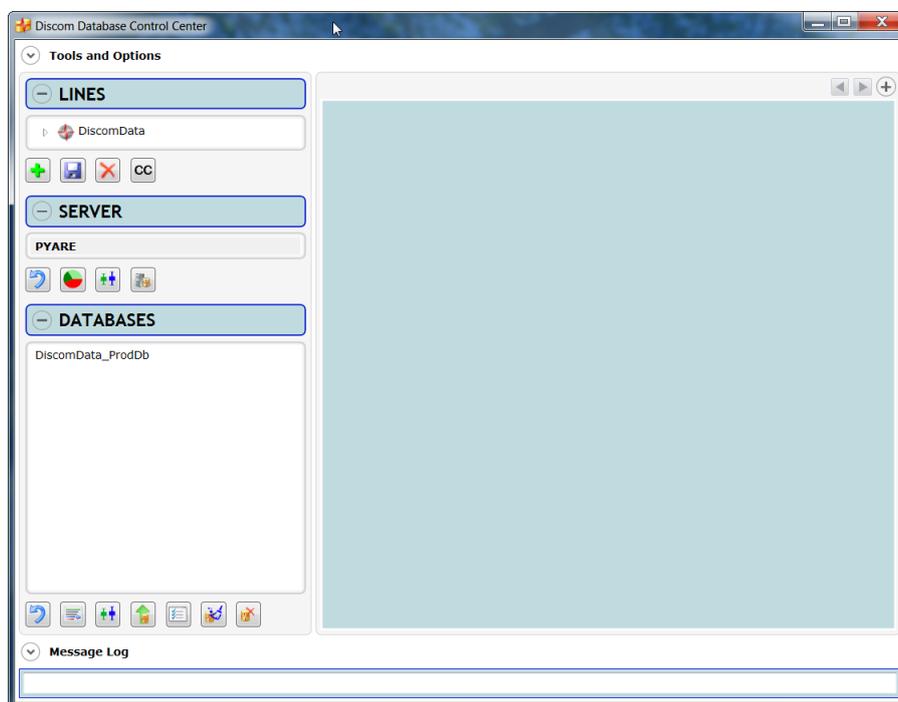


# DbControl

## User Manual

17.06.2016



Build Version 2.1.6008.15216

# I. Contents

I.	Contents.....	0
1.	Introduction .....	1
1.1	File structure .....	1
1.2	Input files .....	2
1.2.1	FactoryData.xml .....	2
1.2.2	DbControlSettings.xml .....	5
1.2.3	ProdDbConf.xml/ShrinkDbConf.xml .....	6
2.	Installation .....	1
2.1	Network (server) settings .....	1
2.2	Copy instructions (stand-alone server operation) .....	1
2.3	Copy instructions (network-based operation).....	1
3.	Graphical User Interface (GUI).....	3
3.1	Views and navigation .....	3
3.2	Lines .....	5
3.3	Servers.....	7
3.3.1	Single server statistics .....	8
3.3.2	Selected server statistics.....	8
3.4	Databases.....	10
3.4.1	Load/reload database .....	10
3.4.2	View database tables .....	11
3.4.3	View database statistics.....	11
3.4.4	Create/update database .....	14
3.4.5	Cleanup database.....	18
3.5	Task Scheduler .....	24
3.6	Additional tools and options.....	28
A.	Supplement: Structure and functions of the “CleanupDb” database tables and procedures.....	30
a.	DeleteLog .....	30
b.	DeleteParameters .....	31
c.	TimeLimitsOfTypes.....	32
d.	Delete process.....	32
B.	Supplement: Troubleshooting .....	33

---

C. Supplement: Application source code info.....	33
--	----

## 1. Introduction

The Discom Database Control Center (DbControl) is a WPF-based GUI for monitoring and controlling Discom production databases and production database related tools. These tools include:

- CreateDb (creation and update of a new default database)
- UpdateDb (update of existing databases)
- CleanupDb (deletion from existing databases; manually or on a regular schedule)
- ManageFiles (copy, move, or deletion of DISCOM files; on a regular schedule)

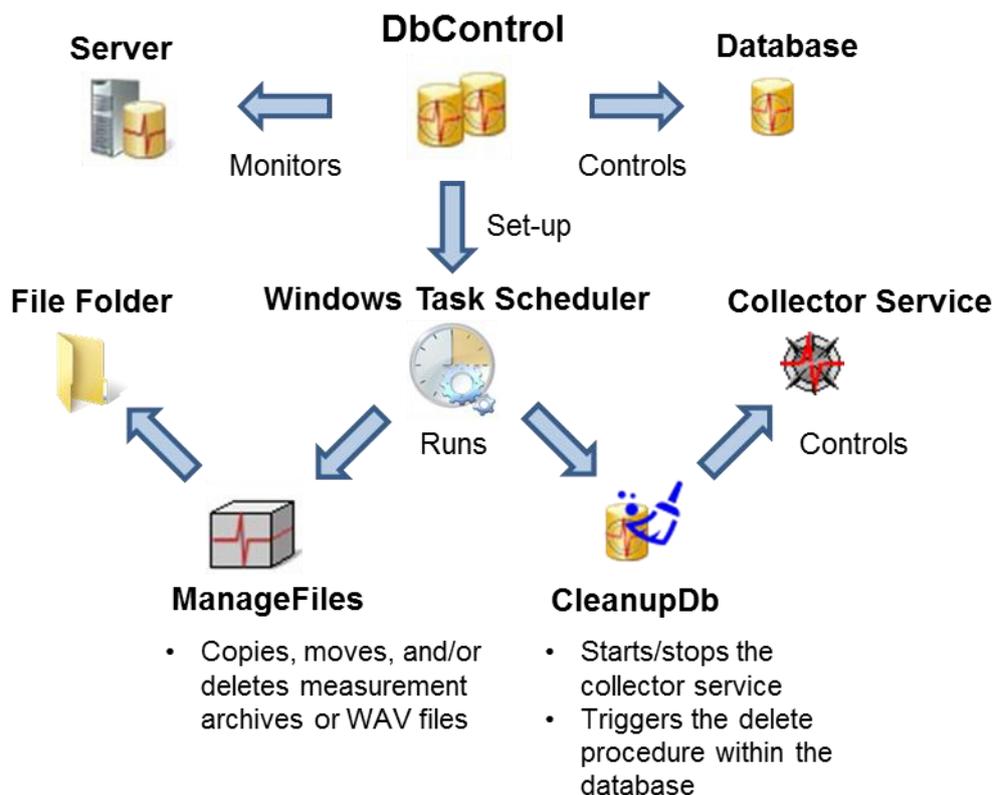


Fig. 1: Schematic program layout. DbControl allows the monitoring of the server disk space usage as well as the monitoring and controlling of DISCOM production databases. The database cleanup and the management of DISCOM measurement files are carried out using the Windows Task Scheduler.

DbControl is designed to operate on a server network, i.e. to monitor and control multiple Discom production databases (and its tools) on multiple servers of a network via a central server. DbControl may also be used on a stand-alone server.

### 1.1 File structure

DbControl consists of a number of program files for the above mentioned tools and a number of configuration files (input XML files and SQL scripts) required for running these tools.

Discom-Installation\Program Files\Discom\DbControl

- **DbControl**: Executables and libraries for the DbControl GUI

- **ManageFiles:** Program for copying, moving, and/or deleting Discom files (measurement archives, WAV files).
- **CleanupDb:** Program for running the SQL script which triggers the delete procedure within a Discom production database. Used for the regular (scheduled) deletion from the database.

### Discom-Installation\Discom\Analysis\ProdDb\DbControl

- **ManageFiles:** Contains the batch and settings files for the ManageFiles program.
- **SqlScripts:** Contains the SQL scripts required for the tools UpdateDb, CreateDb, CleanupDb.
- **ProdDbConf.xml:** Defines which SQL scripts are included in the CreateDb/UpdateDb-tool and how these are structured (see section 1.2.3 for details).
- **ShrinkDbConf.xml:** Defines which SQL scripts are included in the ShrinkDb -tool and how these are structured (see section 1.2.3 for details).
- **DbControl:** Shortcut to the executable of the DbControl-GUI. Please note that the shortcut uses the **DiscomSoftwareRoot** environment variable to set the “target” path. If the environment variable does not exist, either create new or adjust the path accordingly. Adding the input argument “-l” to the target, automatically loads all databases specified in the FactoryData. The “start in” path is set to “C:\Discom\Analysis\ProdDb\DbControl”.
- **FactoryData.xml:** Main settings file for the DbControl-GUI. Contains information on the server network infrastructure, e.g. the SQL Server user credentials required for accessing the production databases (see section 1.2.1 for details).
- **DbControlSettings:** Program settings file. Contains information on different application related settings (see section 1.2.2 for details).

## 1.2 Input files

### 1.2.1 FactoryData.xml

The FactoryData.xml contains all relevant information on a server network hosting one or multiple Discom production databases (lines).

```
<?xml version="1.0"?>
- <FactoryModel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http:
  - <Lines>
    - <Line Name="DiscomData">
      - <Archive Path="Archives\DiscomData\" Line="DiscomData">
        <Location Name="PYARE" IsLocal="true"/>
      </Archive>
      - <Collector Name="Discom.Collector.Service.DiscomData">
        <Location Name="PYARE" IsLocal="true"/>
      </Collector>
      - <ResultDb Name="DiscomData_ProdDb">
        - <SqlServer IntegratedSecurity="false" Password="*****" User="sa">
          <Location Name="PYARE" IsLocal="true"/>
        </SqlServer>
      </ResultDb>
    - <Tasks>
      - <Task Name="TestLocal" Type="CleanupDb">
        <Location Name="PYARE" IsLocal="true"/>
      </Task>
      - <Task Name="ManageArchives" Type="ManageFiles">
        <Location Name="PYARE" IsLocal="true"/>
      </Task>
    </Tasks>
  </Line>
</Lines>
</FactoryModel>
```

Fig. 2: FactoryData.xml

It is used by DbControl as well as the program CleanupDb, therefore it needs to be present on all servers on which the latter runs. Before distributing the FactoryData.xml, make sure that it is up-to-date. In DbControl the FactoryData.xml is represented by the “Lines” tree view model (see Fig. 3).

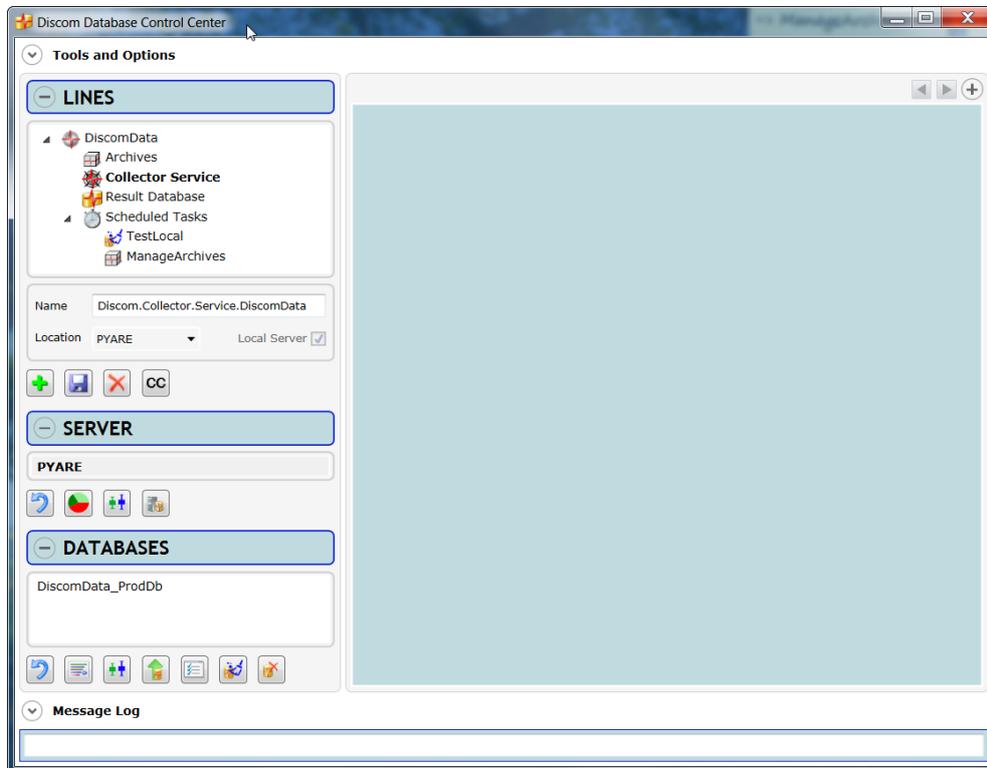


Fig. 3: Lines tree view model (representation of the FactoryData.xml)

The root of the hierarchical model is represented by the “lines” element. **Each “line” element represents a single production line with the following corresponding subordinate elements:**

- **Archives:** The “archives” element is defined by the location (i.e. the server on which the measurement archives are stored) and the path to the archives’ directory.
- **Collector Service:** The “collector service” element is defined by the service name and the location (server) on which the service resides.
- **Result Database:** The “result database” element is defined by the database name and the related SQL Server connection info. This info includes the name of the SQL Server instance, the location (server) on which the SQL Server resides, and the user login info (integrated security, user name and password). If the checkbox “Integrated Security” is checked (i.e. set to true), the user name and password do not need to be specified as the Windows credentials of the currently logged user are used.
- **Scheduled Tasks:** The “scheduled tasks” element contains any Windows Scheduler Task set up for the regular (scheduled) execution of the programs CleanupDb and/or ManageFiles. Each “task” element is defined by the task name, the task type (CleanupDb or ManageFiles), and the task location (server on which the task resides).

The fact that each of these subordinate elements is defined by location provides full flexibility with regard to the server infrastructure (see Figs. 4-6 for three possible server layouts):

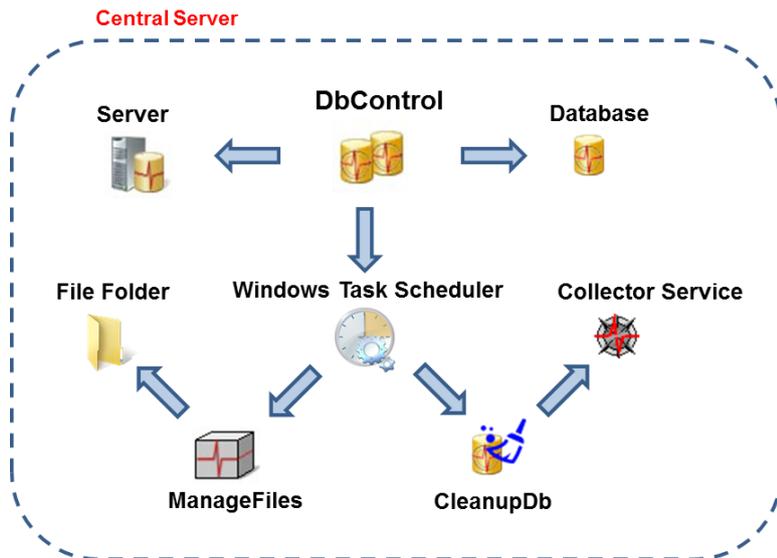


Fig. 4: Stand-alone server layout. All components reside on the same central server.

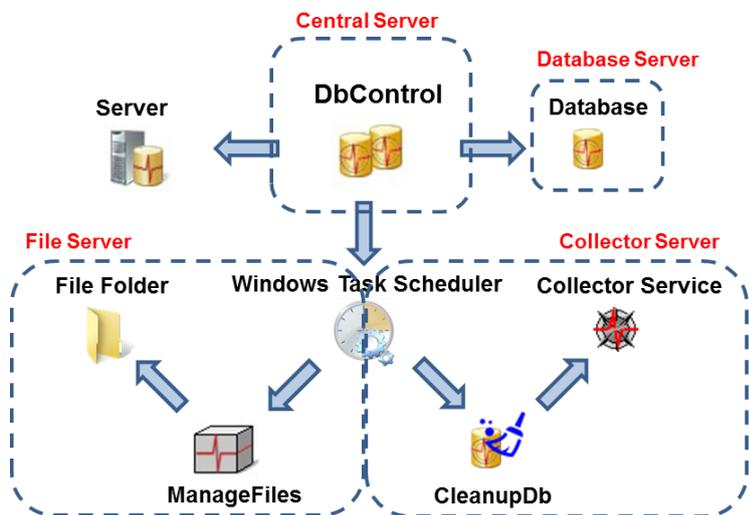


Fig. 5: Server layout where the database, the collector service and the measurement files reside on different servers.

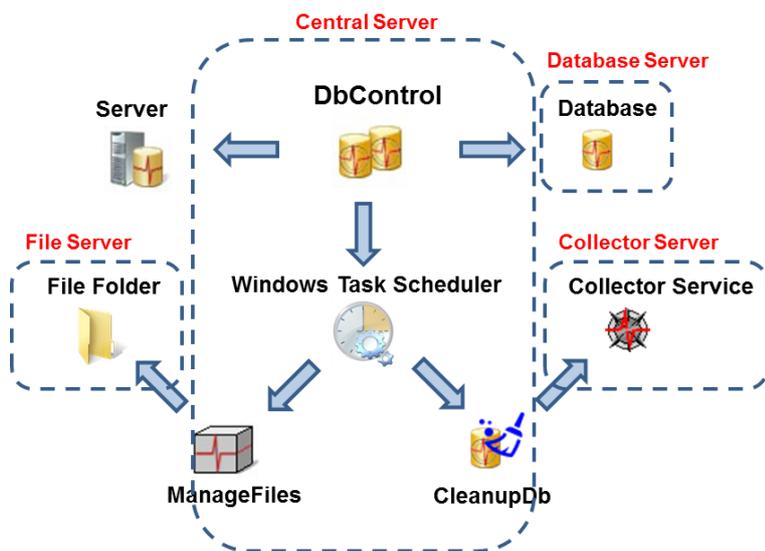


Fig. 6: Server layout where the database, the collector service and the measurement files reside on different servers, but the tasks are set-up on the central server.

By default, the FactoryData.xml contains the info described in Fig. 2, except for the scheduled tasks' info (which by default is an empty list). The location is defined by the server name and the attribute "IsLocal", which indicates whether the location is the local server on which the program DbControl is installed. The default location name ("LocalMachine") of the local server (i.e. "IsLocal" = true) is automatically replaced with the true machine name of the local server upon program startup. The default values can be changed directly in the FactoryData.xml (before program startup) or interactively in the tree view (Fig. 3). Likewise new lines can be added or existing lines removed. See section 3.2 for details.

