic map, it may be helpful to add the file with the meta data to the map, and with the help of its context menu, open the meta-data table. It will give us the long description and units belonging to the data fields in the exported data table.

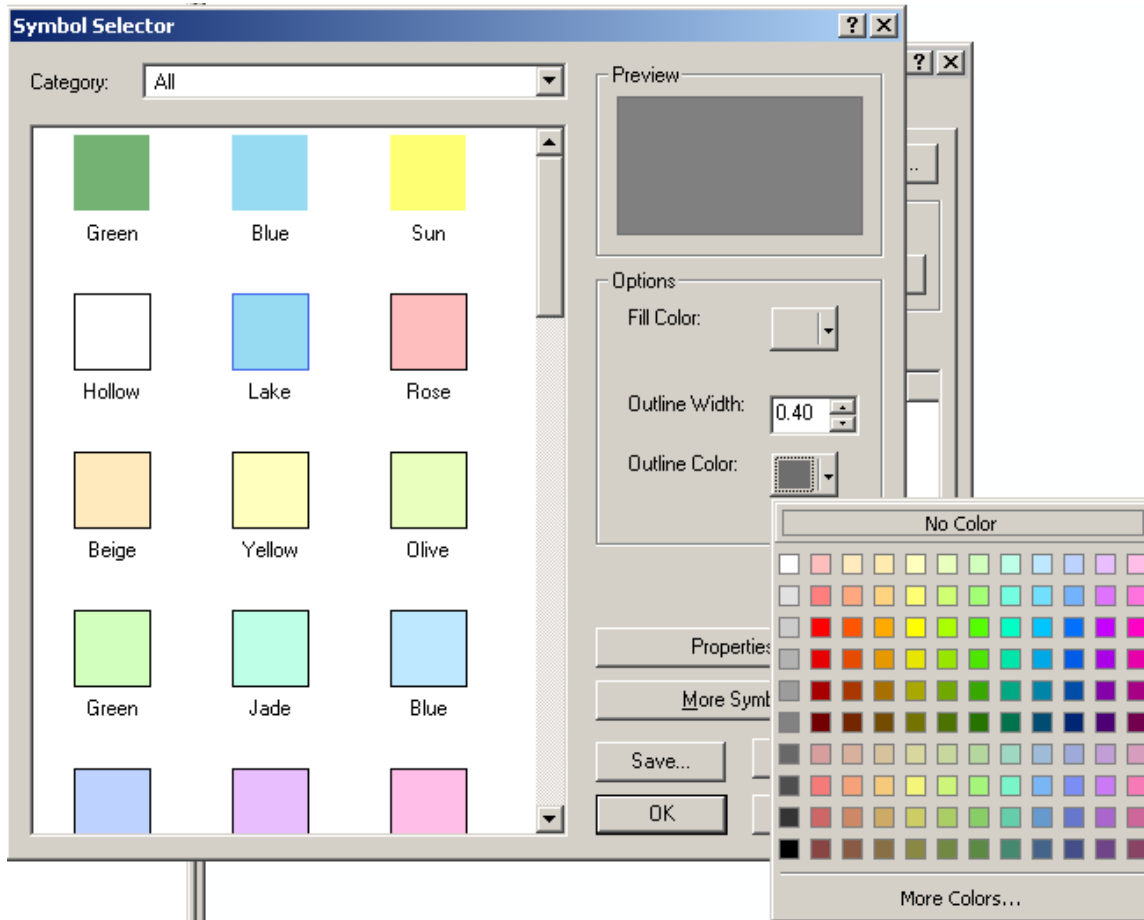
OID	Key	Name	Unit	LongText
0	NMIN	No 5 (a), Mineral Fertilizer Consumption, Nitrogen	kg N/ha	
1	PMIN	No 5 (b), Mineral Fertilizer Consumption, Phosphorous	kg N/ha	
2	NMIN_SWHE	No 5 (c), Mineral Nitrogen Application rate, Soft wheat	kg N/ha	
3	PLAP	No 6, Consumption of Pesticides	Euro /ha	
4	IRR	No 7(a), Irrigation, share	% irrigated	
5	WAT_SURP	No 7(b), Irrigation, abstraction	l/m2	
6	ELEC	No 8 (a), Energy, Electricity	Euro/ha	
7	EGAS	No 8 (b), Energy, Gas	Euro/ha	
8	EFUL	No 8 (c), Energy, Fuels	Euro/ha	
9	LU	No 10 (a), Cropping/Livestock pattern, livestock density	Livestock units / ha UAA	
10	RUMI	No 10 (b) Cropping/Livestock pattern, ruminants density	Livestock units / ha Fodder area	
11	PP	No 10 (c), Cropping/Livestock pattern, non-ruminants density	Livestock units / ha UAA	
12	ARAB	No 10 (d), Cropping/Livestock pattern, arable land density	%	
13	GRAS	No 10 (d), Cropping/Livestock pattern, grass land density	%	
14	PERM	No 10 (d), Cropping/Livestock pattern, permanent crops density(d)	%	
15	INTE	No 12 (a), low-medium-high input farming	Index 0 - 2	
16	H2865	BL21H2865		
17	H2866	BL21H2866		
18	H2867	BL21H2867		
19	H2868	BL21H2868		
20	H2869	BL21H2869		
21	H2870	BL21H2870		
22	H2871	BL21H2871		

Assuming we want to draw now a map with the ruminant stocking density, we find it in row 10 under the key “RUMI”. In order to produce now a map, we have to open the context menu of “HSMU_EU27”, and choose properties, symbology, and choose “Quantities”. Under values, choose “RUMI”, the name before is the name of the DBF-file.