### 1.2.2 DbControlSettings.xml

The DbControlSettings.xml contains all application relevant information (Fig. 7). This includes:

- **Thresholds:** Lists the so-called threshold items which are used to color-code the database statistics "last measurement input" (see 3.4.1 for details). These items are defined in the program and are automatically added to the DbControlSettings.
- **DiskThresholds:** Lists the so-called disk threshold items which are used to color-code the server disk statistics "free space" and to display a warning in case the free space falls below this threshold. The two entries shown in Fig.2 define general thresholds for a system disk partition and a data disk partition. The "threshold" item defines the threshold value and the "RelativeOrAbsolute" item defines whether this threshold relates to an absolute value in GB or a relative percentage (in relation to total disk space) value. The items "ThresholdColor", "FreeSpaceColor", and "UsedSpaceColor" define the colors of the server statistics bar plot (see 3.4.1). The general "DiskThresholds" entries can be overruled by server and disk specific settings. For example in order to add a new data disk threshold item for server "Server\_1", copy the general threshold item for the data partition (Id 1), give it a new Id (2) and the desired threshold and add the xml element <Server>Server\_1</Server>.
- **DeleteSettingsItems:** Lists the so-called DeleteSettings items which are used to pre-define templates for the Cleanup Database settings. Please see section 3.5.1 for details on the elements of such an item.
- **DiscomSoftwareRoot:** Path to the Discom program files folder. By default set to C:\Program Files (x86)\Discom
- **CleanupDbPath:** Path to the CleanupDb program. By default set to C:\Program Files (x86)\Discom\DbControl\CleanupDb\CleanupDb.exe
- **DbControlStartupPath:** Path to the DbControl startup folder. By default set to C:\Discom\Analysis\ProdDb\DbControl\
- **FactoryDataFile:** Name of the main input file which defines the server infrastructure. By default named FactoryData.xml
- **ManageFilesPath:** Path to the ManageFiles program. By default set to C:\Program Files (x86)\Discom\DbControl\ManageFiles\ManageFiles.exe
- **ManageFilesStartupPath:** Path to the startup folder of the ManageFiles folder. By default set to C:\Discom\Analysis\ProdDb\DbControl\ManageFiles\
- **ManageFilesFile:** Name of the main input file for the ManageFiles program. By default named ManageFilesTask.xml
- **DeletionPassword:** Password for the deletion of data. By default set to "discom"

- **SqlScripts:** Path to the SQL scripts required by DbControl and CleanupDb are stored. By default set to C:\Discom\Analysis\ProdDb\DbControl\SqlScripts\
- **ProdDbFile:** Name of the input file for the creation and update of production databases. By default named ProdDbConf.xml
- **ShrinkDbFile:** Name of the input file for the shrinking of production databases. By default named ShrinkDbConf.xml

```
<?xml version="1.0"?>
- <DbControlSettings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001
  - <Thresholds>
    + <ThresholdItem>
    + <ThresholdItem>
    + <ThresholdItem>
  </Thresholds>
+ <DiskThresholds>
- <DeleteSettingsItems>
  - <DeleteSettingsItem>
    <Name>GearTest</Name>
    <DeleteMain>true</DeleteMain>
    <TimeLimitMain>12</TimeLimitMain>
    <DeleteProtocol>false</DeleteProtocol>
    <CaseProtocol>0</CaseProtocol>
    <TimeLimitProtocol>12</TimeLimitProtocol>
    <UnitsToKeep>1000</UnitsToKeep>
    <UnitsToDelete>2000</UnitsToDelete>
    <TypeAndBench>false</TypeAndBench>
  </DeleteSettingsItem>
  - <DeleteSettingsItem>
    <Name>StandAloneServer</Name>
    <DeleteMain>true</DeleteMain>
    <TimeLimitMain>36</TimeLimitMain>
    <DeleteProtocol>true</DeleteProtocol>
    <CaseProtocol>2</CaseProtocol>
    <TimeLimitProtocol>12</TimeLimitProtocol>
    <UnitsToKeep>1000</UnitsToKeep>
    <UnitsToDelete>2000</UnitsToDelete>
    <TypeAndBench>false</TypeAndBench>
  </DeleteSettingsItem>
</DeleteSettingsItems>
<DiscomSoftwareRoot>C:\Program Files (x86)\Discom</DiscomSoftwareRoot>
<CleanupDbPath>C:\Program Files (x86)\Discom\DbControl\CleanupDb\CleanupDb.exe</CleanupDbPath>
<DbControlStartupPath>C:\Discom\Analysis\ProdDb\DbControl</DbControlStartupPath>
<FactoryDataFile>FactoryData.xml</FactoryDataFile>
<ManageFilePath>C:\Program Files (x86)\Discom\DbControl\ManageFiles\ManageFiles.exe</ManageFilePath>
<ManageFilesStartupPath>C:\Discom\Analysis\ProdDb\DbControl\ManageFiles</ManageFilesStartupPath>
<ManageFilesFile>ArchiveTask.xml</ManageFilesFile>
<DeletionPassword>discom</DeletionPassword>
<SqlScripts>C:\Discom\Analysis\ProdDb\DbControl\SqlScripts</SqlScripts>
<ProdDbFile>ProdDbConf.xml</ProdDbFile>
<ShrinkDbFile>ShrinkDbConf.xml</ShrinkDbFile>
</DbControlSettings>
```

Fig. 7: DbControlSettings.xml

### 1.2.3 ProdDbConf.xml/ShrinkDbConf.xml

These files each define a so-called tree-view of SQL scripts. Scripts with thematic similarities are logically grouped into modules. These modules can be batch-executed, meaning all SQL scripts of a module are sequentially executed. New scripts and modules can be added by defining new items of class "BatchModule" or of class "ExecSqlScript". The attribute "userName" represents the display name, "Batch" controls whether the module item executes in batch mode and the attribute "Path" stores the path to the SQL script.

```
<?xml version="1.0"?>
- <ModuleTree xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  - <Root Batch="false" userName="Shrink" instance="root" class="BatchModule">
    - <Modules>
      - <module Batch="true" userName="Initialize shrink" instance="Init" class="BatchModule">
        - <Modules>
          - <module Batch="true" userName="Set max server memory" instance="MaxServerMemory" class="ExecSqlScript" Path="ShrinkDb\01_
            <Modules/>
          </module>
          - <module Batch="true" userName="Create xTimeLimitsOfTypes" instance="TimeLimitsOfTypes" class="ExecSqlScript" Path="ShrinkDb\
            <Modules/>
          </module>
          - <module Batch="true" userName="Check xTimeLimitsOfTypes" instance="CheckTimeLimitsOfTypes" class="ExecSqlScript" Path="Shrin
            <Modules/>
          </module>
        </Modules>
      </module>
    </Modules>
```

Fig. 8: Excerpt from ShrinkDbConf.xml

## 2. Installation

### 2.1 Network (server) settings

In order for DbControl to remotely access other servers, a number of specific settings need to be made on the central server and its target servers. Most importantly, the user operating DbControl on the central server (thereafter denoted as master user) needs login (and preferably admin) rights on all of the target servers, i.e. either through a domain user with respective rights on the target servers or a local user on each target server with the same credentials (and preferably admin rights). In addition, make sure that:

- **.NET Framework 4 or higher** is installed. In case there is no SQL Server installed on the central server, the SQL Native Clients 10 and 11 may need to be installed (in order for DbControl to establish a connection with the SQL Servers on the target server).
- **Only for workgroup-based networks:** The **LocalAccountTokenFilterPolicy** is set in the registry of each server. This is required for the remote access to the Windows Task Scheduler and the Windows Management Instrumentation (WMI). If not set, run the command prompt: `reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\system /v LocalAccountTokenFilterPolicy /t REG_DWORD /d 1 /f`
- Allow **Remote Scheduled Task Management, File and Printer Sharing (Server 2008), WMI and Remote Service Management through firewalls** of the **target servers**.
- The firewalls of both the central and the target server allow the remote access to the SQL Server on the target server (open TCP Port 1433). The SQL Server itself needs to allow remote connections (Server Properties\Connections).
- The directories Discom (project directory) and Archive on the target server are shared on the network with read-/write-access for the master user.

### 2.2 Copy instructions (stand-alone server operation)

- Copy folder DbControl from Discom-Installation\Discom\Analysis\ProdDb to the directory Discom\Analysis\ProdDb on the **local server**.
- Copy folder DbControl from Discom-Installation\Program Files\Discom to the Discom software root directory on the **local server**.

### 2.3 Copy instructions (network-based operation)

- Copy folder DbControl from Discom-Installation\Discom\Analysis\ProdDb to the directory Discom\Analysis\ProdDb on the **central server**.
- Copy folder DbControl from Discom-Installation\Program Files\Discom to the Discom software root directory on the **central server**.

**In case the programs ManageFiles and CleanupDb are to be run on a server different from the central server**, distribute the following from Discom-Installation\Discom\Analysis\ProdDb\DbControl to the directory Discom\Analysis\ProdDb of this (separate) **target server**:

- ManageFiles
- SqlScripts (**only subfolder CleanupDb**)
- FactoryData.xml (**make sure that it is up-to-date before distributing**)

Since the Discom project folder is shared on the network with write rights for the master user, it is possible to distribute the above mentioned files via DbControl. Open the File Distribution Manager (found in the “Tools and Options”), select the files to be distributed and the servers to which they are to be distributed, and press the play button (Fig. 9).

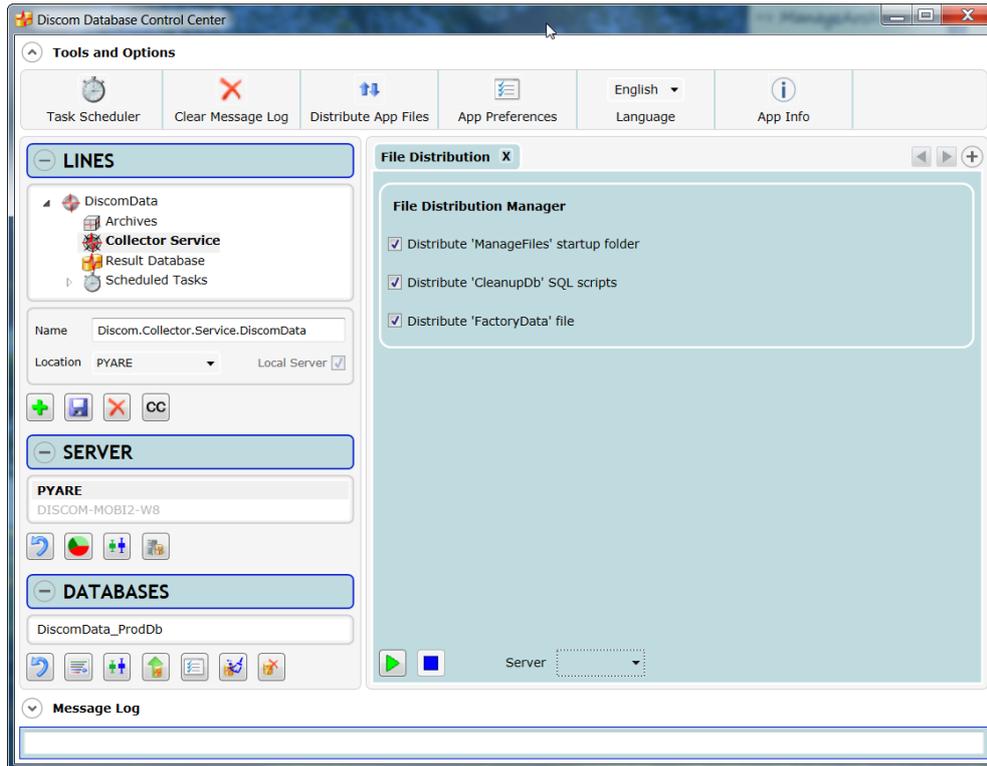


Fig. 9: File Distribution Manager

Since the Discom program files folder **is not** shared on the network, the subfolders “ManageFiles” and “CleanupDb” of the DbControl program folder have to be distributed **manually** from Discom-Installation\Program Files\Discom\DbControl to the DbControl folder in the Discom program files directory of the separate **target server**.

The CleanupDb.exe.config in the CleanupDb folder contains the paths to the FactoryData.xml and to the CleanupDb SQL scripts (Fig. 10). By default, both are located in C:\Discom\Analysis\ProdDb. If these paths differ for a target server, adjust the CleanupDb.exe.config accordingly.

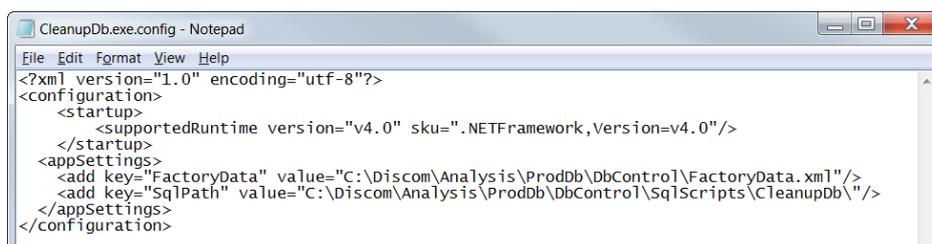


Fig. 10: CleanupDb.exe.config

## 3. Graphical User Interface (GUI)

### 3.1 Views and navigation

The DbControl GUI is subdivided into four parts (Fig. 11): The main “Tools and Options” menu (solid line), the main data view (short dashed line), the explorer view (dotted line), and the message log view (long dashed line).

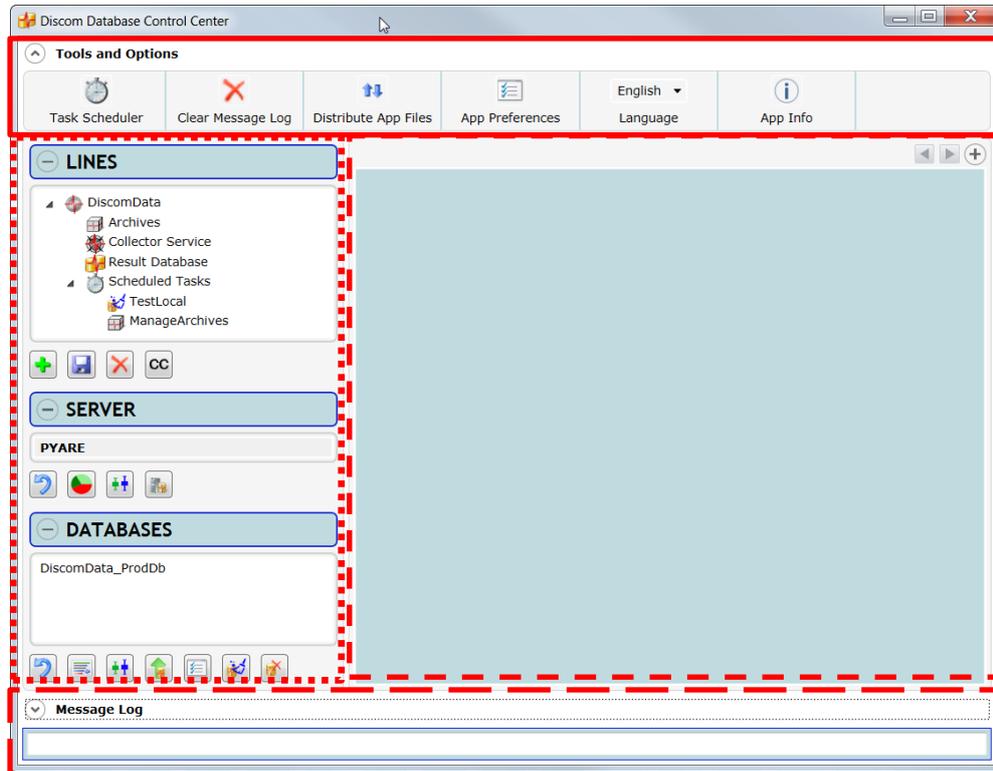


Fig. 11: DbControl GUI.

Most tools and menu items are displayed as closable tabs in the main data view (Fig. 12). Navigate between these tabs either by clicking directly on the tab, by selecting the tab from the tab list, or via the scroll buttons (which are active if the number of tabs exceeds the available window size).

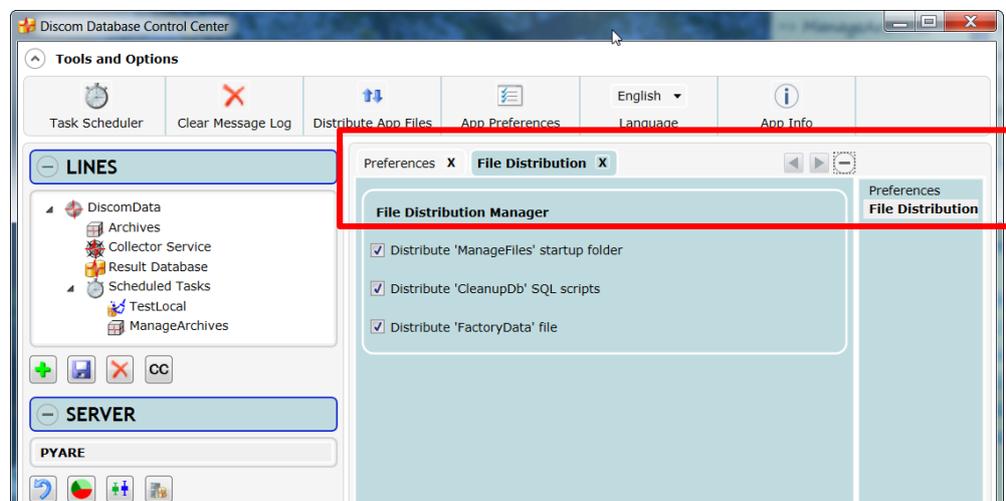


Fig. 12: DbControl main data view.

The message view consists of the message log and the status bar (Fig. 13). While the latter displays all messages generated by DbControl, the former stores only errors, warnings, and the most important info messages. Each message log entry is automatically saved to the InfoLog.xml (found in the DbControl program files folder). The message log can be cleared by selecting the menu item “Clear Message Log”.

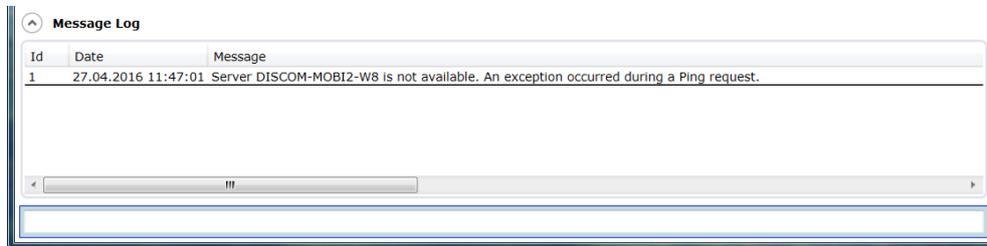


Fig. 13: DbControl Message Log.

The explorer view consists of the three elements “Lines”, “Server”, and “Databases”. While the “Lines” represents the factory model (of different production lines), the “Server” and “Databases” list all servers and databases which the factory model contains. Each of these elements displays a number of buttons, which are hooked up with different functions and tools.

The window space available to the explorer view, respectively the main data view, can be adjusted via moving the mouse over the thin white space between these two elements and when the regular mouse arrow changes to a vertical double arrow (see left image of Fig. 14), hold the left mouse button down and move left or right. If the available window size for the explorer view is less than the size of the explorer view elements, vertical and/or horizontal scroll bars appear (right image of Fig. 14).