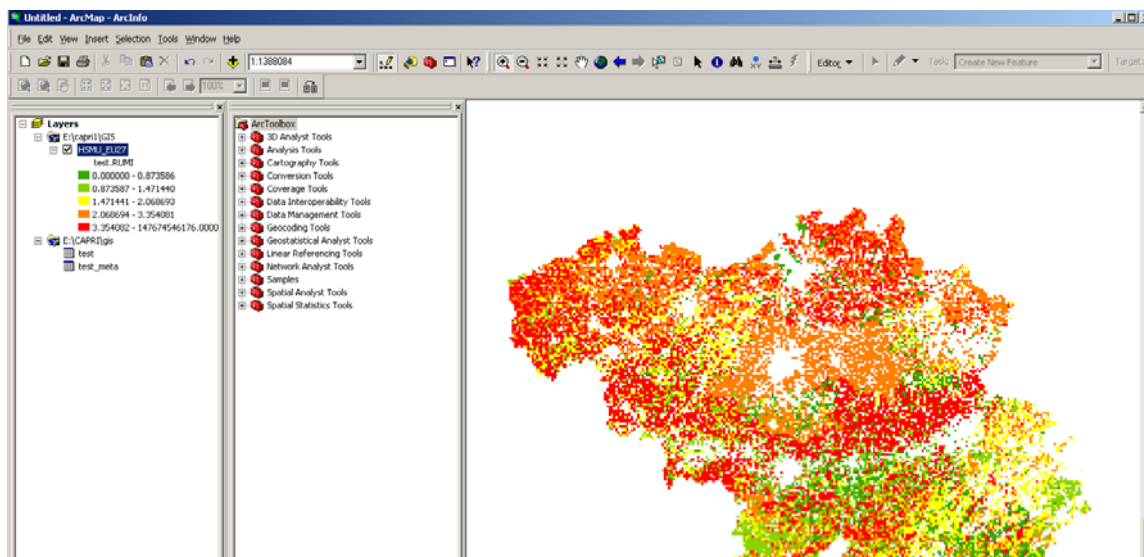


Afterwards, under classification, choose your preferred one. As there many small polygons, the outline of the polygons should not be drawn, click on one of the colors, choose “Properties for all symbols ...” and under “Outline color” chose “No Color”.





Afterwards, if anything went well, you should see your map.



What are the HSMUs and what information is available?

The HSMU are the so-called Homogenous Soil Mapping Units. Each HSMU contains one or several 1x1 km grid cells, not necessarily adjacent, and are defined so that these are more or less homogenous regarding climate, soil, slope, CORINE land cover class and NUTS II region. There are about 110.000 HSMUs for EU15. The spatial downscaling introduced in CAPRI-Dynaspat provides the following information per HSMU:

- Cropping shares and animal stocking densities
- Yields
- Economic indicators per crop and animal, and in relation to UAA
- Fertilizer application rates
- Environmental indicators

How to visualize the HSMU information

Given the 1x1 grid resolution, the most obvious way to look at the information is to produce maps with the CAPRI GUI. There is a co-ordinate set available which is called “HSMU.zip” which comprises the geometry for about 1.8 Mio Polygon which represent the HSMUs.

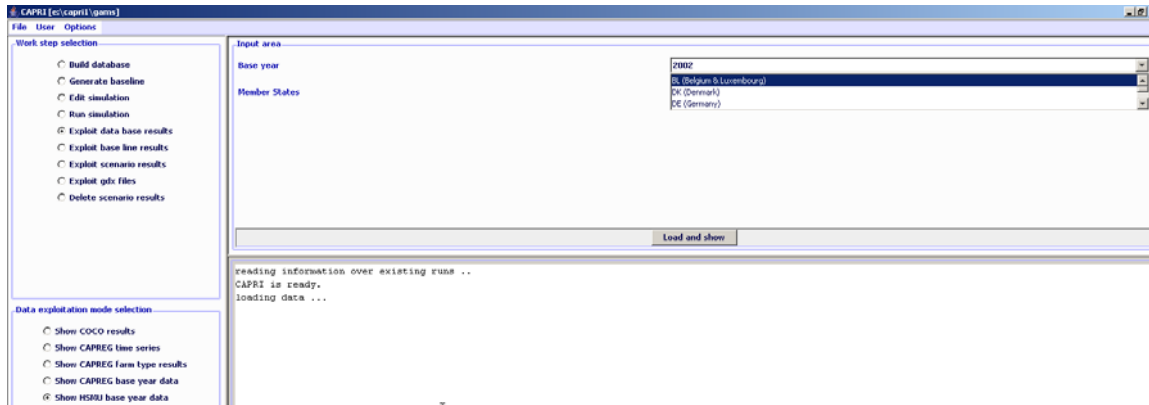
There are four options to view HSMU data:

1. Loading data for one or several Member States for the base year (dis-aggregated information from the NUTS II CAPRI data base).
2. Loading data for one or several Member States for the base year and the baseline, the latter representing dis-aggregated data from NUTS II results of the baseline calibration.
3. Loading data for one or more scenarios for a given year.
4. Loading data manually

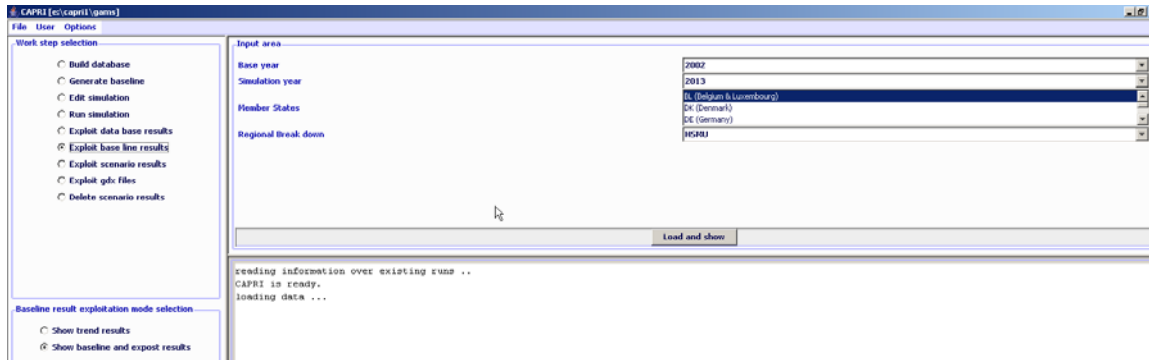
Loading data based on the GUI presets

As for the results at NUTS II level, there are three pre-defined exploitation possibilities included in the CAPRI GUI:

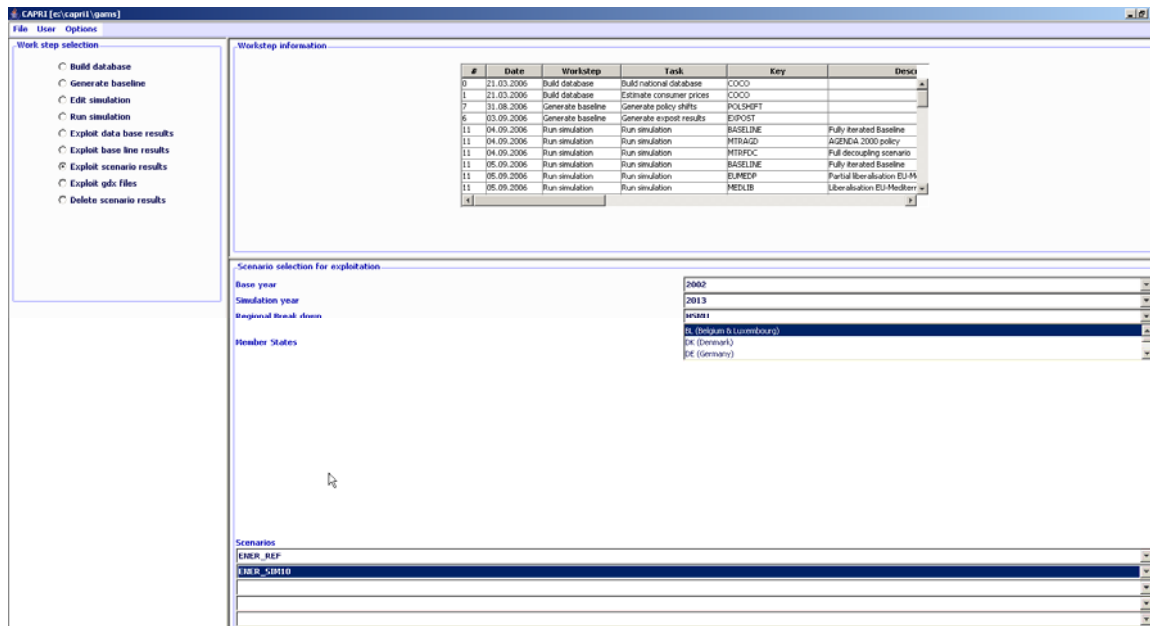
1. Viewing the results for the base year. Given the tremendous number of HSMUs, the user can select for which Member States to load the information.



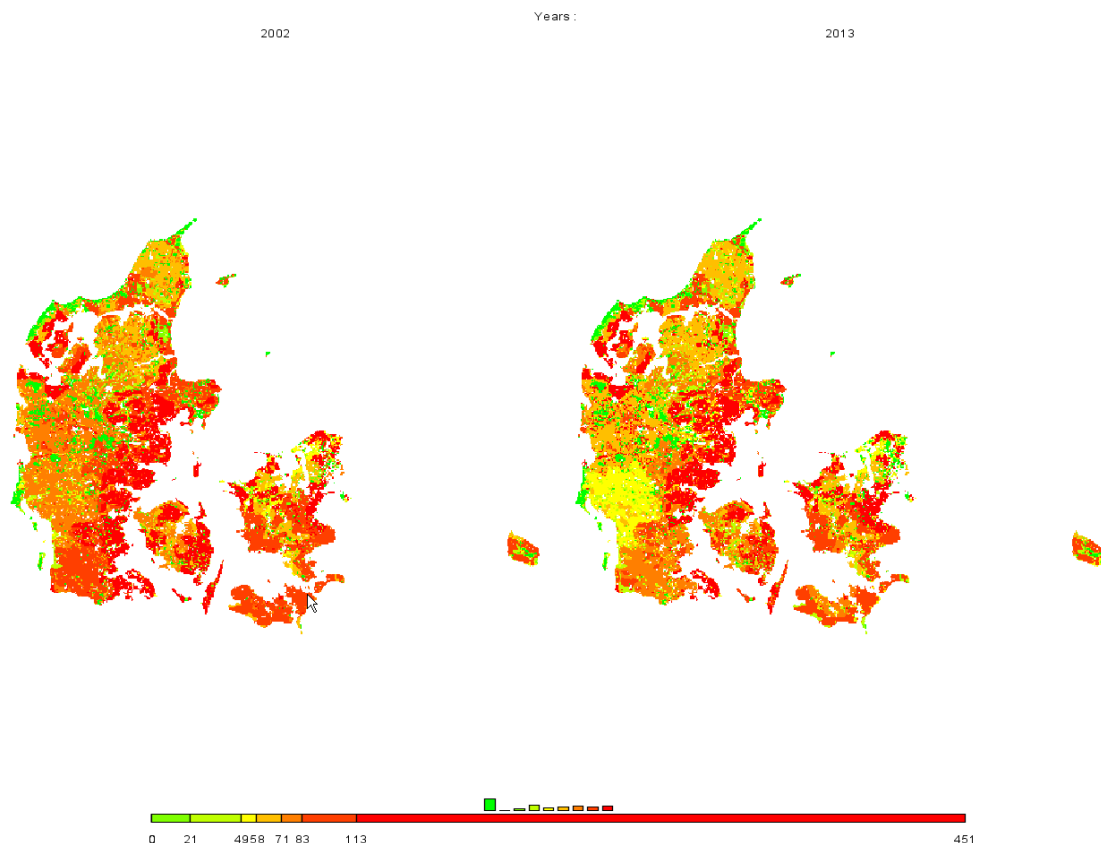
2. Comparing the results for the base with the baseline (projection results), comparison between two points in time. Again, the user can select the Member States.



3. Comparing results from different scenarios (comparisons for one point in time, but changes in drivers/assumptions relevant for the CAPRI economic model).



Once the loading is accomplished, the right hand side of the GUI is filled with a tabular view of the results, which can be turned easily into maps (other forms of exploitation as graphs are less suited given the large number of observations). Details how to work with the exploitation tools are found in a separate document. The screen-shot below shows results for Denmark as an example.



In order to ease exploitation of the results, pre-defined tables are set up, currently broken down into five categories:

1. Agri-environmental indicators, driving forces (mineral fertilizer consumption, consumption of pesticides, irrigation shares, energy consumption, livestock densities, shares or arable / grass land or permanent crops)
2. Agri-environmental indicators, pressures and benefits (Gross nitrogen and phosphorous balance, green house gas emissions, High Nature Value Farm land indicator)
3. Economic indicators at HSMU level (market revenues, variable production costs, income)
4. Climate, soil, slope and altitude
5. Results from the DNDC meat model (gas losses for different nitrogen compartments, mineralization, leaching)

The tables on agri-environmental indicators (dirving forces, pressues and benefits) are set-up as close as possible according to the official EU Guidelines for Rural Development indicators.