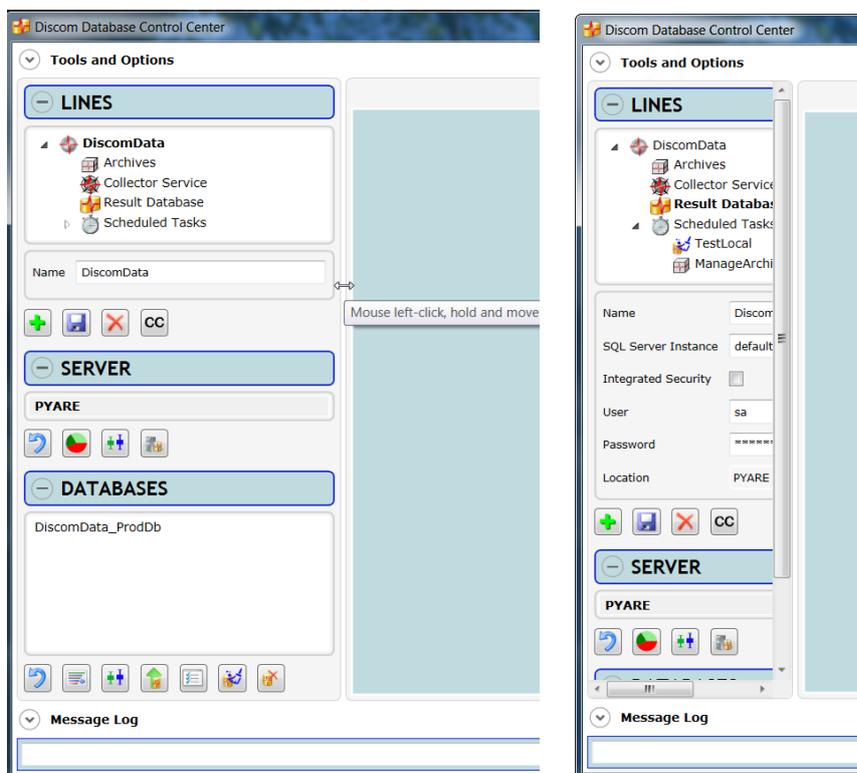


Fig. 14: DbControl explorer and main data view.

## 3.2 Lines

As detailed in section 1.2.1 the “Lines” tree view is the internal representation of the input file FactoryData.xml which defines a factory model consisting of one or multiple production lines, each of which is considered a separate entity (an entity is itself defined by consisting of a single result database, an archive collector, measurement archives, and scheduled tasks). By default, a single line called “DiscomData” with default subordinate elements exists. These are defined:

- **Archives:** The default path to the archives’ directory is “Archives\DiscomData\”.
- **Collector Service:** The default service is named “Discom.Collector.Service.DiscomData”.
- **Result Database:** The default result database is named “DiscomData\_ProdDb”. The default SQL Server connection info is defined by the name of the SQL Server instance (the default value is “default”), and the login info (integrated security set to false, user name set to “sa”, and password set to the typical password).
- **Scheduled Tasks:** By default there are no scheduled tasks defined.

By default, all these elements have their location defined as the local server, representing the stand-alone server layout (see Fig. 4). The location’s default name is a dummy name which is automatically replaced with the local machine name upon program start-up. Upon selecting one of the elements in the lines tree view, the element’s properties are displayed below the tree view (Figs. 15-17). These properties can be freely edited, except for the scheduled task’s properties (which can only be edited via the “Task Scheduler” in the “Tools and Options” menu). After editing, make sure to save the changes. **Note: Please also make sure that the elements to edit or add exist physically (e.g. that the collector service really exists on the specified location) or make sure to create them (e.g. create a result database with the specified name using the tool CreateDb).**

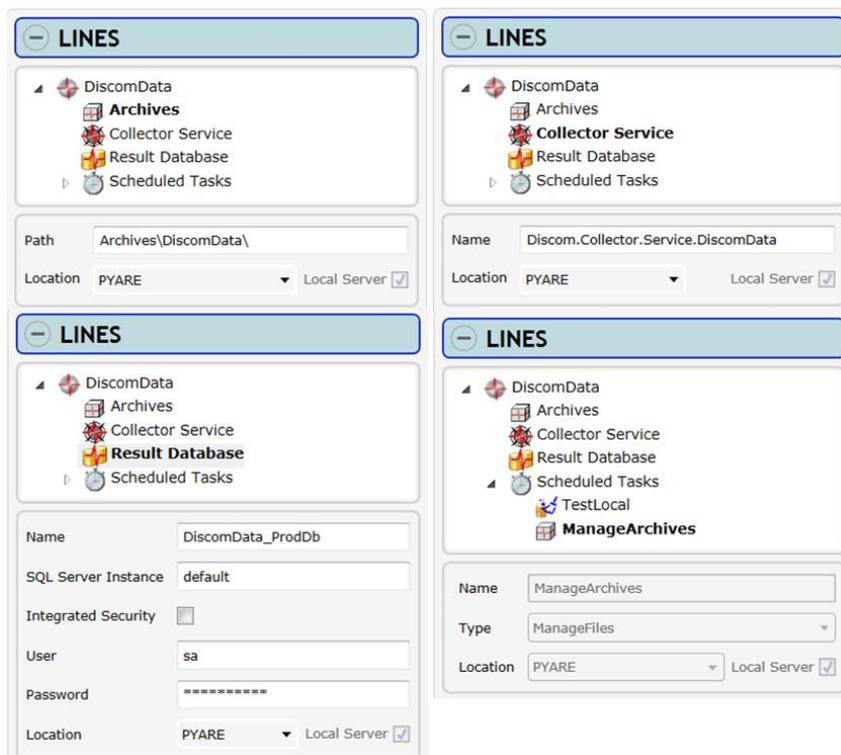


Fig. 15: Lines tree view and its elements.

In case the line name is to be changed and the line already has scheduled tasks defined, then make sure to adjust the line name of these tasks accordingly using the “Task Scheduler”.

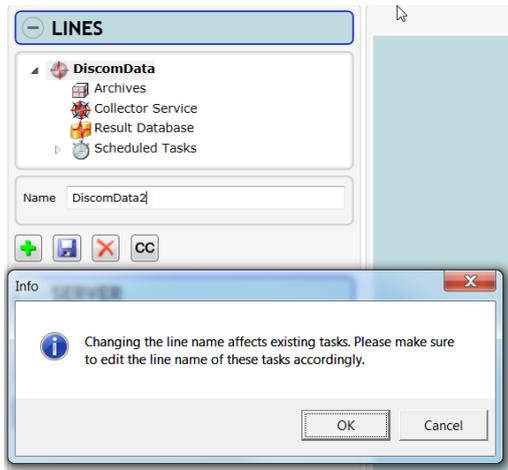


Fig. 16: Lines tree view: Changing the line name.

New elements can be added using the “plus” button (the button with the red frame in Fig. 17). You can add a complete new line (with default elements), a new location (server), or a new task. After selecting the element to add from the drop-down list, the respective properties’ view appears. Fill in the properties and press the “plus” button next to the drop-down list (the button with the blue frame in Fig. 17). In case the adding of new elements should be canceled, press the “cancel” button (next to the “plus” button).

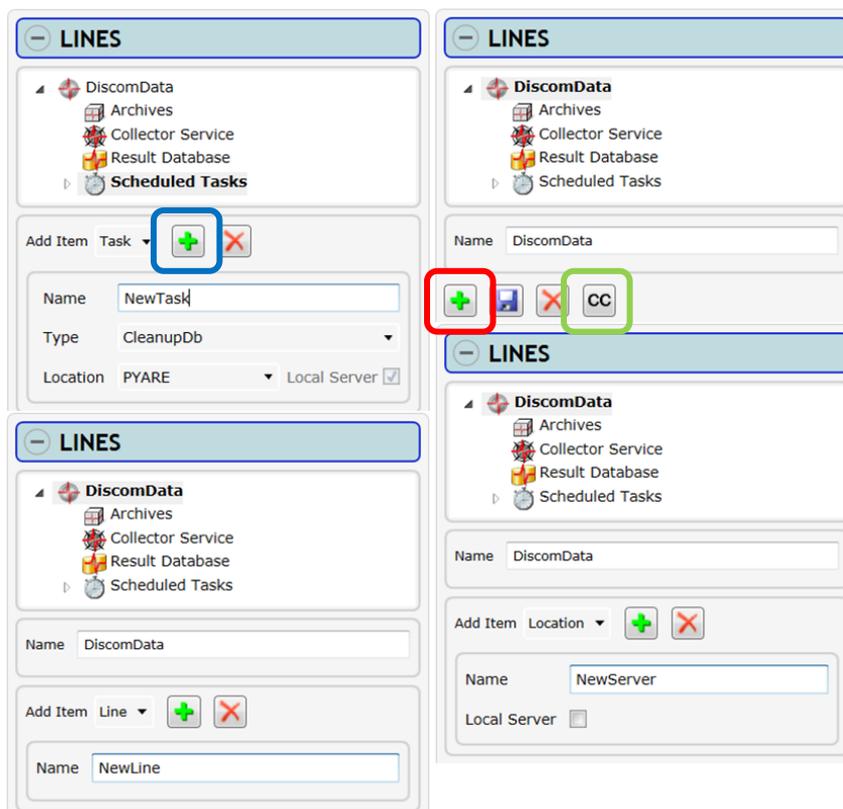


Fig. 17: Lines tree view: Adding new elements.

**Note:** Adding a task element only adds the task's definition to the factory model. The actual Windows Scheduler task has to be set-up using the "Task Scheduler" in the "Tools and Options" menu. Doing so will automatically add the task's definition to the factory model, therefore the option to add the task's definition is only for cases where the Windows Scheduler tasks already exists.

Elements can also be removed from the tree view using the "cancel" button (next to the "save" button). This, however, only applies to line or scheduled task elements. Again, bear in mind that this will only remove the task's definition, respectively the line's definition, from the tree view (and the factory model). It will not remove any of the physical elements behind the model (collector service, scheduled task, etc.). In case that a lot of changes have been made to the tree view, the conformity of the tree view (i.e. whether any element's properties are missing) can be checked using the "cc" button (button with the green frame in Fig. 17).

### 3.3 Servers

The server element of the explorer view lists all servers contained in the factory model and provides all server relevant functions. Upon program start-up, all servers contained in the factory model are automatically loaded (as part of a background thread) into this list. In case a server cannot be reached (via pinging), it is greyed out in the list (Fig 18). The server availability can be checked any time using the "Check Server Availability" button (fourth button from the left in the server view of Fig. 18). Likewise, the server list can be updated by clicking on the update/load button (first button from the left in the server view of Fig. 18).

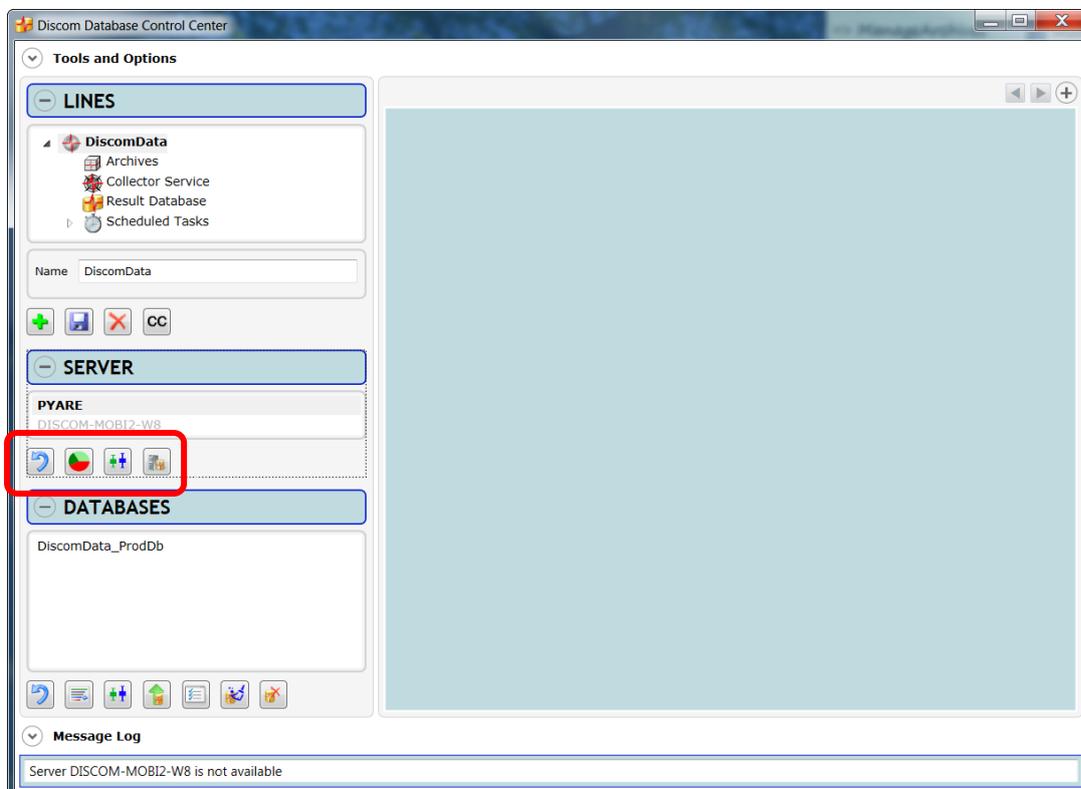


Fig. 18: Server element of the explorer view.

### 3.3.1 Single server statistics

Statistics can be plotted for a single server or for all selected servers in the server list. The single server statistics view displays the usage information on all logical disks of a single server selected from the list (Fig. 19). The update button manually updates the statistics for the respective servers.

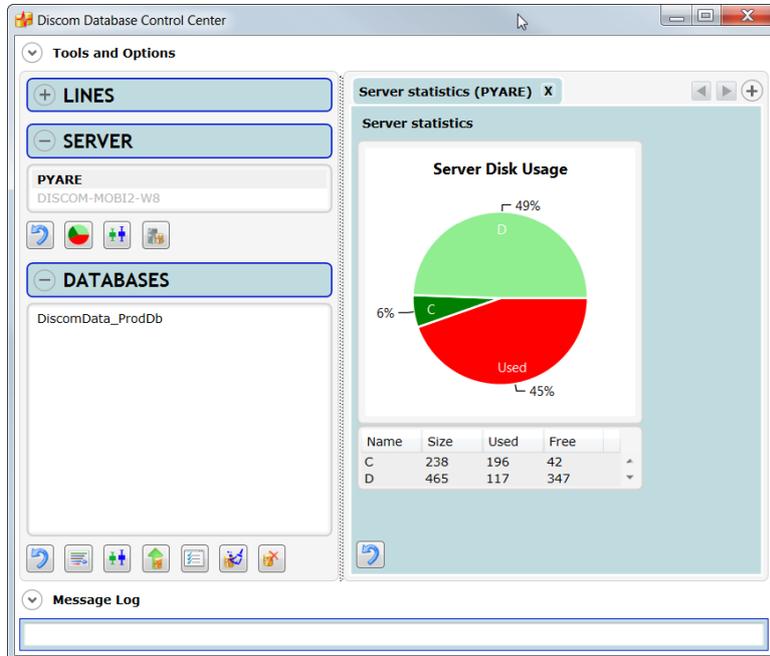


Fig. 19: Single server statistics. The percentages shown in the pie chart relate to the server's total physical disk size. The size, used space, and free space for the logical disks are listed below.

### 3.3.2 Selected server statistics

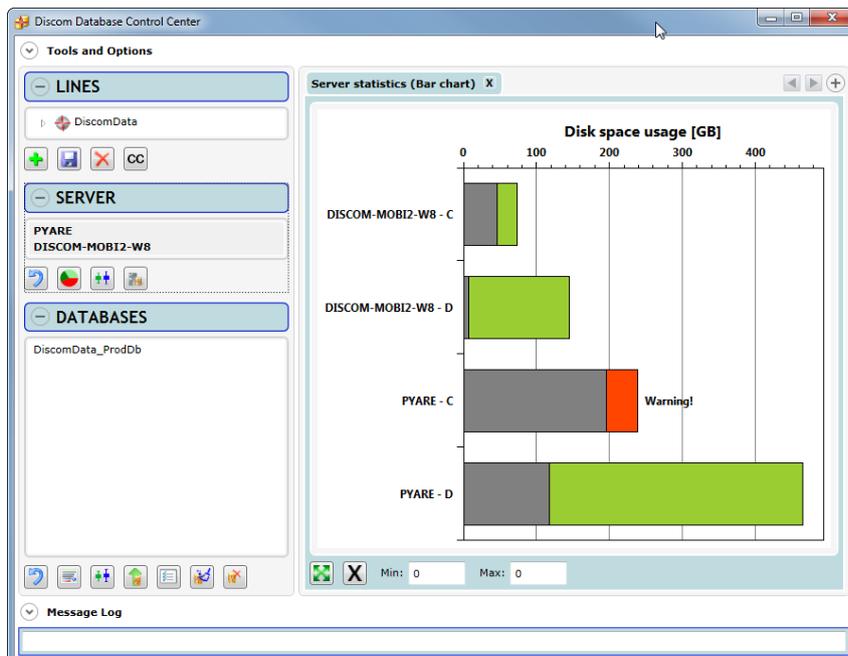


Fig. 20: Selected server statistics. Gray and green depict the used and free disk space respectively. If the free space falls below the general disk thresholds defined in the DbControlSettings (cf. 1.2.2), the free space bar will be colored in red and a warning message is displayed.

The selected server statistics displays the usage information on all logical disks of all selected servers (Fig. 20). **Note: The extraction of the disk info for remote server only works if the WMI (Windows Management Information) remote access is allowed through the firewall of the remote server.**

A left-mouse click on a (used or free space) bar will show the exact disk usage value (Fig. 21). Holding the right-mouse click allows moving along the x- and y-axes. Using the mouse scroll allows zooming into the bar plot. In addition, the x-axis can be scaled to user-defined minimum and maximum values and clicking on the X-button (Fig. 22). The auto-scale button resets the axes to their default values.