Activity	2002	2013
H22500		
H22508		
H22510		
H22511		
H22512		
H22513		
H22514		
H22515		
H22516		
H22517		
H22518		
H22602	51.77	51.84
H22603	104.82	118.87
H22604	102.53	111.82
H22605	64.34	68.89
H22606	65.15	67.51
H22607	128.25	135.80
H22608	132.88	145.71
H22609	78.41	71.88
H22610	69.56	70.93
H22611	132.88	138.82
H22612	123.17	129.35
H22613	56.99	53.48
H22614	55.87	52.14
H22615	52.52	49.16
H22616	63.58	67.83
H22617	55.88	51.88
H22618	48.33	62.82
H22619	38.84	36.16
H22620		
H22621		
H22622		
H22623		
H22624	51.87	38.92
H22625	69.95	72.25
H22626	126.26	131.89
H22627	62.88	62.87
H22628	66.58	65.29

Examples

Drawing a map showing the nitrate surplus for EU27 at regional level in the base year

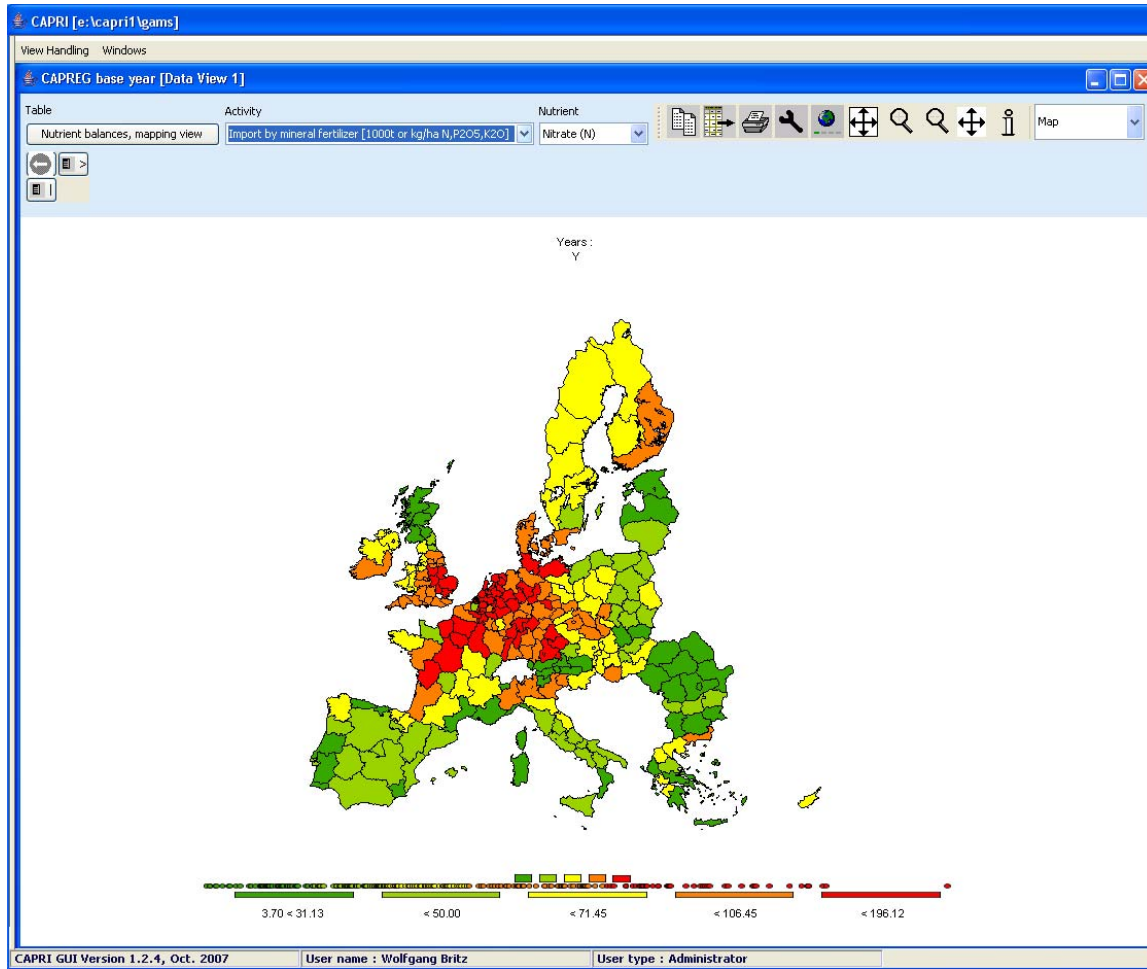
Firstly, we need to select “Exploit data base results” in the work step selection panel, and then choose the radio button “Show CAPREG base year data”. Then, in the Member States drop-down list, right click the mouse, and select “EU27”. The “Load and show button will then load the results.

	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Crop share/Animal density [% or 0.01 animals heads/ha]	Production per UAAR [kg/ha]
Cereals	308.22	1484.26	53.27	3317.92
Oilseeds	373.27	63.49	2.28	67.13
Other arable crops	1086.43	140.57	5.05	1977.45

Next, select a different selection of tables by pressing on the button below “Table”, which currently shows the topic “Supply details”. In the drop-down list, go-to “Environment” and select “Nutrient balances, mapping view”.

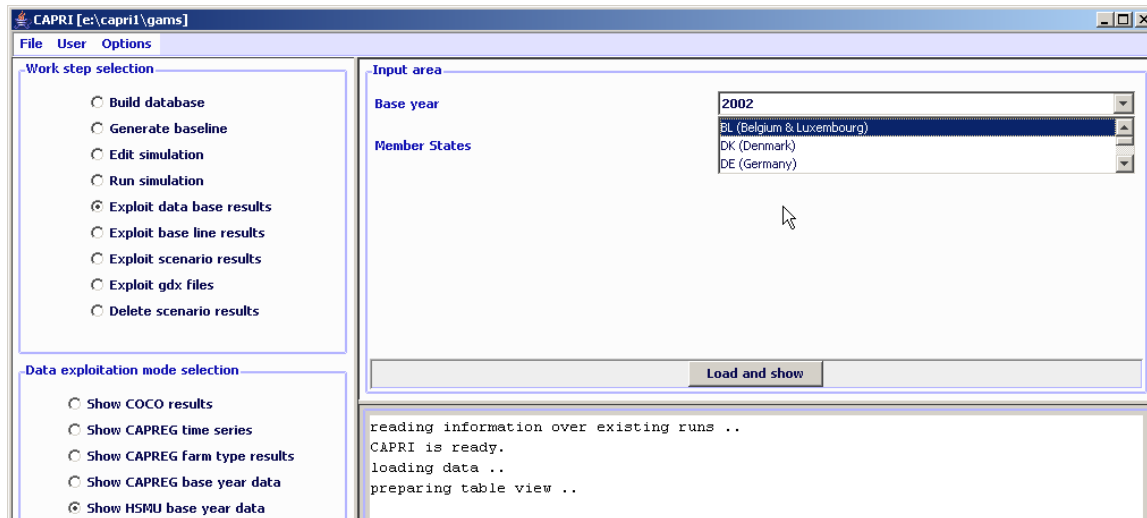
	Income [Euro/ha or head]	Hectares or herd size [1000 ha or hds]	Crop share/Animal density [% or 0.01 animals heads/ha]	Production per UAAR [kg/ha]
Cereals	308.22	1484.26	53.27	3317.92
Oilseeds	373.27	63.49	2.28	67.13
Other arable crops	1086.43	140.57	5.05	1977.45

The following map should appear. You may select now different elements of the balance, by using the drop-down boxes under “activity” or change the nutrient, by using the drop-down box under “nutrient”.

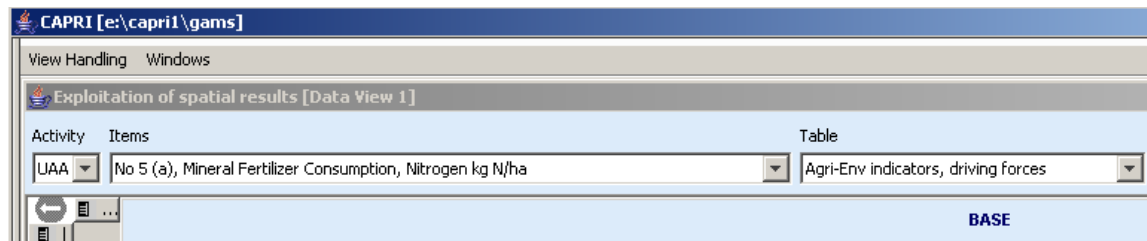


Drawing a map of the High Nature Value Farmland indicator for Belgium for the base year

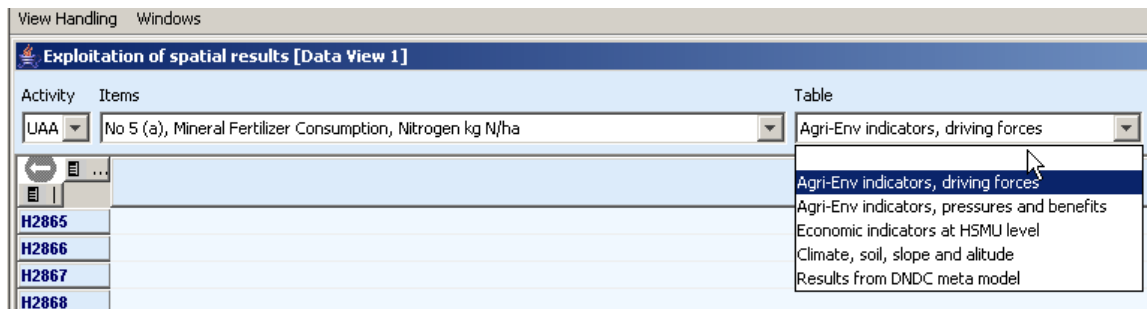
Firstly, we need to select “Exploit data base results” in the work step selection panel, and then choose the radio button “Show HSMU base year data”. Then, in the Member States drop-down list, Belgium must be selected, as shown below.



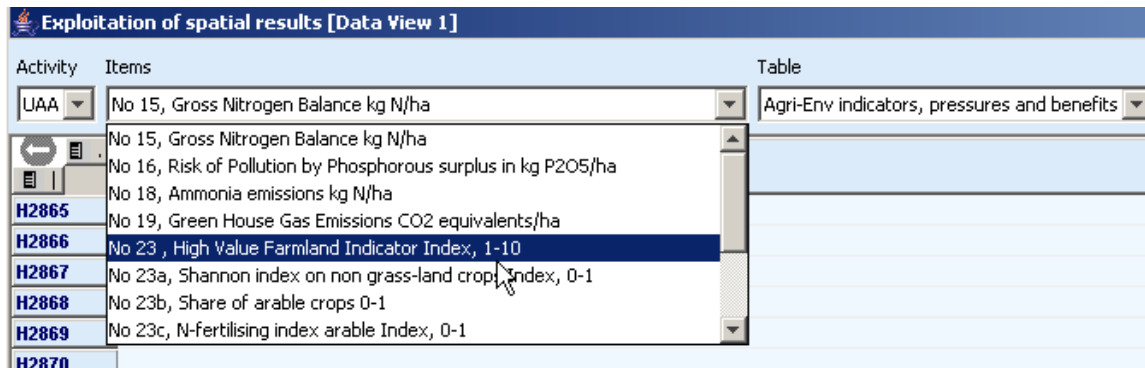
Pressing the load and show button will then bring up the first table links to the HSMU results as shown below.



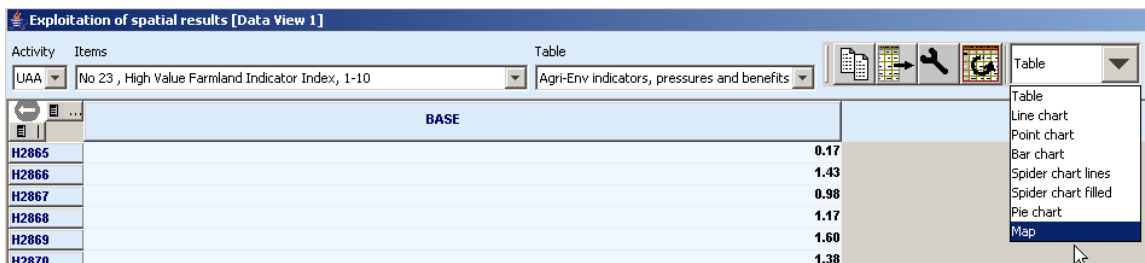
Now, according to the guidelines for RD indicators, HNV is to be found under “Pressures and benefits”, which is stored under another table. So opening the drop down list for the tables allows us to select the correct table.




Next, we need to find the correct item, to do so, the drop-down lists for the items must be selected, and the indicator 23 selected, as seen below.