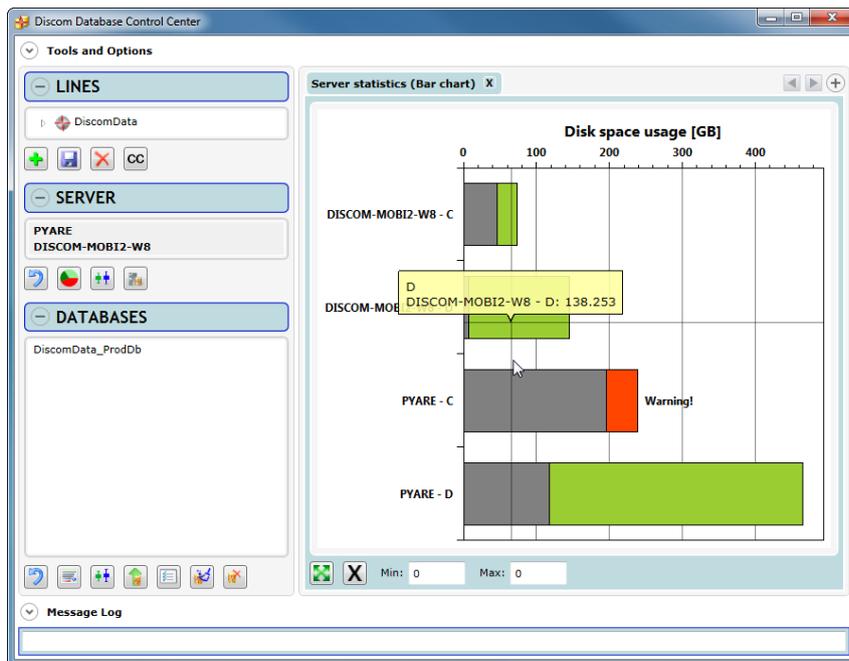


Fig. 21: Selected server statistics: Left-mouse click info.

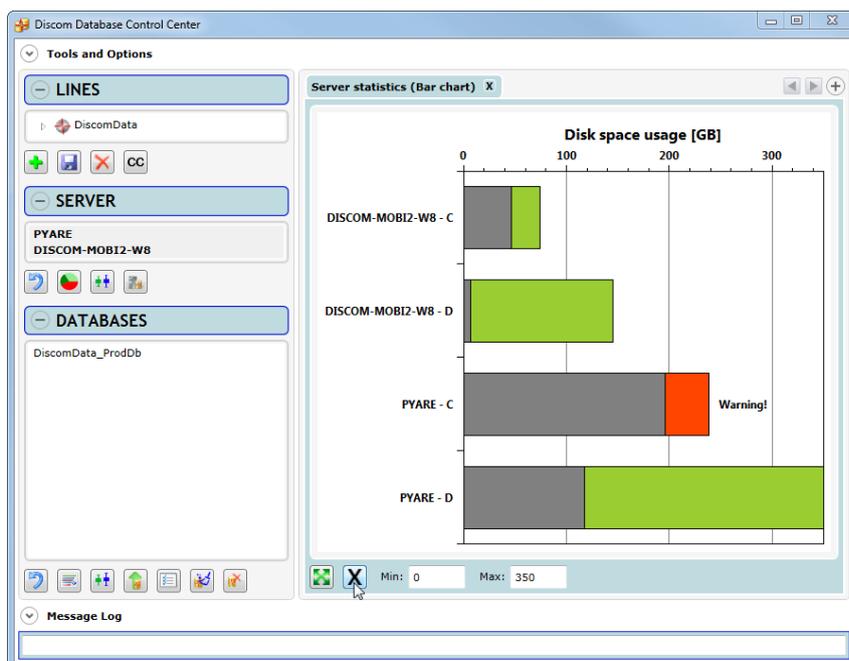


Fig. 22: Selected server statistics: User-defined x-axis scaling.

## 3.4 Databases

The databases element of the explorer view lists all databases contained in the factory model and provides all database relevant functions. Upon program start-up, all databases contained in the factory model are automatically loaded (as part of a background thread) into this list. In case a database cannot be loaded, it is greyed out in the list (Fig 23).

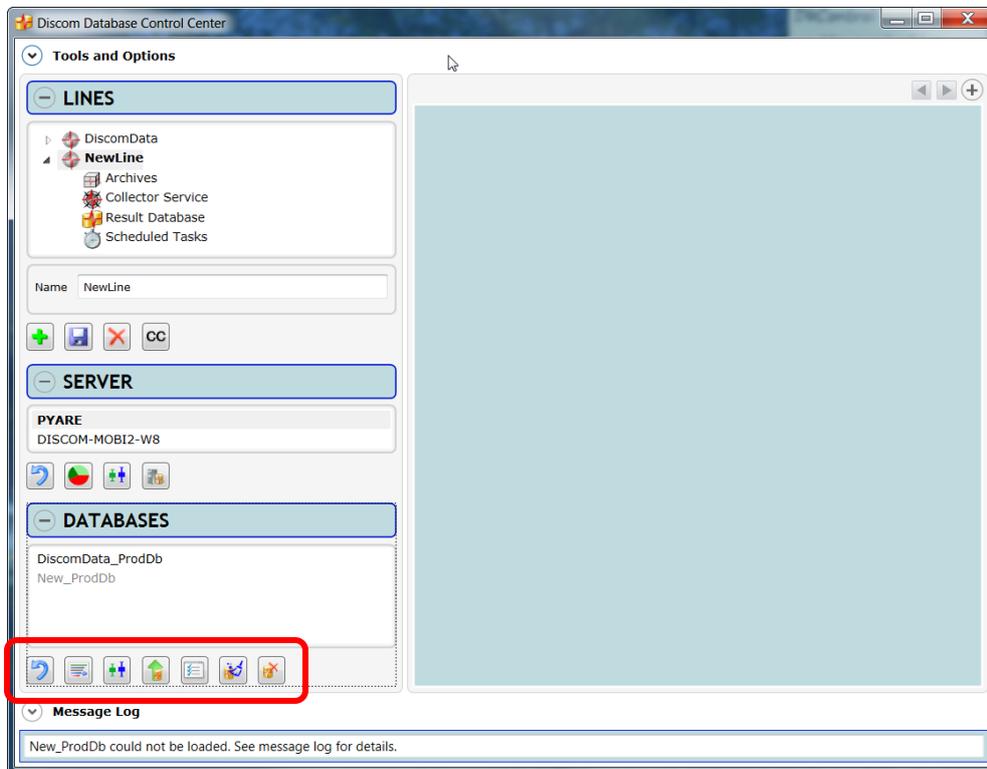


Fig. 23: Databases element of the explorer view.

The database relevant functions and tools are as follows (buttons from the left to right):

- Load/reload database
- View database tables
- View database statistics
- Create/update database
- Cleanup database settings
- Cleanup database
- Shrink database

### 3.4.1 Load/reload database

This button triggers the manual loading of all databases listed in the FactoryData.xml, respectively the “Lines” tree view.

### 3.4.2 View database tables

The “database tables” view lists all database tables along with their row count (Fig. 24). In addition the columns (and their properties) of the currently selected table from the list are shown. To display the tables and columns’ info, select a database from the drop-down list. To update this info click the “update” button. In order to view the currently selected table, specify the number of rows to display and click on “Preview table”.

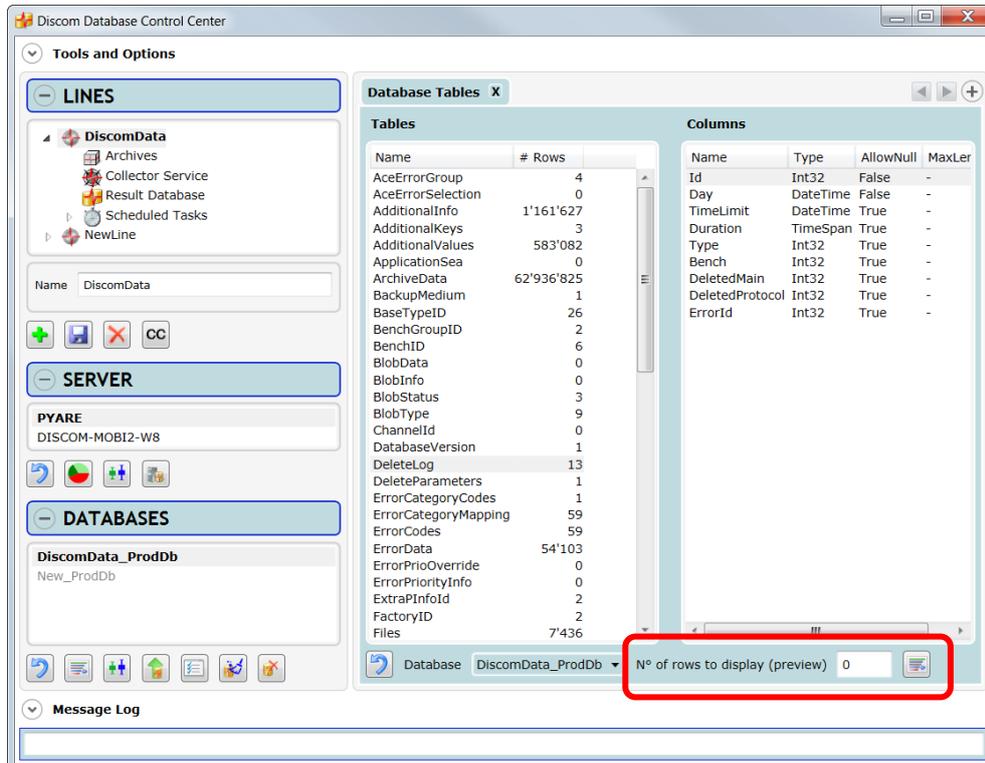


Fig. 24: Database tables. “Tables” lists all the tables of the selected database and “Columns” lists all columns of the currently selected table.

### 3.4.3 View database statistics

The “database statistics” view lists a number of basic database statistics as well as more specific measurement statistics (Fig. 25). To load the info, select one or multiple databases from the drop-down list and click the “update” button. The “basis statistics” view displays the basis statistics of all selected databases, whereas the “selected database” view displays more specific info only for the currently selected database in the “basis statistics” view (Fig. 25).

The basic statistics include:

- **Total Size:** Size of the database file in GB (excluding the logfile).
- **Used Space:** Used space respectively available space for the database. For SQL Server Express editions, the available space is restricted by the maximum database size (4 GB for Express editions SQL Server 2008 and earlier and 10 GB for Express editions SQL Server 2008R2 and later). For SQL Server Enterprise editions, the theoretical maximum database

size is 16 TB therefore the available space of the logical disk which the database resides on is the limiting factor.

- **Last input:** Date & time of the last unit (measurement) input. Color-coded.

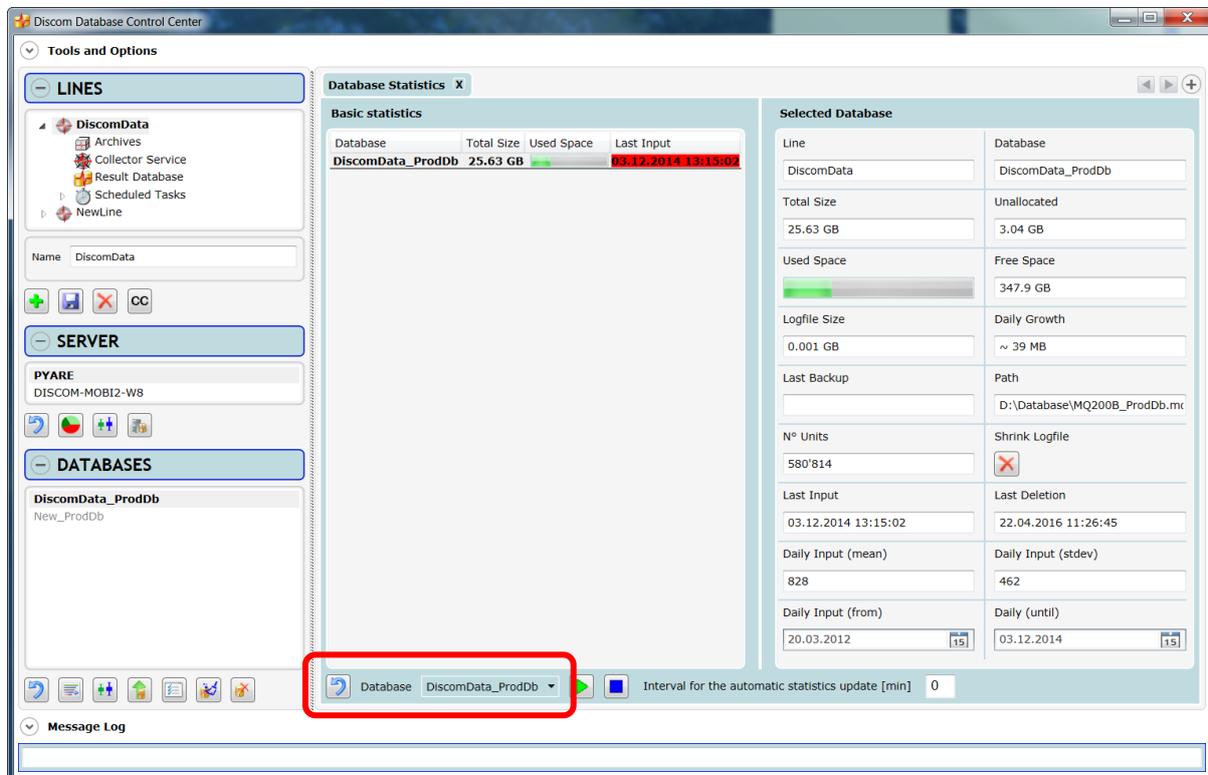


Fig. 25: Database statistics.

The more specific statistics include:

- **Line:** Name of the production line that the database belongs to.
- **Unallocated (Space):** Unallocated space within the database in GB.
- **Free Space:** Available space for the database in GB (see used space of the basis statistics).
- **Logfile Size:** Size of the logfile in GB. Since (theoretically) there is no size limitation for the logfile (even for SQL Server Express editions), the available space of the logical disk that the logfile resides on is the limiting factor.
- **Daily Growth:** The daily growth in database size is only a rough estimate as it is calculated solely based on the size requirement of the single values' data (ProtocolData). The true size growth is therefore underestimated (depending on the overall database size). The formula for the calculation is:

$$\bar{u}_{\text{daily}} * \bar{sv}_{\text{unit}} * sv_{\text{size}}$$

with  $\bar{u}_{\text{daily}}$  = mean daily input of units,

$\bar{sv}_{\text{unit}}$  = mean number of single values' entries per unit (mean of 1000 units),

$sv_{\text{size}}$  = the size requirement of a single values' entry

By default, the average daily measurement input is calculated based on the time interval between the first and last measurement. If the daily input varies strongly across time, the

time interval can be adjusted by specifying the time interval and updating the statistics manually (Fig. 25: “update” button in the database statistics’ view).

- **Last Backup:** Date of the last database backup.
- **Path:** Path to the primary database file (\*.mdf).
- **N° Units:** Total number of units in the database (MainData).
- **Shrink Logfile:** This button triggers the shrinking of the logfile.
- **Last Deletion:** Date & time of the last deletion from the database.
- **Daily Input (mean):** Mean daily input of units.
- **Daily Input (stdev):** Standard deviation of the daily unit input.
- **Daily Input (from):** Basis for the daily input calculation. By default the date & time of the first unit input.
- **Daily Input (until):** Basis for the daily input calculation. By default the date & time of the last unit input.

The measurement statistics (N° Units, Last Input, Last Deletion, etc.) can be automatically updated in order to monitor the unit input into the database. To do so, specify an interval (in minutes) and click the start button (“play” button in Fig. 25). To stop the automatic update, click the stop button. To manually update all database statistics, click the “arrow” button.

In addition, the “Last Input” variable can be linked to a threshold date, in order to aid the monitoring of data input. If the last input exceeds this threshold, the entry in the “Measurement statistics” list lights up in a pre-defined color. To define a threshold, go to the menu item “App Preferences”. In the “statistics” tab, click on the “plus” button, then type in a threshold, select its interval, pick a color and press save (Fig. 26).

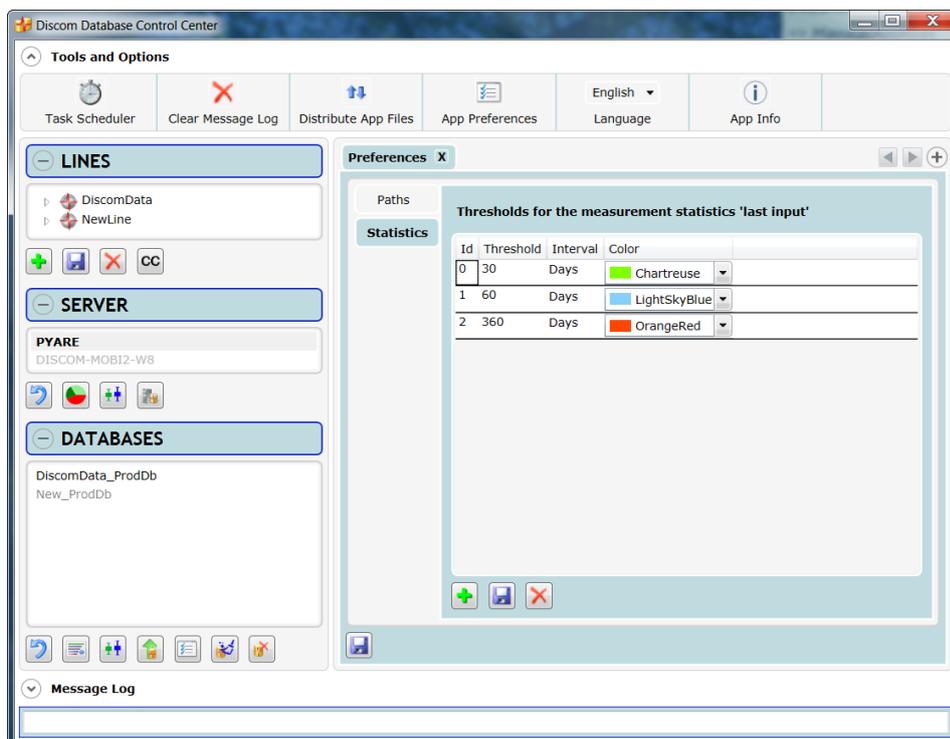


Fig. 26: App Preferences – Statistics: Color thresholds.

In this example, “Last Input” entries which are not older than 30 days will be displayed in green, entry dates between the last 30 days and the last 60 days will be displayed in light blue, entry dates between the last 30 and the last 360 days will be displayed in red, and entry dates older than 360 days will not be color-coded (compare Figures 25 and 26).

### 3.4.4 Create/update database

The tool “Create/Update Database” substitutes the “DbCreator” and “CheckDb” of the Discom.Cp40 program (i.e. the Cs40Bin no longer has to be distributed to all servers of a factory’s network). Its tree-view structure, however, is adopted from Cp40. SQL scripts (yellow icons in the module tree of Fig. 23) are grouped into modules (box icons in the module tree of Fig. 23).

Executing a module executes all its subordinate elements which have their property “Batch Execute” activated (see check box in Fig. 23). In case the subordinate element is a SQL script it is executed; in case the child element is a module, the execution command propagates down this module executing all its child elements in turn (again only those which have their property “Batch Execute” activated). In case the property “Batch Execute” is changed, save the changes back to the ProdDbConf.xml (“save”-button).

In order to execute a module or SQL script, select it in the module tree of the “Create/Update” view and press “play” (Fig. 27). If no element is selected or if the top icon is selected, the complete module tree will be executed. Before executing, however, make sure to fill in the settings. **Note:** One crucial difference to the batch execution of the Cp40 program is that if one of the SQL scripts causes an error, the batch execution won’t quit immediately as would be the case for the Cp40 program, instead the batch execution will continue until all scripts have finished executing. In case of an error, error message will be written to the “Message Log” and show up in the status bar.