Those numbers should now be shown as a map, to do so, select “Map” from the drop-down list where “Table” is shown.



The “hour glass” cursor is shown and the geometry will be loaded which may take a few seconds. Afterwards the standard map comes up (green-yellow-red color model, quantile classification, polygons shrunk, no area weights, zeros included in classification). Now, for the HNV indicator ranging from 0-10 where 10 are the favorite numbers, and real zeros indicate missing values, the following settings could be appropriate:

- Equal interval classification with 10 classes
- Zeros treated as missing values
- And, using area weights may be appropriate so that the frequency graph below the maps shows the share of UAA in each of the ten classes
- A linear scale works nicer for our example, and if we are at it, as the data only range to 9.88, we should round the number to integer (use the “tools” button , and choose “0” in the “Fraction digits” scroll-down box.

Map option dialogue

Color table: Start color Mid color End color Green yellow red
 Set value for middle color 2,96
 Treat zeros as missing values, exclude from classification and don't draw
 Use area weights for classification

Classification method: Equal interval
 Number of classes: 10
 Number of regions with small values to remove from class definition: 0
 Number of regions with large values to remove from class definition: 0

Legend: Continuous bar, linear scale

#	label	class limit	% of obs	color
1	0.00 < 0.99	0.998	28.034	Green
2	< 1.98	1.975	9.972	Light Green
3	< 2.96	2.963	18.063	Yellow-Green
4	< 3.95	3.951	7.123	Yellow
5	< 4.94	4.938	9.744	Light Yellow
6	< 5.93	5.926	6.781	Orange
7	< 6.91	6.914	7.066	Light Orange
8	< 7.90	7.901	4.9	Red-Orange
9	< 8.89	8.889	5.812	Red
10	< 9.88	9.877	2.507	Dark Red

Cummulative distribution graph Frequency groups: 100 Draw mean and +/- 1 std. dev.

Show small circles, showing distribution of regions n 1755.0
 Show rectangle representing distribution of classes Min 0.0
 Draw in high quality Mean 3.227795
 Shrink polygons according to share of UAA Median 2.6468632
 Std.Dev 9.876518
 Std.Dev 2.772104

Draw outline in same color

Dimension shown in columns of result window for current region: Scenario
 Dimension shown in rows of result window for current region: Activity
 Title on top of map: Standard title

ok

Customize view

Font: Arial 11 pt

Fraction digits and decimal separator: 2

Column width: 62
 Row width: 62

Separator between merged data dms: 6



Use default pivoting for tables
 Hide empty rows
 Hide empty columns

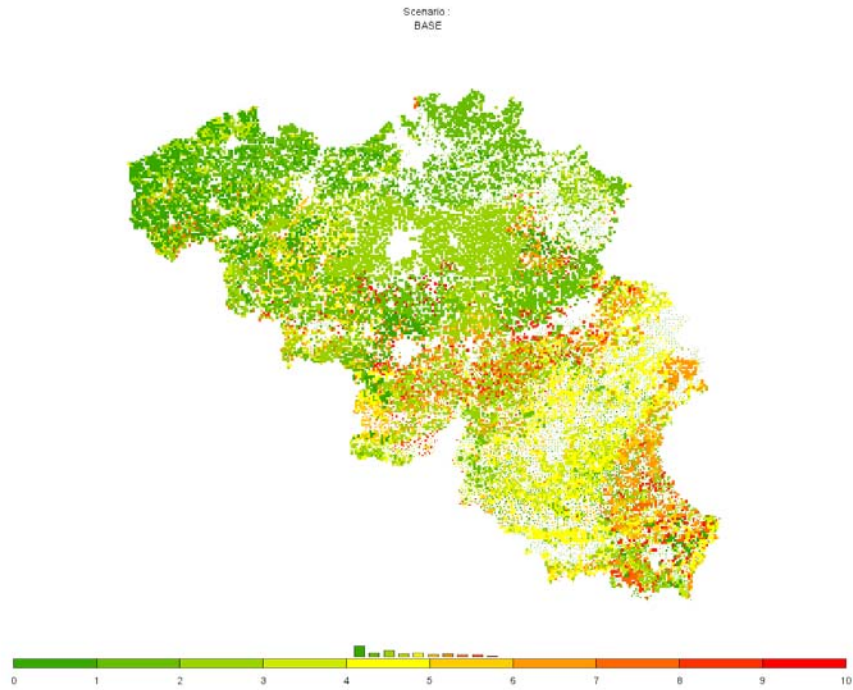
Show only selected items

Long texts only

Comparison output: Only values
 Data dimension used for comparisons: HMSUs
 Element used for comparisons: BE21HC965

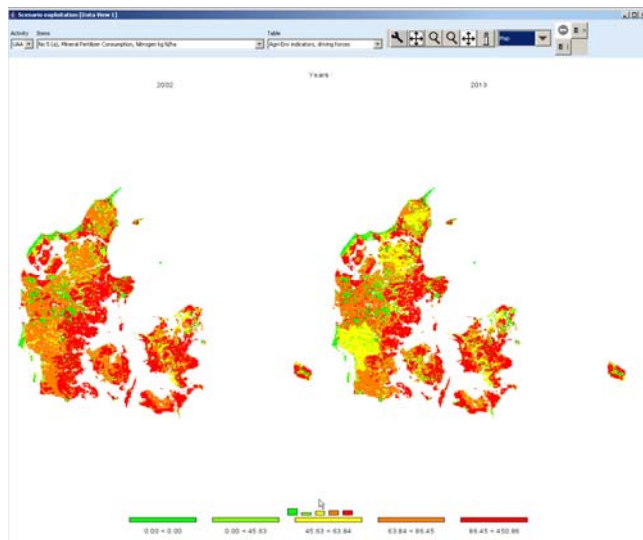
ok

That should give the following map which then can be exported to other application via the clipboard  or can be send to the printer .




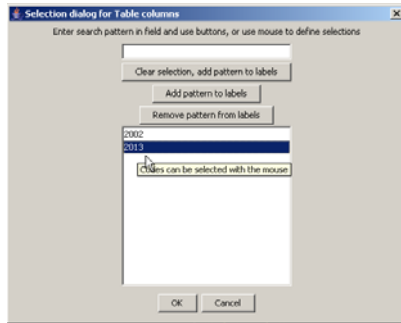
Drawing a map showing changes between the base year and the ex-ante results from the baseline


When scenarios or different points are compared with each other, it is often useful to draw maps which show relative or absolute changes. The following map is the typical starting point when the baseline is analyzed: two maps with identical class definitions, one for the base and one for projection year.