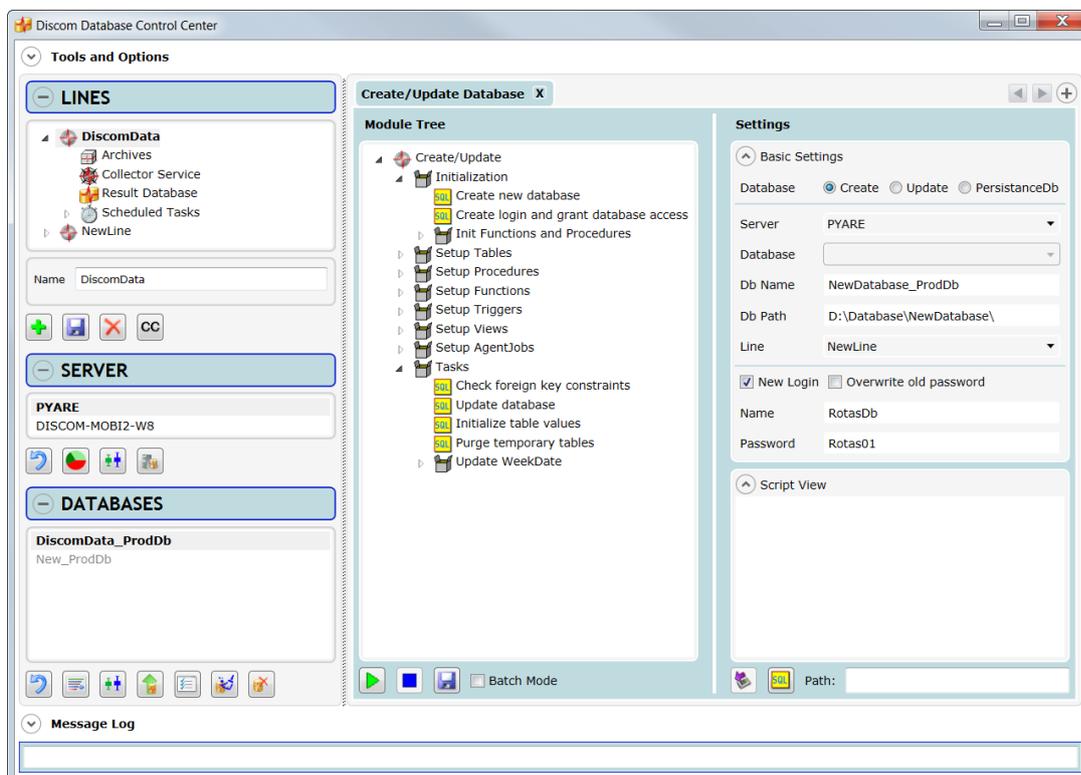
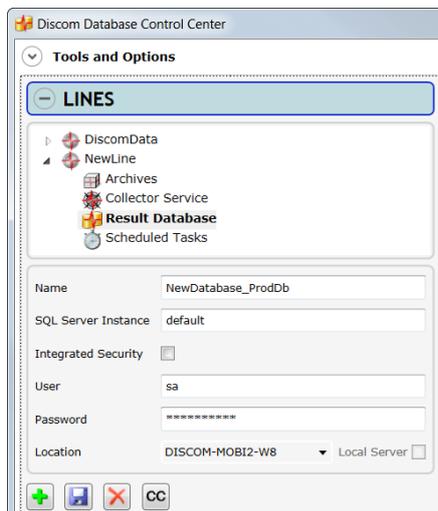


Fig. 27: Create Database.

The basic settings include:

- **Create/Update/PersistenceDb:** Select button for the database creation, update, or creation of the database used by DISCOM's statistical analysis software WebPal.
- **Server:** SQL Server on which to run the selected tool.
- **Database:** Database to be updated.
- **Db Name:** Name of the database to be created.
- **Db Path:** Directory where the database files are created.
- **Line:** Select the line to which the new database should belong to.
- **New Login:** Checkbox for activating the login script
- **Name:** Additional user login (default: RotasDb). The specified user is granted access to and ownership of the new database.
- **Password:** Password of the additional user login.
- **Overwrite old password:** If checked, the current password of the specified login (if existent) is overwritten with the new password.

Prior to creating a new production database, the SQL-Server login data has to be specified in the "Result Database" element of the line which the new database should belong to (Fig. 28). If the "create" button (see Fig. 27) is set, then the option "New login" is activated and set to the user "RotasDb" by default. **Note: This should not be changed, as this will grant "RotasDb" database ownership.** In addition, if a general "RotasDb" login for the SQL-Server does not yet exist, it will thus be created. If the password is not specified, the typical "Rotas01" password will be used. In case the general "RotasDb" login already exists, the old password can be overwritten by checking "Overwrite old password" and specifying a new password. By pressing of the "play" button the batch execution of the SQL scripts is started.



**Fig. 28: Lines: Add the SQL connection info to the "Result Database" element before creating/updating a database.**

An existing database is updated using the same set of SQL scripts which also create a new database. Again, the SQL connection info of the existing database(s) has to be specified in the "Lines" tree view. Checking the box "Update" will automatically enable a drop-down list of all available databases for the selected SQL Server (Fig. 29). Since in case of existent databases it is safe to assume that the RotasDb user already exists, the "new login" fields are disabled by default.

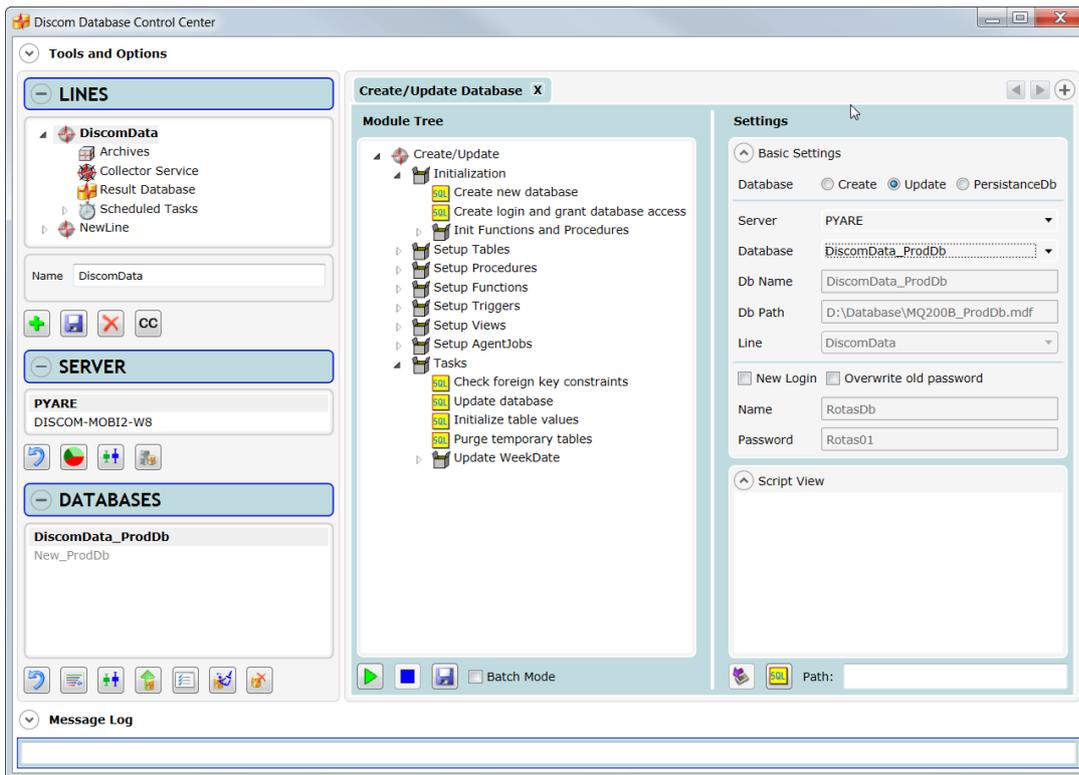


Fig. 29: Update Database.

To start the update process, select the root element of the module tree (“Create/Update”) and press the “play” button. All related print statements of the SQL scripts are written to a result log window and the message log (Fig. 30). In addition, the SQL script icons in the module tree are color-coded indicating whether they executed ok or triggered an error (Fig. 31, green indicates ok, red indicates error).

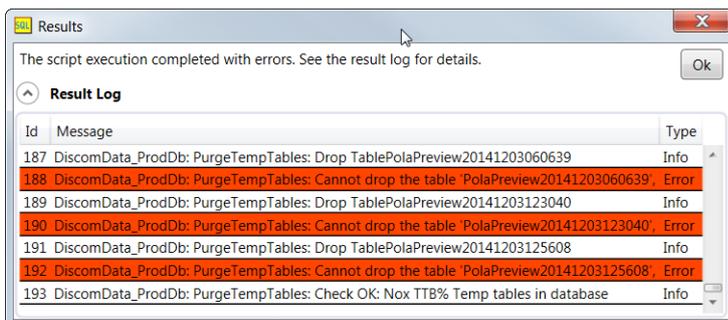


Fig. 30: Result log window: Displayed after the SQL script execution finishes.

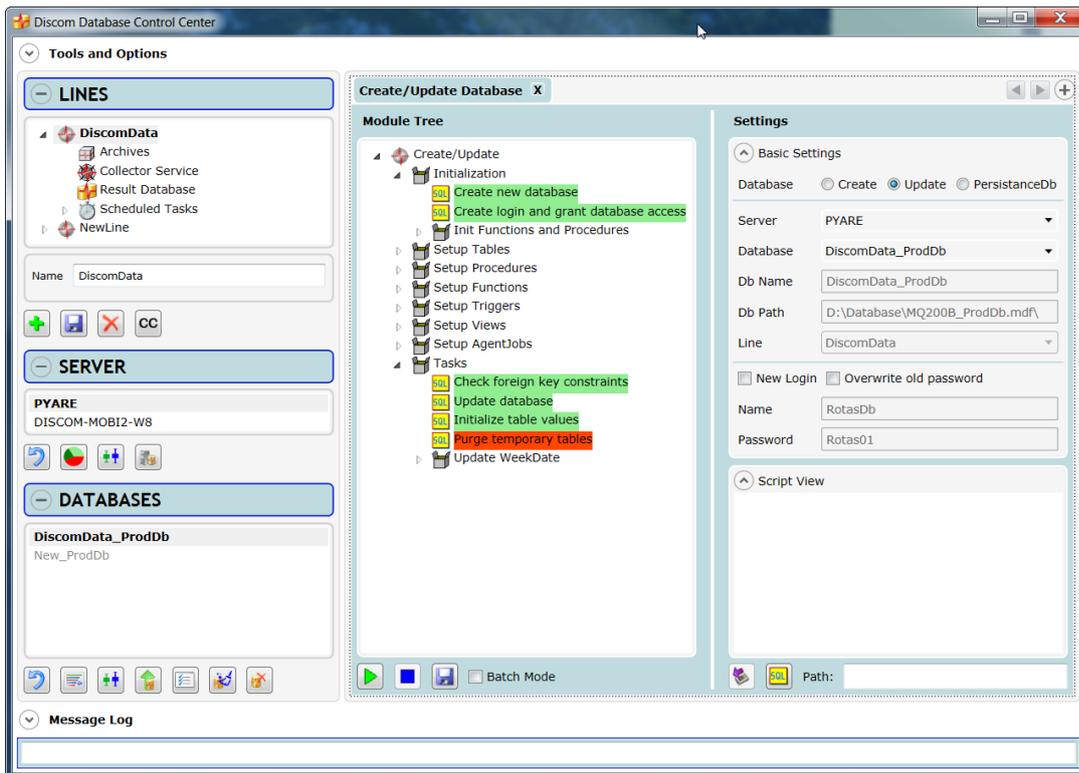


Fig. 31: Color-coded module tree. After the SQL script execution finishes, the icons are color-coded to indicate whether they executed ok or triggered any error. Green indicates ok, red indicates error.

The code of a SQL script can be viewed in the “Script View” by selecting it from the module tree, and pressing the “SQL” button below the “Basic Settings” view (Fig. 32).

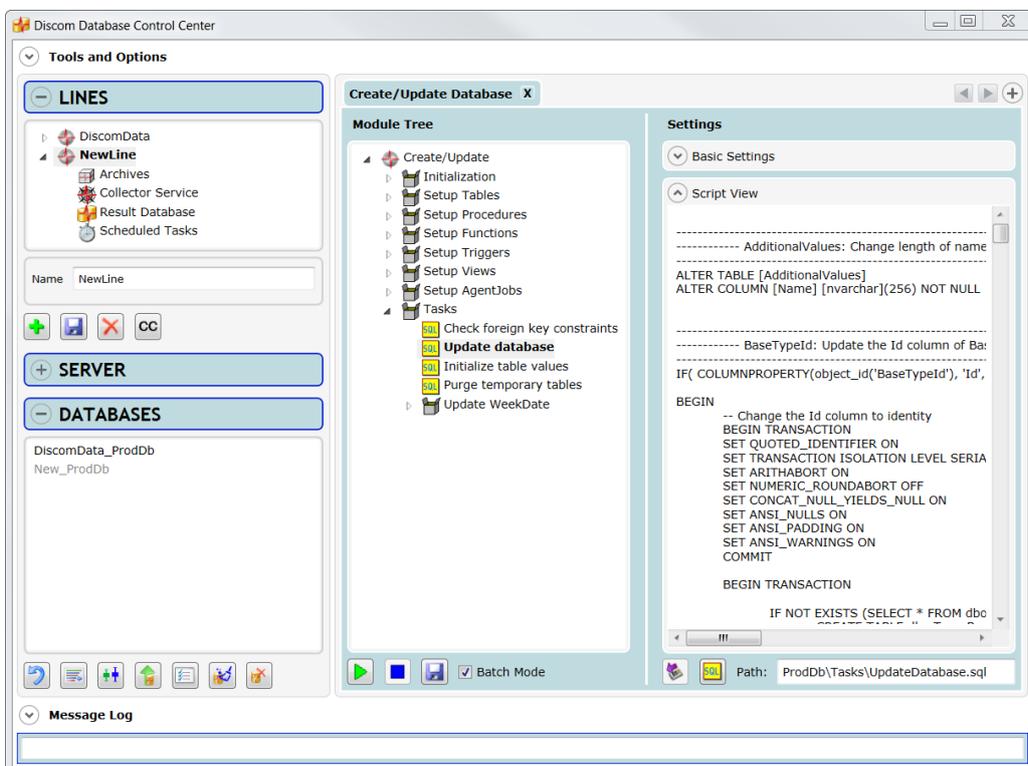


Fig. 32: Script view of the “Create/Update Database” tool.

The creation of the so-called persistence database required for the DISCOM statistical production analysis software WebPal is triggered by selecting the “PersistenceDb” button, selecting a SQL Server and then clicking the “Create PersistenceDb” button (red frame in Fig. 33).

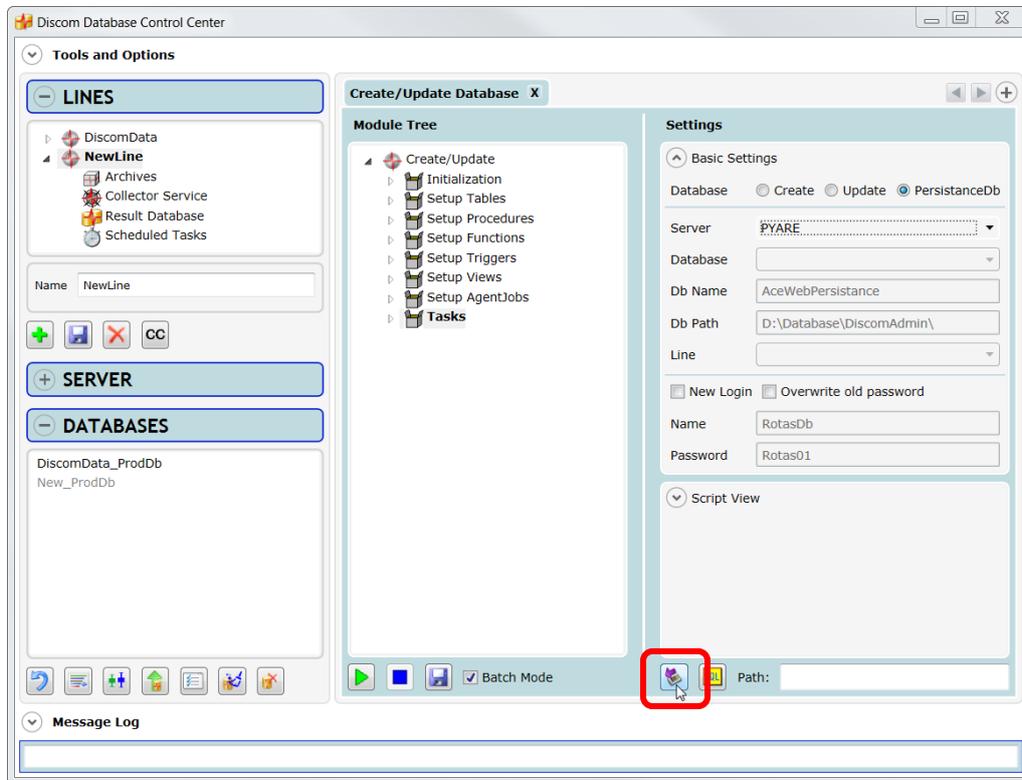


Fig. 33: Create persistence database.

### 3.4.5 Cleanup database

DbControl provides four different ways of deleting data from the database (i.e. to clean up the database):

- Deleting single or multiple measurements by using SQL query syntax.
- Deleting multiple measurements using a stepwise procedure (manually/regular schedule).
- Deleting a large number of single values' data using a one-time procedure (shrinking).
- Deleting all data from a database (truncating the database).

#### Deleting single or multiple measurements by using SQL query syntax

Of these approaches, deleting single or multiple measurements using SQL query syntax is the most straightforward approach. Click the “cleanup database” button (green frame in Fig. 34) and then select the database to delete from. Selecting a database automatically queries whether the collector service associated with the database' line is active. The dates of the first and last measurements as well as the total number of measurements are also queried (Fig. 34). The “Type” and “Bench” drop-down boxes allow filtering the measurements according to type and bench, the “From” and “Until” date boxes allow filtering the measurements according to a period in time, and the “Serial N” field allows filtering the measurements according to serial number. The latter allows for filtering only

parts of the serial number, e.g. if “555\*” is specified, all measurements with serial numbers starting with a “555” will be selected. If “\*555” is specified, all measurements with serial numbers ending with a “555” will be selected. If “\*555\*” is specified, all measurements with serial numbers that contain a “555” will be selected. The character “%” can also be used instead of the asterisk.

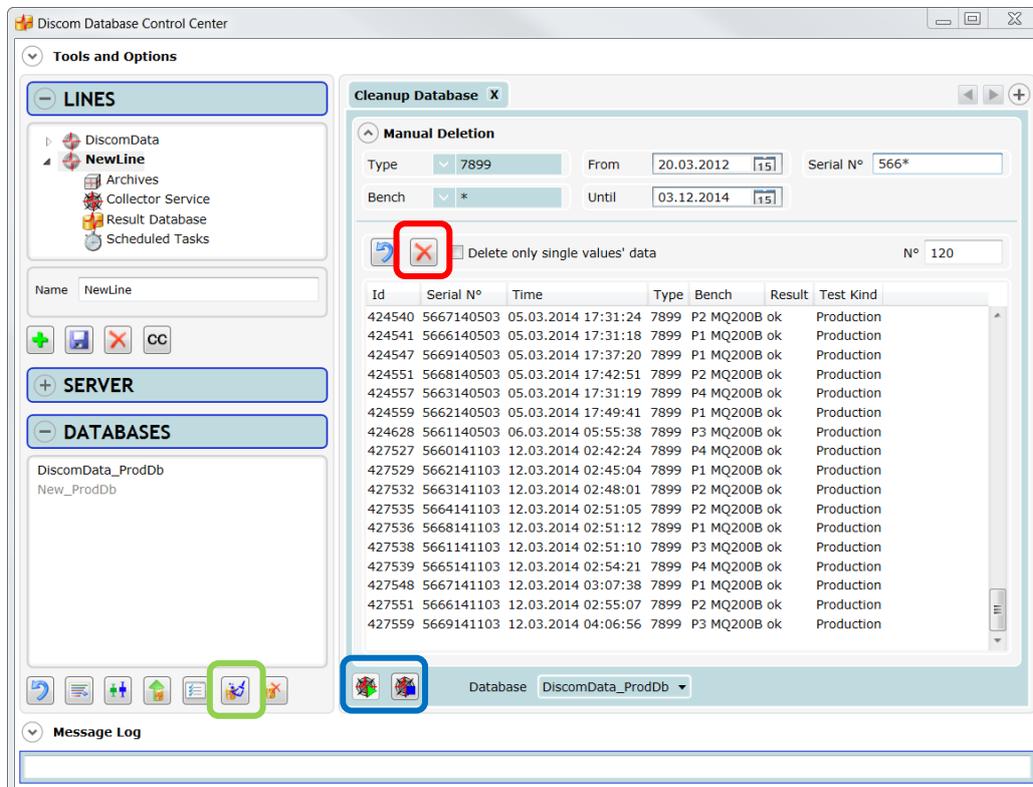
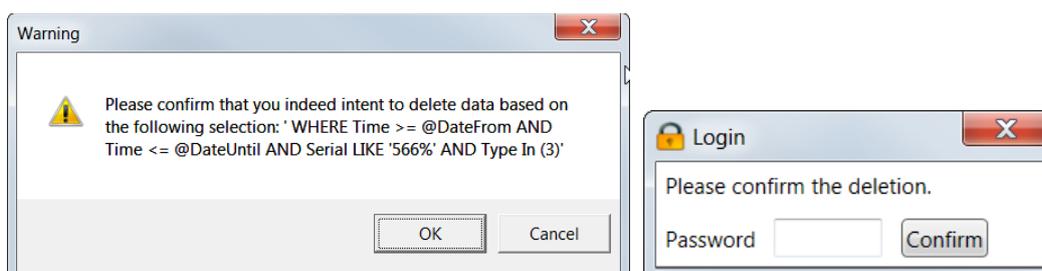


Fig. 34: Cleanup database: Manual deletion using SQL query syntax.

After having chosen the desired filter variables, click on the “load/update” button to load the data which meets the filter criteria into the view. If all displayed measurements are to be deleted, click on the “delete” button (red frame in Fig. 34). The following pop-up window informs the user about the chosen filter criteria and prompts the user to confirm or cancel. If confirmed, a password needs to be entered as an extra step of security. If confirmed, the SQL script “ProcDelete” (contained in the SqlScripts\CleanupDb folder of the configuration files) is executed using the selection string displayed in the pop-up window. This script contains a number of delete statements which deletes all data associated with the selection from the respective tables in the database. If only the single values’ data (data in table ProtocolData) of the selection is to be deleted, check the box “Delete only single values’ data” before clicking delete (Fig. 34). If only a subset of the filtered measurements is to be deleted, select the measurements to be deleted from the list and click delete.



**Note:** In case a large number of measurements is to be deleted (> 1000 units), executing the delete statement might conflict with the collector service's trying to sort new measurements into the database (i.e. due to the fact that the delete statement holds an exclusive lock on the data tables in the database, therefore all input statements will fail). In order to avoid such conflict, the collector service should be stopped before starting the deletion and the re-started once the deletion completes. In order to do so use the "stop collector service" and "start collector service" buttons (blue frame in Fig. 34).

### Deleting multiple measurements using a stepwise procedure

The deletion of measurements from the database using the stepwise procedure was implemented in order to delete small chunks of data on a regular schedule to avoid a constant size growth of the database. The stepwise procedure requires the following:

- To set-up the database tables "DeleteLog", "DeleteParameters", "TimeLimitsOfTypes", as well as the stored procedurs "DeleteData", "TimeLimitsRenew", "TimeLimitsPreview" (respectively the stored procedures "DeleteData\_TypeAndBench", "TimeLimitsRenew\_TypeAndBench", "TimeLimitsPreview\_TypeAndBench") using the update database tool (databases created with DbControl already have those tables and procedures implemented).
- To parameterize the stepwise procedure via the "CleanupDb" settings.

To parameterize the stepwise procedure, click the "CleanupDb settings" button (red frame in Fig. 35) and select the database to parameterize. Either parameterize manually or select one of the templates from the "Template" list, which can be defined in the DbControlSettings.xml.

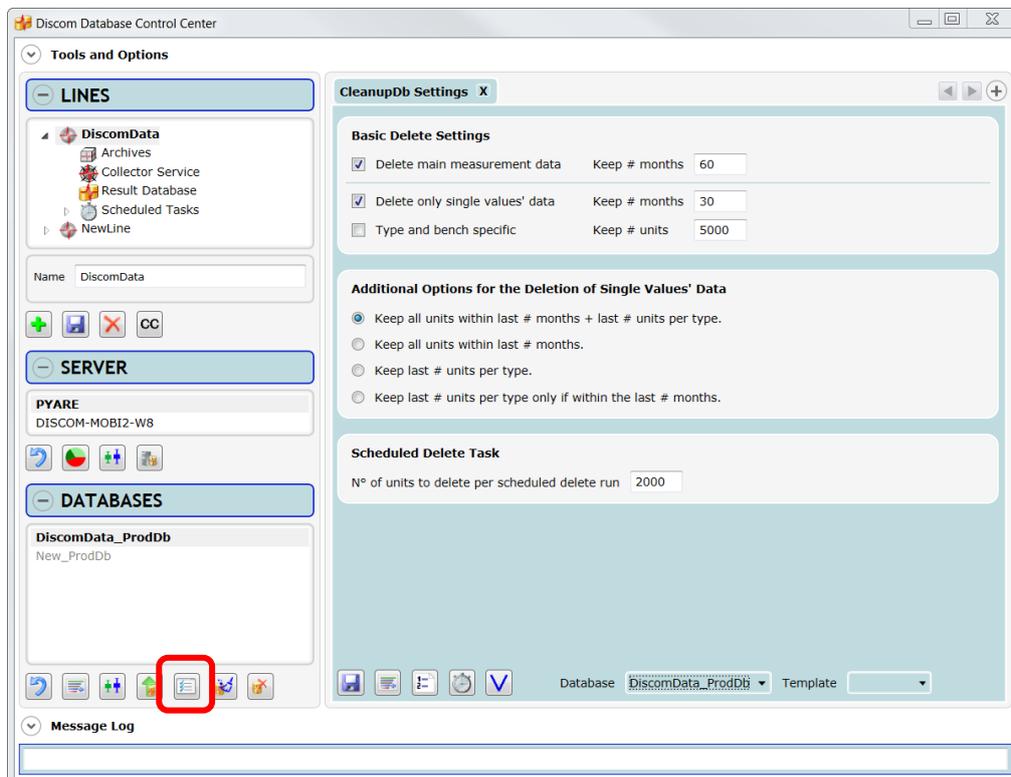


Fig. 35: Cleanup database settings.

In order to activate the cleanup of the basic measurement and single values' data and/or only of the single values' data, click "Delete main measurement data", respectively "Delete only single values' data". The basic measurement data comprises all relevant meta data (type and bench of the unit measured, result and potential error information of the measurement, time of measurement, etc.). The single values' data comprises the actual values of the measured variables (crest, peak, rms, spectral value, etc.). Upon clicking "Delete main measurement data" and/or "Delete only single values' data", pop-up window will prompt the user to confirm the activation (password set in the DbControlSettings.xml). After confirming, fill in the required fields of the settings. The basic delete settings are:

- **Keep # months:** All units measured within the last # number of months are kept; all units measured prior to this time frame are deleted.
- **Keep # units per type:** Additional parameter for the deletion of single values' data. See the following description.
- **Type and bench specific:** If activated the # units per type are kept for each bench.

The additional options for the deletion of single values' data are:

- **Keep all units within last # months + last # units per type:** All single values' data of the units (per type) measured within the last # number of months + at least the single values' data of the last # number of units per type are kept.
- **Keep all units within last # months:** All single values' data of the units (per type) measured within the last # number of months is kept.
- **Keep last # units per type:** All single values' data of the last # number of units per type is kept.
- **Keep last # units per type only if within the last # months:** Single values' data of the last # number of units per type is kept only if the units were measured within the last # number of months.

The regular delete task option has only one parameter which is the number of units to delete per regular (scheduled) delete run. **Note: Always save the changes made to the settings back to the database by clicking on the save button. This triggers the (re-)calculation of the database table "TimeLimitsOfTypes" which is the base for the stepwise deletion procedure.**

The "CleanupDb settings" view buttons provide the following functions:

- **Save changes:** Saves the changes made to the settings back to the database and recalculates the delete table (TimeLimitsOfTypes) which represents the base for the delete procedure.
- **View delete table:** Displays the delete table which gives an overview of the number of units (per type) to keep, respectively to delete.
- **View delete log:** Displays the delete log. Each time the delete procedure within the database is run, an entry will be written to this log.
- **Renew time limits manually:** This allows the manual re-calculation of the delete table based on the Cleanup settings.

- **Preview changes:** In order to check how changes to the Cleanup settings affect the delete table, this will calculate a temporary delete table reflecting the changes – without changing the original delete table or the original delete parameters (unless “save” is pressed).

After parameterizing the stepwise procedure, if the goal is:

- To set-up the regular cleanup task, go to “Task Scheduler” (see section 3.5).
- To manually delete stepwise, click the “Cleanup Database” button (Fig. 36).

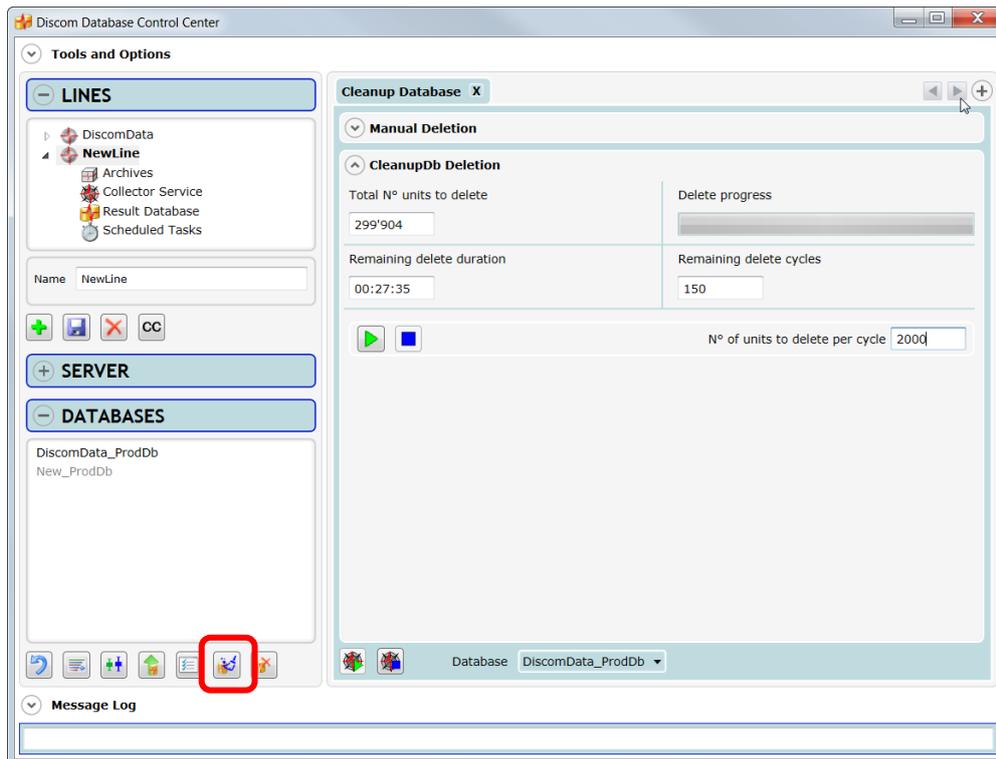


Fig. 36: Cleanup database: Stepwise deletion (“CleanupDb” deletion).

After selecting the database and stopping the collector service (if active), specify the number of n° units to delete per step (cycle) and start the deletion by pressing “play” (red frame in Fig. 36). The deletion process runs as a background thread and loops over the table “TimeLimitsOfTypes” where the number of units to delete per type and/or bench are defined (Fig. 37). The process can be stopped any time by clicking the stop button.

Type Id	Type Name	Bench Id	Bench Name	# Units to Delete	# Units to Keep	Time Limit
0	-			0	583373	19/03/2012 16:22:46
1	3511			0	0	22/09/2013 09:39:28
2	3500			41250	5000	10/09/2013 06:56:50
3	7899			8196	5000	13/09/2013 11:09:57
4	3505			0	5	23/03/2012 17:00:38
5	7777			0	1	26/03/2012 13:03:05
6	3521			249591	5000	12/09/2013 09:04:53
7	9999			0	402	10/07/2012 13:24:08
8	7119			0	4	17/08/2012 10:35:32
9	8888			0	214	29/08/2012 14:55:54
10	3739			0	172	19/10/2012 15:59:04
11	7889			0	2297	08/05/2013 19:27:19

Fig. 37: “TimeLimitsOfTypes”.

## Deleting a large number of single values' data using a one-time procedure (shrinking)

Large chunks of single values' data can also be deleted by applying the one-time procedure of shrinking the database. Like the stepwise manual deletion, this approach loops over the entries defined in the table "TimeLimitsOfTypes" but instead of directly deleting the single values' data of the units which are to be deleted, the single values' data of the units which are to be kept are copied to a new database file. After the copying completes the single values' table is deleted from the database and is replaced with the new single values' table in the new secondary database file.

The main difference to the stepwise manual deletion is that (1) only single values' data can be deleted, (2) a bulk copy operation is more efficient than deleting huge chunks of data, and (3) the shrinking procedure includes physically shrinking the primary database file therefore reducing the database size (depending on the amount of single values' data deleted). Furthermore, by copying the data to be kept to a new database file, a new improved index structure for the single values' table is implemented (additional size reduction of ~ 50 %) and the so-called index fragmentation is avoided.

To utilize this approach, click on the "Shrink database" button (rightmost button in the "Databases" view). Select the SQL Server and the database to shrink (Fig. 38). Before starting the shrinking process by pressing "play", make sure that the database cleanup is parameterized (see "Deleting multiple measurements using a stepwise procedure") and the database table "TimeLimitsOfTypes" really specifies units to delete, respectively to keep.

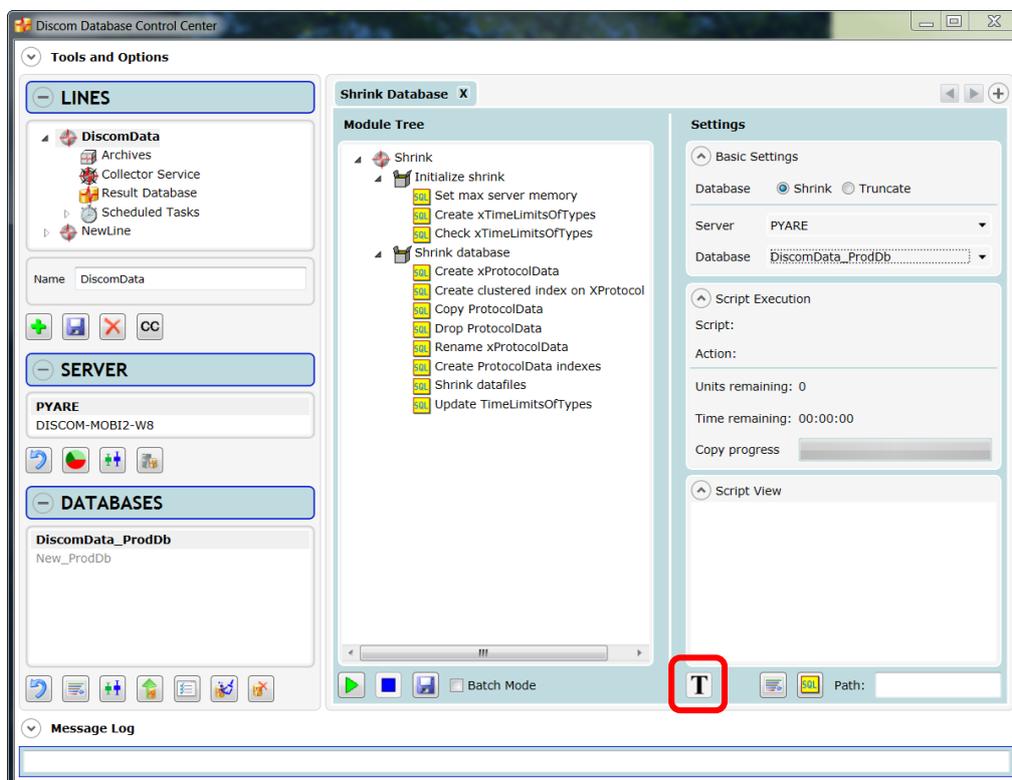


Fig. 38: Shrink Database.

In a first step the units to be kept along with their time limits will be copied to a temporary table (xTimeLimitsOfTypes). This table will be checked for corresponding single values, i.e. whether single

values' data exist that correspond to the units from the cleanup settings. If this is not the case, then a warning message will pop up and inform that no single values will be copied to the new datafile and that all existing single values will be dropped. If single values to be copied exist, then the shrink operation will continue without prompting for a user input.

### Deleting all data from a database

In case all data is to be deleted from a database, then choose the "Truncate" option of the "Shrink database" view (Fig. 38). This will remove all data from the databases' tables without deleting the tables itself, i.e. effectively preserving the database structure. Such an approach may be required if for some reason a new database cannot be created. After choosing the "Truncate" option and selecting a server and database, click the "T" button (red frame in Fig. 38) to activate the truncate operation.