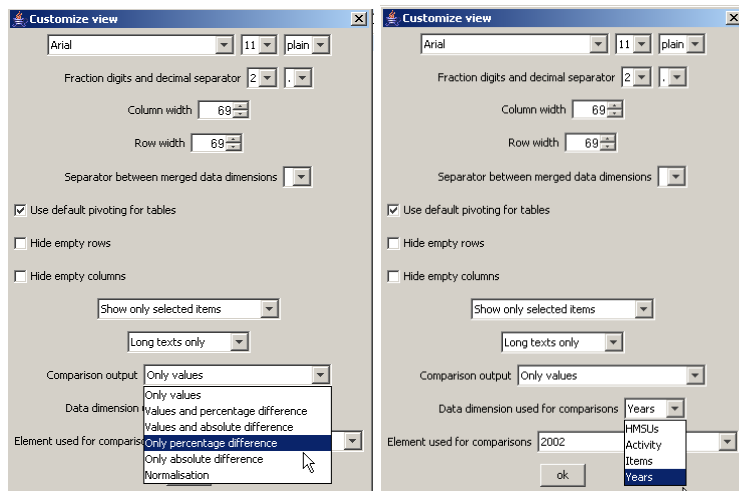


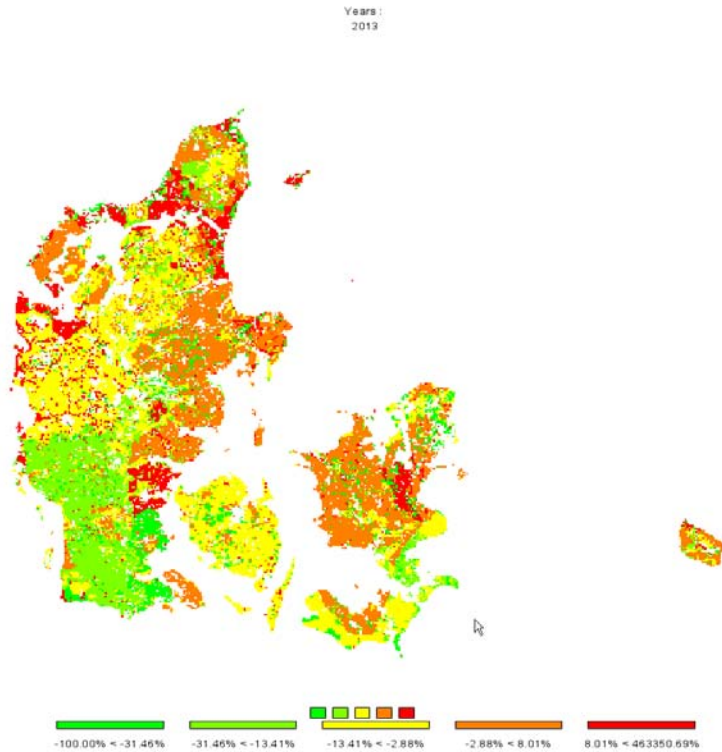
In order to draw a map with changes, we must first get rid of the base year by de-selected the first map.

This is done by using the “column selection” button  which is found in upper right corner of the window. When the button is pressed, a dialog opens, and one can select with the mouse the projection year, only. Afterwards, the left map will not longer be present.

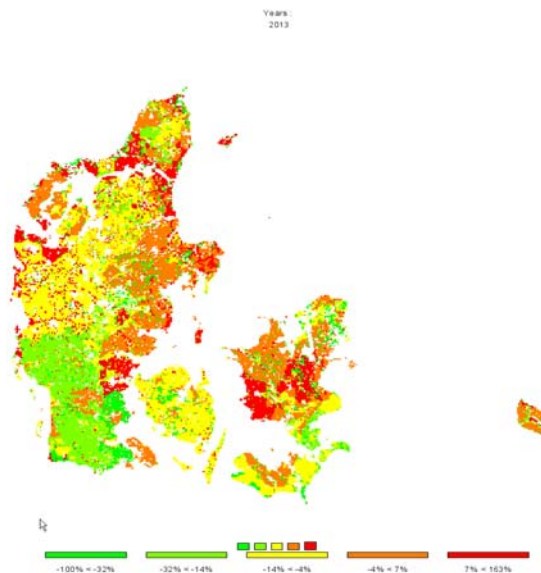
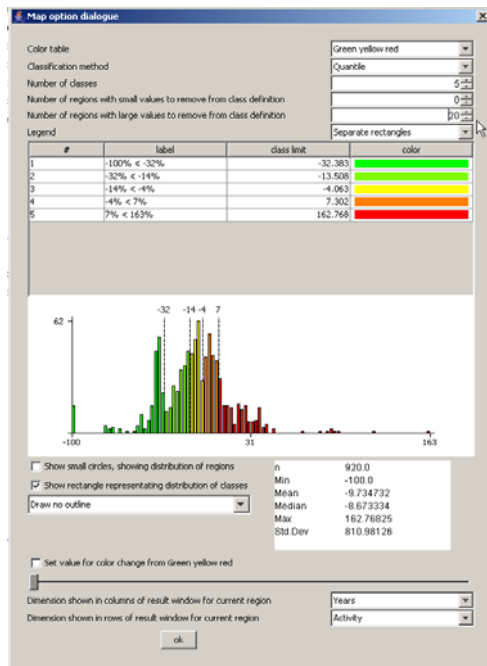


Next, we need to change what is shown in the map to relative changes to the base year. That can be accomplished by using the tool dialogue (press  button). In the tool dialogue, select “only percentage differences” in the drop-down box labeled “comparison output”, and then put the “data dimension used for comparisons” to “Years”. The “Element used for comparisons” should be “2002”. After pressing o.k., the map will change as shown.



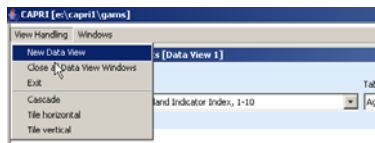


Now, there are two things we would most probably like to change: the number of digits shown in the legend, and getting rid of very large values shown in the legend. The number of digits can be changed again with the “tool” dialogue by changing the fraction digits. The large number can be excluded from the classification by increasing the “Number of regions with larger numbers to exclude from class definition”, in the example below the number had been set to 20.