## 3.5 Task Scheduler

As mentioned in the previous section, the deletion of measurements from the database using the stepwise procedure can be carried out on a regular scheduled using the Windows Task Scheduler. DbControl provides the functionality to set up these Windows Scheduler tasks, not only for the regular database cleanup but also for the management of DISCOM measurement files (Fig. 39).

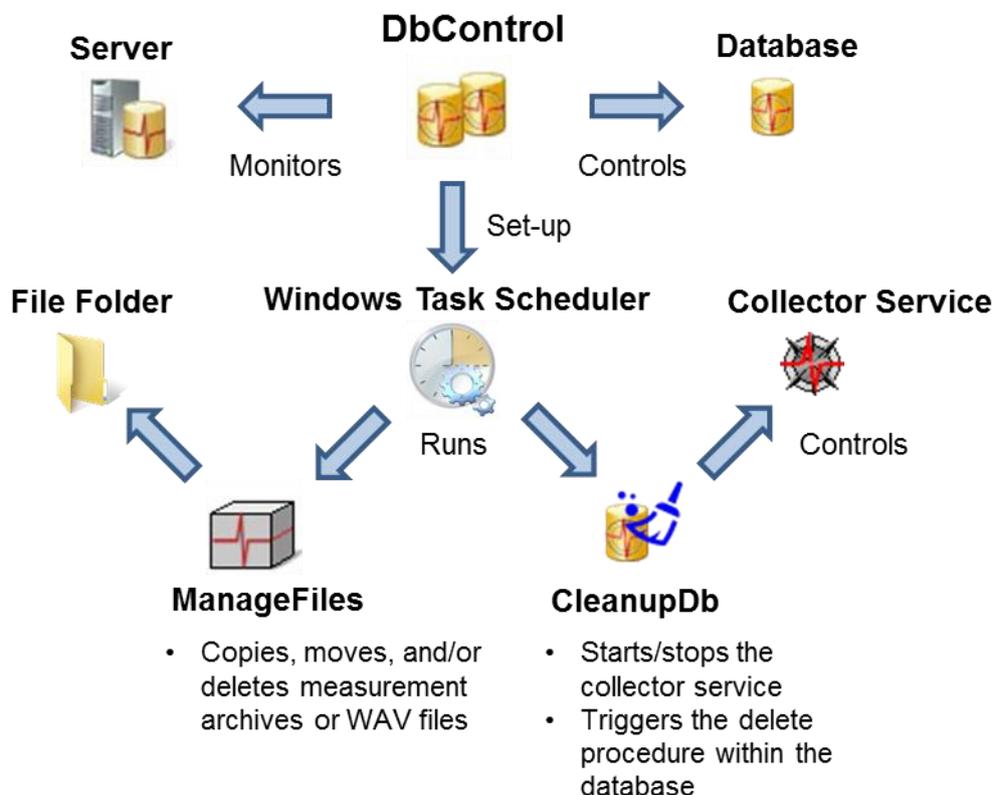


Fig. 39: Schematic program layout. The database cleanup and the management of DISCOM measurement files are carried out using the Windows Task Scheduler.

Go to "Task Scheduler" in the "Tools and Options" menu (Fig. 40).

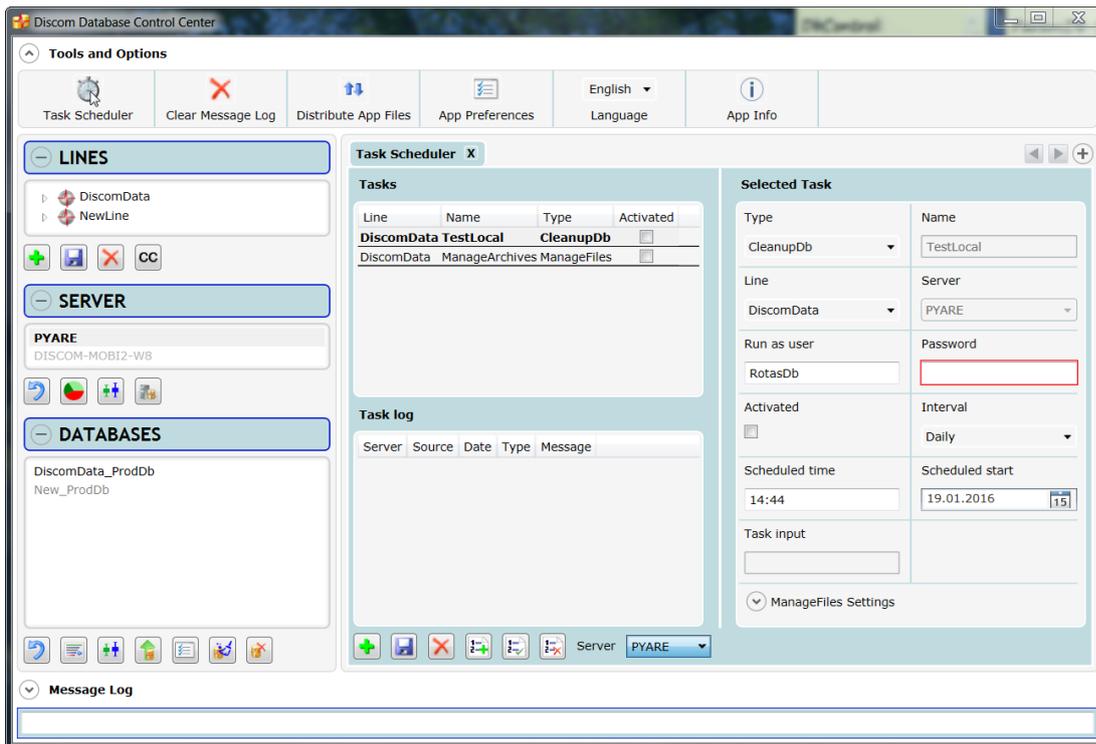


Fig. 40: Task Scheduler.

**Note:** First and foremost, in order to gain access to the Windows Task Scheduler and read and write tasks as well as to create the “Discom” log in the Windows Event Viewer, DbControl needs to be run as administrator (unless the user that is logged in has admin rights).

The “Task Scheduler” view lists all scheduled tasks contained in the “Lines” tree view model (and which could be accessed) and their basic properties, whereas the “Selected Task” (on the right) displays all relevant properties (Fig. 40). The “TaskLog” view lists all event viewer messages generated by the programs CleanupDb and ManageFiles. In order to set-up a new task, click on the “Create new task” button (“plus”-button) and fill in the fields in the “Selected Task” view. These include:

- **Type:** Drop-down list of the two task type “CleanupDb” and “ManageFiles”.
- **Name:** Name of the task. A task’s name is its unique identifier (i.e. duplicate tasks with the same name cannot be set-up) therefore this field only allows unique task names. If there is already a task with the specified name, the field will be show a red frame and trying to save the task will generate an error.
- **Line:** Drop-down list of all the lines contained in the “Lines” tree view model. Select the line that the task should belong to. **Note:** For the “CleanupDb” task, the line name is used to extract the database and SQL Server connection info.
- **Server:** Server location where the task is to be set-up. Drop-down list of all servers contained in the “Lines” tree view model (FactoryData.xml). **Note:** If the task is set-up on a remote server, make sure that the required program and configuration files are copied to the remote server (see section 2.1).
- **Run as user:** The user under which the task is registered. **Note:** The user needs to have the rights to start/stop the Collector Service as well as to login onto a remote server in case the

task is to be set-up on a remote server. Ideally the user has admin rights on all machines where tasks might be set-up. If the user is a domain user, specify the user as domain\user. In case of setting up a “CleanupDb” task and the SQL Server connection info makes use of the integrated security (i.e. Windows authentication), then the task needs to run as the user who is logged in at the time that the task is triggered.

- **Password:** Fill in the password of the user specified.
- **Activated:** Check this box to enable the task. If unchecked the task is disabled.
- **Interval:** Drop-down list of the three interval options: Daily, weekly, or monthly.
- **Scheduled Time:** Time of day at which the task is executed.
- **Scheduled Start:** Start date of the task execution. By default, this is the server’s current time.
- **Task Input:** Specify the configuration file used by the “ManageFiles” program. By default, the file is located in C:\Discom\Analysis\ProdDb\DbControl\ManageFiles. The storage location can be changed in the path tab of the app preferences (“Tools and Options”). If the specified file cannot be located in the given path, it is created new.

If the task to be created is a “ManageFiles” task, expand the “ManageFiles Settings” in the “Selected Task” view and set-up the action that is to be carried out by executing the “ManageFiles” task (Fig. 41). If the configuration file exists and already specifies actions, then click the “load” button (red frame in Fig. 41). If no actions have been defined yet, click on the “create” button (green “plus” button next to the “load” button). After filling in the required fields for the action, make sure to click the “save” button. **Note:** Multiple actions can be defined for a single “ManageFiles” task.

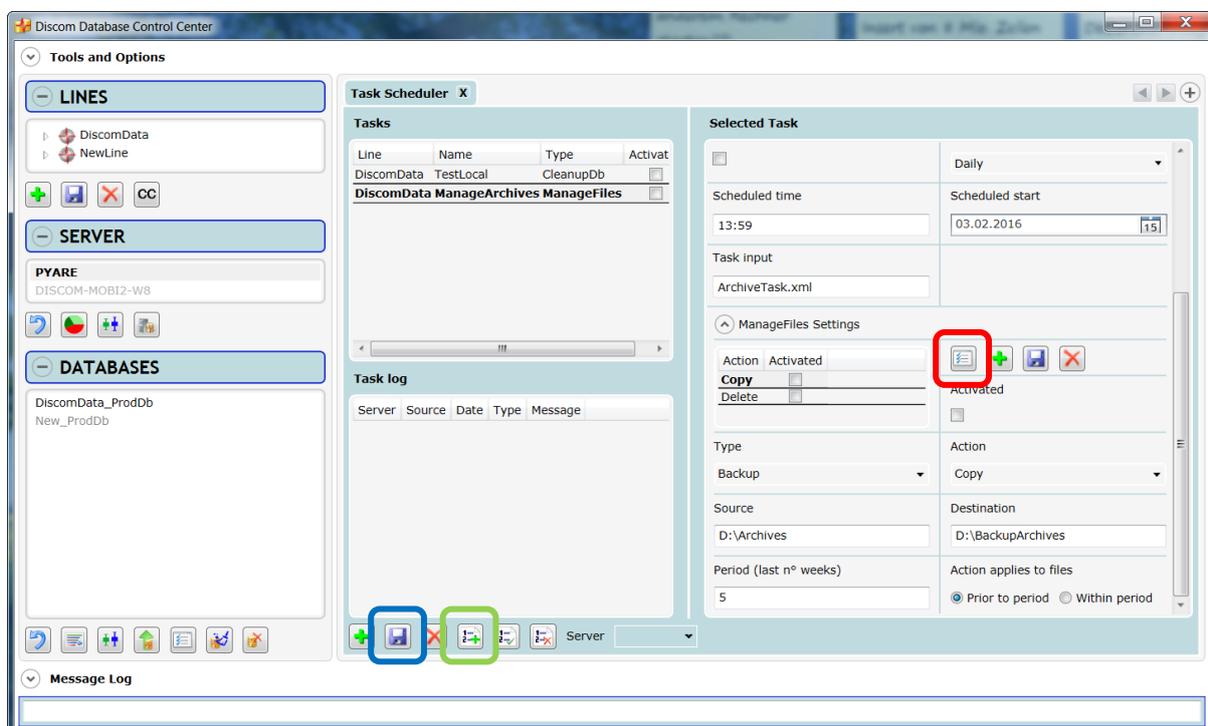


Fig. 41: Task Scheduler: ManageFiles Settings

The fields of the “ManageFiles Settings” include:

- **Activated:** Check to enable the action, uncheck to disable the action.
- **Type:** Drop-down list of the two basic “ManageFiles” types: “Backup” and “Deletion”

- **Action:** Drop-down list of the actions associated with the selected type. “Backup” allows either a copy or a move action. “Deletion” only allows a delete action.
- **Source:** Specify the source folder of the DISCOM measurement files (archives, WAV). **Note:** The files stored within the source folder have to be stored in sub folders which specify the week of the year as folder name (e.g. KW 12). Else the “ManageFiles” program is unable to run its actions.
- **Destination:** Specify the destination folder. Only applicable for backup actions.
- **Period (last n° weeks):** Specify the period that the action relates to.
- **Action applies to files:** Two options: If the option “Prior to period” is selected, the specified action applies to all files which are contained in the sub folders with dates prior to the specified period (last n° weeks). If the option “Within period” is selected, the specified action applies to all files which are contained in the sub folders with dates within the specified period (last n° weeks).

After all the required fields of the “Selected Task” view are filled in, click on “save task” (blue frame in Fig. 41). A message is displayed in the status bar informing about whether the task could be saved or not. If the task could not be saved, check the message log for more details on the errors. After the task was successfully created, make sure to register the “Discom” event log in the Application and Services Logs of the server’s Event Viewer.

In order to do so, select the server from the drop-down check list where the task is set-up and click on the “register” button (green frame in Fig. 41). This will register the event log for both the “CleanupDb” and the “ManageFiles” event sources. To view the tasks’ log, select the server whose logs are to be displayed (drop-down check list) and click the second last button in the button panel. To clear the tasks’ log, click the last button in the button panel.

**Note: As already mentioned, in order for the tasks to work properly, a number of requirements have to be met. These are:**

For tasks that are set-up on the local (central) server but need to access files and/or the collector service on a remote server:

- Allow “Remote Service Management” through firewall of the target server.
- The source and destination directories of the DISCOM measurement files are shared on the network with read-/write-access for the task user.
- In case of a “ManageFiles” task, the files are copied and/or moved via the local (central) server, which takes up resources of the local server.
- Task user needs rights to start/stop the service on the remote server (only in case of “CleanupDb” tasks).

For tasks that are set-up on a remote server:

- Allow “Remote Scheduled Task Management” through firewall of the target server.
- Task user needs login rights to the remote server.
- DbControl user needs rights to create tasks on the remote server.

- The DbControl configuration directory and the source and destination directories of the DISCOM measurement files are shared on the network with read-/write-access for the task user.
- The “CleanupDb” and “ManageFiles” program and configuration files are copied to the remote server (see section 2.1).
- In case the paths to the program and configuration files on the remote server differ from the default paths (see section 3.6), adjust (and save) the paths in the app preferences before saving the task. **Note:** Make sure to change the paths back to the default values before adding tasks on the local and/or remote servers which use the default paths.

**Note:** In case the task to be created is a “CleanupDb” task then the triggered run of the background program “CleanupDb” includes three steps (compare Fig. 39): Stopping the collector service, triggering the deletion procedure within the database, and starting the collector service. If the stopping/starting of the collector service should be excluded for some reason (e.g. cluster collector), then either remove the respective collector service entry from the “Lines” tree view or add the argument “DeleteOnly” to the actions of the “CleanupDb” task (see Fig. 42).

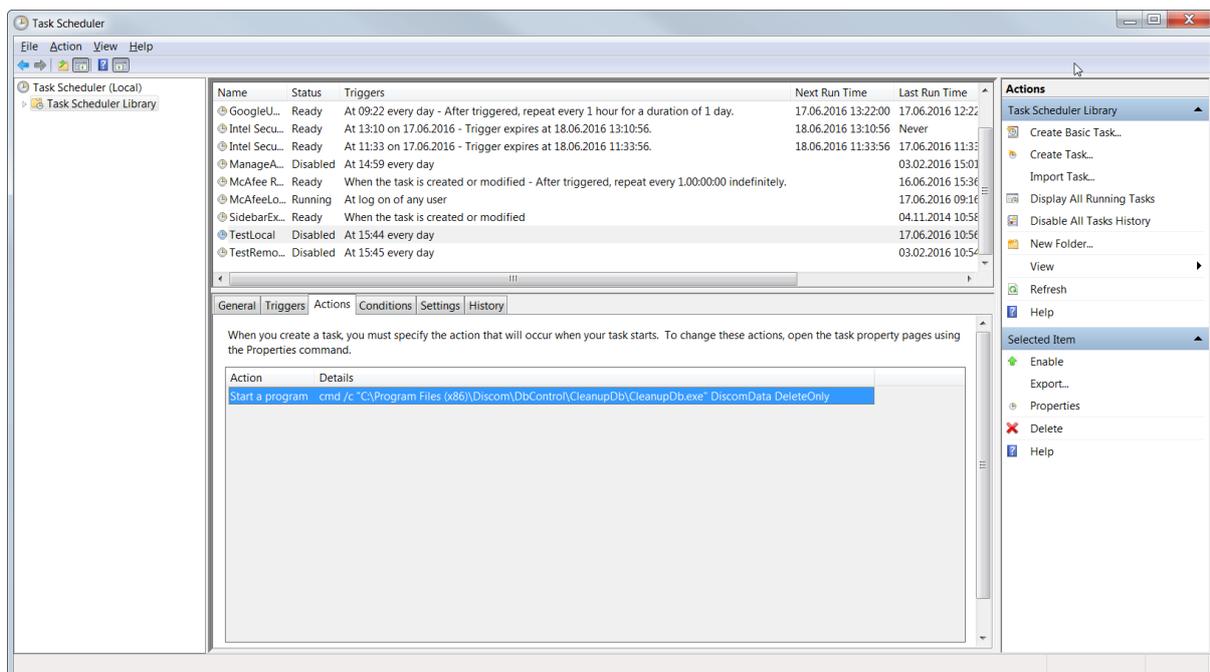


Fig. 42: CleanupDb task. Additional argument “DeleteOnly” triggers only the delete procedure within the database.

### 3.6 Additional tools and options

In addition to the “Task Scheduler” tool, there are a couple of other tools and functions available in the “Tools and Options” menu (Fig. 43):

- **Clear Message Log:** This option clears the message log.
- **Distribute App Files:** This option opens the “Distribute App Files” tab view (see section 2.3).
- **App Preferences:** This option opens the “App Preferences” tab view (see below).
- **Language:** Two interface language options are available: English and German

- **App Info:** This option displays general app info (e.g. version number and build date).

DbControl requires that a couple of paths are specified (e.g. the paths to the program and configuration files). These are listed in the “DbControlSettings” (see section 1.2.2). In case the DbControlSettings.xml is missing or cannot be read, DbControl applies default values for these paths (Fig. 43).