Drawing a map with the base year results next to one showing changes from the base year to the baseline results

There are two ways to draw different maps. If more than one column is selected in the underlying table view, several maps with identical coloring and scaling will be drawn as shown in the sample above. That is not advisable – in our example we need to different scales, one appropriate for the absolute values and one appropriate for changes. In order to do so choose “View Handling”, “New Data View” and then “Tile vertical”.

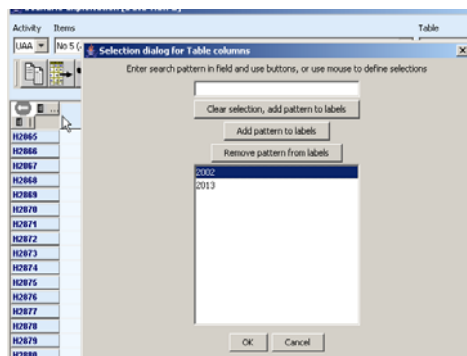


We will then see something as shown below.

The screenshot shows two side-by-side data views of a table. Both views display the same data for the years 2002 and 2013. The table has columns for '2002' and '2013'. The rows are labeled with activity codes from H2005 to H2009. The data values are as follows:

Activity	2002	2013
H2005	154.89	178.40
H2006	165.89	209.40
H2007	122.77	223.82
H2008	119.37	207.01
H2009	95.68	116.22
H2010	90.81	113.86
H2011	86.37	167.78
H2012	77.68	128.46
H2013	155.88	182.24
H2014	75.71	167.21
H2015	28.18	34.82
H2016	28.90	68.61
H2017	29.99	106.15
H2018	30.67	66.87
H2019	47.64	59.47
H2020	46.99	46.99
H2021	53.06	281.88
H2022	53.06	281.88
H2023	53.06	281.88

Now, in the left hand side, only the results for the base year should be shown. That can be accomplished by deselecting the column for “2013” – use the column selection button to do so.



The very same trick should be applied for the other view, only deselecting the results for “2002”. Now we see something as below:

2002		2013	
H2865	151.99	H2865	178.40
H2866	195.89	H2866	289.48
H2867	122.77	H2867	223.82
H2868	118.37	H2868	287.81
H2869	95.88	H2869	116.22
H2870	98.81	H2870	113.86
H2871	86.37	H2871	167.78
H2872	77.68	H2872	126.46

Now, for the map with the results for 2013, we should switch to percentage different to the “2002” by opening the “tool” dialog . There, under “Comparison output” choose ”Only percentage difference”, Select “Years” under “Data dimension used for comparisons” and ensure that “2002” is selected in “Element used for comparisons”.

Customize view

Font: Arial, Size: 11, Style: plain

Fraction digits and decimal separator: 2, .

Column width: 67

Row width: 62

Separator between merged data dimensions: [dropdown]

Use default pivoting for tables

Hide empty rows

Hide empty columns

Show only selected items: [dropdown]

Long texts only: [dropdown]

Comparison output: Only percentage difference

Data dimension used for comparisons: Years

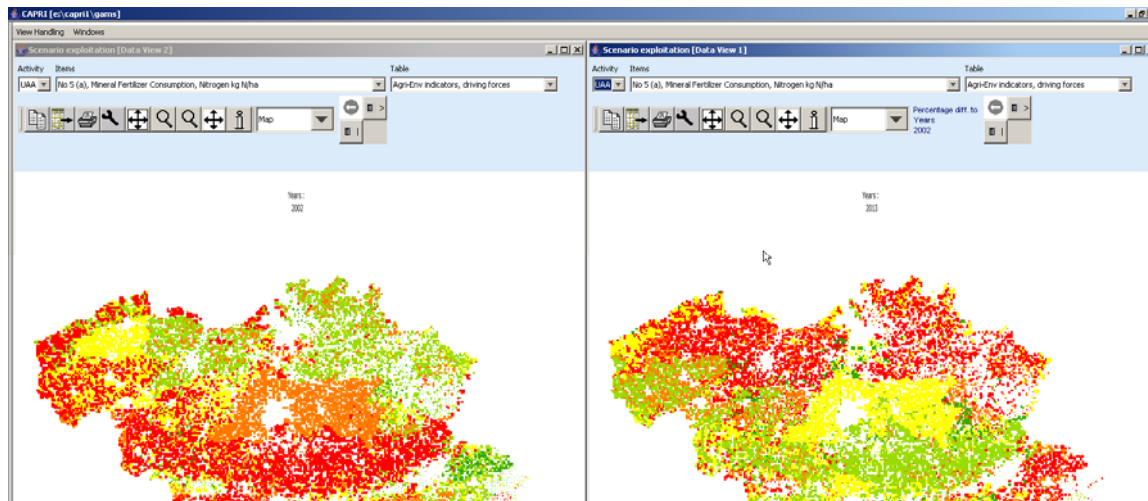
Element used for comparisons: 2002

OK

Now we should get a result as below.

2002		2013	
H2865	151.99	H2865	12.11%
H2866	195.89	H2866	47.83%
H2867	122.77	H2867	82.39%
H2868	118.37	H2868	73.48%
H2869	95.88	H2869	23.66%
H2870	98.81	H2870	25.39%
H2871	86.37	H2871	94.25%
H2872	77.68	H2872	65.59%
H2873	155.88	H2873	16.99%
H2874	75.71	H2874	126.99%

Now, for both views, the output should be switched to maps, and there we are ...



Index

Area weighted classification 57

CAPRI installation 17

Classification

Equal interval 58

Manual classification 59

Mean standard dev 59

Nested mean 59

Optical control 55

Quantile 58

Classification method 57

clipboard 66

Color table 60

Column and row selection 40

drag 67

drawing several maps 98

Excluding zeros from classification 57

file export 66

Frequency diagram in map 60

full extent 67

Gams settings 18

Getting data for specific polygons 67

Graphics

Bar charts 49

Line and point charts 50

Pie charts 51

Spiderweb plots 52

Highlighting specific regions in the map 68

info pointer 67

legend

continuous linear scaling 64

continuous logarithmic scaling bar 65

Login in 16

map with percentage changes 95

Natural breaks 58

numeric filter 69

Online help 18

Shrinking polygons according to UAA share 56

Tables

Filtering 46

Sorting 45

title of the map 66

Updating the map 72

View

scenario result 87

View data

Base year 87

baseline 87

zoom in 66

zoom out 67