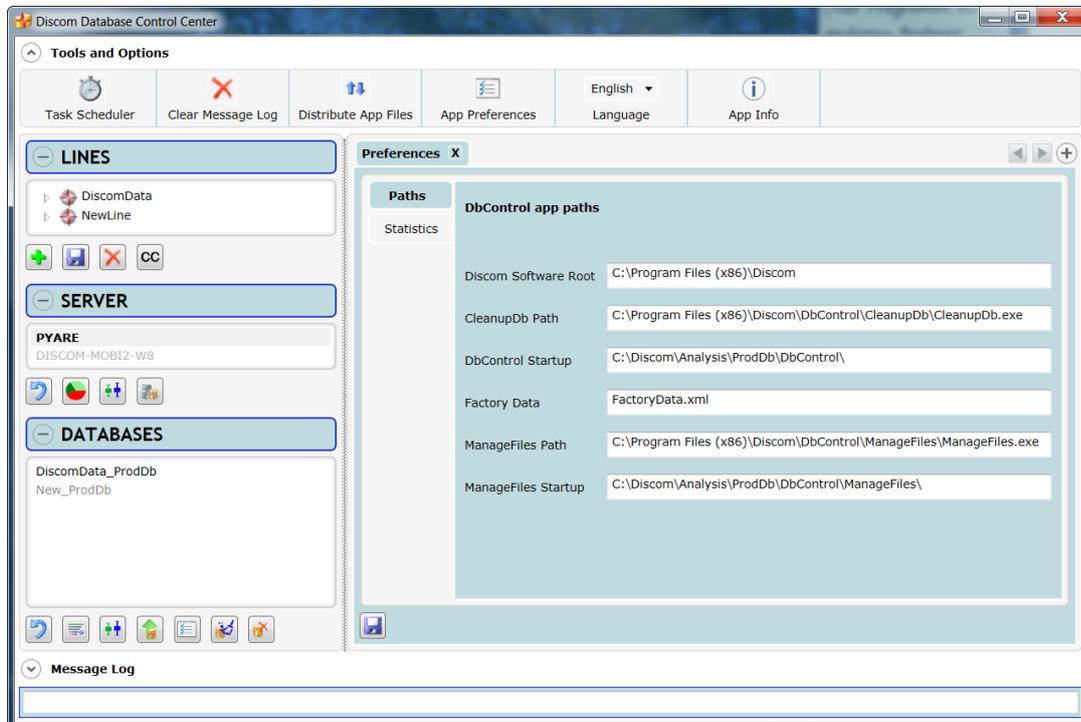


Fig. 43: App Preferences: DbControl default paths.

## A. Supplement: Structure and functions of the “CleanupDb” database tables and procedures

As part of SQL script collection of the tools “Create/Update Database”, the following tables and procedures within the database will be setup:

- Delete Log: Table which stores information about each single delete process.
- Delete Parameters: Table which stores the parameters that control the delete process.
- DeleteData: Stored procedure which implements the actual delete routine.
- TimeLimitsOfTypes: Table which stores information about the number of units to delete and keep. These can be set independent of or dependent on type.
- TimeLimitsRenew: Stored procedure which needs to be executed in case of changed delete parameters.
- DeleteData\_TypeAndBench: Stored procedure which implements the actual delete routine for the case where the deletion of single values is type and bench specific.
- TimeLimitsOfTypes\_TypeAndBench: Table which stores information about the number of units to delete and keep. These can be set independent of or dependent on type and bench.
- TimeLimitsRenew\_TypeAndBench: Stored procedure which needs to be executed in case of changed delete parameters (deletion of single values is type and bench specific).

### a. DeleteLog

The table DeleteLog contains the following columns:

- Id: [int] Incremental identifier.
- Day: [datetime] Date and time of execution of single delete run.
- TimeLimit: [datetime] Time limit applied in the single delete run.
- Duration: [time] Duration of single delete run.
- Type: [int] Type of units which were deleted.
- Bench: [int] Bench of units which were deleted.
- DeletedMain: [int] Number of units deleted from table MainData.
- DeletedProtocol: [int] Number of units deleted from table ProtocolData.
- ErrorId: [int] SQL-internal error code in case of an error associated with the respective DELETE statements.

Tab. 1: Table DeleteLog.

Id	Day	TimeLimit	Duration	Type	DeletedMain	DeletedProtocol	ErrorId
1	19.04.2015 18:55	19.07.2012 18:54	00:00:24.5	0	2000	1033395	NULL
2	19.04.2015 19:16	19.07.2012 18:54	00:00:22.1	0	2000	1010591	NULL
3	19.04.2015 19:37	19.07.2012 18:54	00:00:14.5	0	2000	992187	NULL
4	19.04.2015 20:29	19.07.2012 18:54	00:00:12.7	0	2000	1012815	NULL
5	19.04.2015 21:05	19.07.2012 18:54	00:00:12.4	0	2000	1010654	NULL
6	19.04.2015 21:05	19.07.2012 18:54	00:00:03.9	0	254	130248	NULL
7	19.04.2015 21:05	19.08.2012 18:54	00:00:14.6	6	1746	895633	NULL

## b. DeleteParameters

CleanupDb was scripted to enable automatic deletion of units from MainData and its associated data tables (ArchiveData, AdditionalInfo, ErrorData, ModesPerMeasurement, ProtocolData) **AND/OR** from ProtocolData in order to accommodate the deletion of single values while allowing the analysis of basic long-term production statistics.

The table DeleteParameters contains the following columns:

- Id [int] Incremental identifier.
- UnitsToDelete [int] Number of units to delete from MainData (type-dependent or independent). Note: This is the only parameter that does not require the recalculation of the time limits (TimeLimitsOfTypes).
- CaseProtocol [int] Parameter that controls how units are deleted from ProtocolData.
- TimeLimitProtocol [int] Time limit (number of past months) for the deletion from ProtocolData.
- UnitsToKeep [int] Number of units to keep (type-dependent or independent).
- CaseMain [int] Parameter that controls whether units are deleted from MainData and its associated data (ArchiveData, AdditionalInfo, ErrorData, ModesPerMeasurement).
- TimeLimitMain [int] Time limit (number of past months) for the deletion from MainData and its associated data.
- RenewLimits [tinyint] Parameter that controls whether the time limits need to be recalculated (TimeLimitsOfTypes). 0 = no recalculation, 1 = recalculation  
Note: This parameter is set automatically by the delete routine and should not be changed manually.

Tab. 2: Table DeleteParameters.

Id	UnitsToDelete	CaseProtocol	TimeLimitProtocol	UnitsToKeep	CaseMain	TimeLimitMain	RenewLimits
1	2000	4	32	2500	1	33	0

If the parameter CaseMain is set to 1, all units are deleted from MainData and its associated data tables which were measured prior to the number of months specified by the parameter TimeLimitMain. If the parameter CaseMain is set to 0, no units are deleted from MainData and its associated data tables. The parameter CaseProtocol controls if and how units are deleted from ProtocolData. **Note: The deletion from Protocol Data can be type-dependent or type- and bench-dependent.**

- CaseProtocol = 0: Delete routine is inactive.
- CaseProtocol = 1: Only the most recent units (*UnitsToKeep*) per type (or per type and bench) measured within the past months (*TimeLimitProtocol*) are kept.
- CaseProtocol = 2: The most recent units per type (or per type and bench) are kept, in addition to all units measured within the past months.
- CaseProtocol = 3: Only the most recent units per type (or per type and bench) are kept.
- CaseProtocol = 4: All units per type (or per type and bench) that were measured within the past months are kept.

### c. TimeLimitsOfTypes

The table TimeLimitsOfTypes contains the following columns:

- Id [int] Type-Id.
- Name [nvarchar] Type-Name.
- BenchId [int] Bench-Id.
- BenchName [nvarchar] Bench-Name.
- ActualTimeLimit [datetime] Type (or type- and bench) specific time limit.
- PreviousTimeLimit [datetime] Type (or type- and bench) specific time limit.
- UnitsToDelete [int] Number of total units per type (or type- and bench) to be deleted.
- UnitsRetained [int] Number of total units per type (or type- and bench) to be retained.
- Deleted [tinyint] Parameter that stores the info whether there are units left to be deleted (= 0) or not (=1).

The table TimeLimitsOfTypes is crucial for the delete routine insofar as it controls the number of units per type (or per type and bench) to be deleted and the respective time limits in-between those units have to lie. If parameter CaseMain in table DeleteParameters is set to 1, the entry with Id = 0 in TimeLimitsOfTypes represents a type-independent deletion (i.e. all units irrespective of type and measured prior to the ActualTimeLimit are deleted). If the parameter CaseProt is set to a value between 1–4, entries for each type are added to the table TimeLimitsOfTypes.

Tab. 3: Table TimeLimitsOfTypes.

Id	Name	Bench-Id	Bench-Name	ActualTimeLimit	PreviousTimeLimit	UnitsTo-Delete	Units-Retained	Deleted
2	3500	1	P2	2013-08-02 12:00:43.017	2012-03-20 14:48:35.000	14343	4173	0
2	3500	2	P4	2013-08-02 12:00:43.017	2012-03-20 21:50:28.000	12879	3702	0
2	3500	3	P3	2013-08-02 12:00:43.017	2012-03-29 22:19:57.000	13097	3859	0
2	3500	4	P1	2013-08-02 12:00:43.017	2012-06-28 09:16:08.000	14281	4363	0

### d. Delete process

Each delete run (started by the CleanupDb task set-up in the Windows Task Scheduler or within DbControl), executes the stored procedure DeleteData (or DeleteData\_TypeAndBench). This procedure has one input parameter (i.e. the number of units to delete per run) taken from the column UnitsToDelete of the table DeleteParameters. Based on the time limits of types, the delete routine always starts deleting from the entry with the lowest Id. In case the UnitsToDelete are less than the UnitsToKeep, the remaining number (UnitsToKeep – UnitsToDelete) is deleted from the entry with the next higher Id during the same delete run. If there are no units left to delete (UnitsToDelete sum to 0), the parameter RenewLimits in table DeleteParameters is set to 1, which in turn triggers the re-calculation of the time limits (by executing the stored procedure TimeLimitsRenew or TimeLimitsRenew\_TypeAndBench).

## B. Supplement: Troubleshooting

### Server statistics view:

The open-source package OxyPlot implemented in DbControl as part of the server statistics' views might cause DbControl to crash in case the .NET 4.0 security update KB2604121 is not installed. Solution: Install the update (<https://support.microsoft.com/en-us/kb/2604121>).

### Task Scheduler:

When setting up a new task via the Task Scheduler, the following error might occur upon trying to register the task: "A specified logon session does not exist. It may already have been terminated." This happens when the storage of passwords and credentials for network authentication is disabled. Solution: Run SECPOL.MSC and go to Security Settings/Local Policies/Security Options. Disable the network access option "Do not allow storage of passwords and credentials for network authentication" (<https://blogs.technet.microsoft.com/askperf/2012/04/18/task-scheduler-error-a-specified-logon-session-does-not-exist/>).

## C. Supplement: Application source code info

DbControl uses the following excellent open source tools:

- OxyPlot (<http://oxyplot.org/>)
- Xceed Extended WPF Toolkit (<http://wpftoolkit.codeplex.com/>)
- Task Scheduler Managed Wrapper (<https://taskscheduler.codeplex.com/>)
- A number of open-source customized WPF user controls published under the "Code Project Open License (CPO) 1.02"

### Extended WPF Toolkit license

#### Microsoft Public License (MS-PL)

This license governs use of the accompanying software. If you use the software, you accept this license. If you do not accept the license, do not use the software.

#### 1. Definitions

The terms "reproduce," "reproduction," "derivative works," and "distribution" have the same meaning here as under U.S. copyright law.

A "contribution" is the original software, or any additions or changes to the software.

A "contributor" is any person that distributes its contribution under this license.

"Licensed patents" are a contributor's patent claims that read directly on its contribution.

#### 2. Grant of Rights

(A) Copyright Grant- Subject to the terms of this license, including the license conditions and limitations in section 3, each contributor grants you a non-exclusive, worldwide, royalty-free copyright license to reproduce its contribution, prepare derivative works of its contribution, and distribute its contribution or any derivative works that you create.

(B) Patent Grant- Subject to the terms of this license, including the license conditions and limitations in section 3, each contributor grants you a non-exclusive, worldwide, royalty-free license under its licensed patents to make, have made, use, sell, offer for sale, import, and/or otherwise dispose of its contribution in the software or derivative works of the contribution in the software.

#### 3. Conditions and Limitations

(A) No Trademark License- This license does not grant you rights to use any contributors' name, logo, or trademarks.

(B) If you bring a patent claim against any contributor over patents that you claim are infringed by the software, your patent license from such contributor to the software ends automatically.

(C) If you distribute any portion of the software, you must retain all copyright, patent, trademark, and attribution notices that are present in the software.

(D) If you distribute any portion of the software in source code form, you may do so only under this license by including a complete copy of this license with your distribution. If you distribute any portion of the software in compiled or object code form, you may only do so under a license that complies with this license.

(E) The software is licensed "as-is." You bear the risk of using it. The contributors give no express warranties, guarantees or conditions. You may have additional consumer rights under your local laws which this license cannot change. To the extent permitted under your local laws, the contributors exclude the implied warranties of merchantability, fitness for a particular purpose and non-infringement.

### OxyPlot and Task Scheduler Managed Wrapper license

The MIT License (MIT)

Copyright (c) 2014 OxyPlot contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### The Code Project Open License (CPOL) 1.02

This License governs Your use of the Work. This License is intended to allow developers to use the Source Code and Executable Files provided as part of the Work in any application in any form.

The main points subject to the terms of the License are:

Source Code and Executable Files can be used in commercial applications;

Source Code and Executable Files can be redistributed; and

Source Code can be modified to create derivative works.

No claim of suitability, guarantee, or any warranty whatsoever is provided. The software is provided "as-is". The Article(s) accompanying the Work may not be distributed or republished without the Author's consent. This License is entered between You, the individual or other entity reading or otherwise making use of the Work licensed pursuant to this License and the individual or other entity which offers the Work under the terms of this License ("Author").

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CODE PROJECT OPEN LICENSE ("LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HEREIN, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE AUTHOR GRANTS YOU THE RIGHTS CONTAINED HEREIN IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS. IF YOU DO NOT AGREE TO ACCEPT AND BE BOUND BY THE TERMS OF THIS LICENSE, YOU CANNOT MAKE ANY USE OF THE WORK.

Definitions.

"Articles" means, collectively, all articles written by Author which describes how the Source Code and Executable Files for the Work may be used by a user.

"Author" means the individual or entity that offers the Work under the terms of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works.

"Executable Files" refer to the executables, binary files, configuration and any required data files included in the Work.

"Publisher" means the provider of the website, magazine, CD-ROM, DVD or other medium from or by which the Work is obtained by You.

"Source Code" refers to the collection of source code and configuration files used to create the Executable Files.

"Standard Version" refers to such a Work if it has not been modified, or has been modified in accordance with the consent of the Author, such consent being in the full discretion of the Author.

"Work" refers to the collection of files distributed by the Publisher, including the Source Code, Executable Files, binaries, data files, documentation, whitepapers and the Articles.

"You" is you, an individual or entity wishing to use the Work and exercise your rights under this License.

Fair Use/Fair Use Rights. Nothing in this License is intended to reduce, limit, or restrict any rights arising from fair use, fair dealing, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

License Grant. Subject to the terms and conditions of this License, the Author hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

You may use the standard version of the Source Code or Executable Files in Your own applications. You may apply bug fixes, portability fixes and other modifications obtained from the Public Domain or from the Author. A Work modified in such a way shall still be considered the standard version and will be subject to this License.

You may otherwise modify Your copy of this Work (excluding the Articles) in any way to create a Derivative Work, provided that You insert a prominent notice in each changed file stating how, when and where You changed that file.

You may distribute the standard version of the Executable Files and Source Code or Derivative Work in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution.

The Articles discussing the Work published in any form by the author may not be distributed or republished without the Author's consent. The author retains copyright to any such Articles. You may use the Executable Files and Source Code pursuant to this License but you may not repost or republish or otherwise distribute or make available the Articles, without the prior written consent of the Author.

Any subroutines or modules supplied by You and linked into the Source Code or Executable Files of this Work shall not be considered part of this Work and will not be subject to the terms of this License.

Patent License. Subject to the terms and conditions of this License, each Author hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, import, and otherwise transfer the Work.

Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You agree not to remove any of the original copyright, patent, trademark, and attribution notices and associated disclaimers that may appear in the Source Code or Executable Files.

You agree not to advertise or in any way imply that this Work is a product of Your own.

The name of the Author may not be used to endorse or promote products derived from the Work without the prior written consent of the Author.

You agree not to sell, lease, or rent any part of the Work. This does not restrict you from including the Work or any part of the Work inside a larger software distribution that itself is being sold. The Work by itself, though, cannot be sold, leased or rented.

You may distribute the Executable Files and Source Code only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy of the Executable Files or Source Code You distribute and ensure that anyone receiving such Executable Files and Source Code agrees that the terms of this License apply to such Executable Files and/or Source Code. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute the Executable Files or Source Code with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License.

You agree not to use the Work for illegal, immoral or improper purposes, or on pages containing illegal, immoral or improper material. The Work is subject to applicable export laws. You agree to comply with all such laws and regulations that may apply to the Work after Your receipt of the Work.

Representations, Warranties and Disclaimer. THIS WORK IS PROVIDED "AS IS", "WHERE IS" AND "AS AVAILABLE", WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS OR GUARANTEES.

YOU, THE USER, ASSUME ALL RISK IN ITS USE, INCLUDING COPYRIGHT INFRINGEMENT, PATENT INFRINGEMENT, SUITABILITY, ETC. AUTHOR EXPRESSLY DISCLAIMS ALL EXPRESS, IMPLIED OR STATUTORY WARRANTIES OR CONDITIONS, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF MERCHANTABILITY, MERCHANTABLE QUALITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY WARRANTY OF TITLE OR NON-INFRINGEMENT, OR THAT THE WORK (OR ANY PORTION THEREOF) IS CORRECT, USEFUL, BUG-FREE OR FREE OF VIRUSES. YOU MUST PASS THIS DISCLAIMER ON WHENEVER YOU DISTRIBUTE THE WORK OR DERIVATIVE WORKS.

Indemnity. You agree to defend, indemnify and hold harmless the Author and the Publisher from and against any claims, suits, losses, damages, liabilities, costs, and expenses (including reasonable legal or attorneys' fees) resulting from or relating to any use of the Work by You.

Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL THE AUTHOR OR THE PUBLISHER BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK OR OTHERWISE, EVEN IF THE AUTHOR OR THE PUBLISHER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Termination.

This License and the rights granted hereunder will terminate automatically upon any breach by You of any term of this License. Individuals or entities who have received Derivative Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 6, 7, 8, 9, 10 and 11 will survive any termination of this License.

If You bring a copyright, trademark, patent or any other infringement claim against any contributor over infringements You claim are made by the Work, your License from such contributor to the Work ends automatically.

Subject to the above terms and conditions, this License is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, the Author reserves the right to release the Work under different license terms or to stop distributing the Work at any time;

provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

Publisher. The parties hereby confirm that the Publisher shall not, under any circumstances, be responsible for and shall not have any liability in respect of the subject matter of this License. The Publisher makes no warranty whatsoever in connection with the Work and shall not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. The Publisher reserves the right to cease making the Work available to You at any time without notice

#### Miscellaneous

This License shall be governed by the laws of the location of the head office of the Author or if the Author is an individual, the laws of location of the principal place of residence of the Author.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this License, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed herein. There are no understandings, agreements or representations with respect to the Work not specified herein. The Author shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Author and You.