



# Users Guide

## Assentis:DocAdmin Console

---

# Users Guide: Assentis:DocAdmin Console

Version 6.7.0

Copyright © 2020 Assentis Technologies AG, Blegistr. 1, 6343 Rotkreuz, Switzerland.

## Abstract

This document explains the usage of DocAdmin Console.

## Legal Notice

THIS DOCUMENTATION IS PROVIDED FOR REFERENCE PURPOSES ONLY AND COPYRIGHT PROTECTED (C) BY ASSENTIS TECHNOLOGIES. COPYRIGHTS FOR COMPONENTS OF THIS WORK OWNED BY OTHERS THAN ASSENTIS TECHNOLOGIES MUST BE HONORED. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS DOCUMENTATION, THIS DOCUMENTATION IS PROVIDED "AS IS" WITHOUT ANY WARRANTY WHATSOEVER AND TO THE MAXIMUM EXTENT PERMITTED, ASSENTIS TECHNOLOGIES DISCLAIMS ALL IMPLIED WARRANTIES, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SAME. ASSENTIS TECHNOLOGIES SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION, DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES, ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS DOCUMENTATION OR ANY OTHER DOCUMENTATION. NOTWITHSTANDING ANYTHING TO THE CONTRARY, NOTHING CONTAINED IN THIS DOCUMENTATION OR ANY OTHER DOCUMENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM ASSENTIS TECHNOLOGIES (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT GOVERNING THE USE OF THIS SOFTWARE.

---

---

# Table of Contents

1. Introduction .....	1
1.1. Prerequisites .....	1
1.2. Audience .....	1
2. General principles .....	2
2.1. Important terms .....	3
3. Technical overview .....	5
4. The graphical user interface .....	6
4.1. General application elements .....	6
4.1.1. Login page .....	6
4.1.2. Application page .....	7
4.2. Job Browser .....	10
4.2.1. Jobs .....	10
4.2.2. Errors .....	13
4.2.3. Raw Data .....	15
4.2.4. Documents .....	17
4.3. Server Monitor .....	19
4.3.1. View details .....	20
4.4. Start Job .....	21
4.5. Render Test .....	22
4.6. Job Scheduler .....	23
4.7. PROM Import .....	26
5. Attachment Management .....	29
5.1. Configuration .....	29
5.1.1. Attachment Meta-Information .....	30
5.1.2. Channels .....	33
5.2. Documents .....	34
5.3. Attachments .....	36
5.4. Attachment Instruction .....	40
5.5. Attachment Campaigns .....	43

---

## List of Figures

2.1. DocAdmin Console principles .....	3
3.1. DocBase processing - Overview .....	5
4.1. Login Page - Overview .....	6
4.2. General elements - Task bar .....	7
4.3. General elements - Table .....	7
4.4. General elements - Selection .....	7
4.5. General elements - Table operations bar .....	8
4.6. General elements - Actions .....	8
4.7. General elements - Filters .....	9
4.8. General elements - Parameters .....	9
4.9. General elements - Result .....	9
4.10. General elements - Navigation .....	10
4.11. Job Browser - Jobs - Overview .....	10
4.12. Job Browser - DocBase Job Details .....	11
4.13. Job Browser - DocBase Job Details - Postprocessing Details .....	12
4.14. Job Browser - Jobs - Filter .....	12
4.15. Job Browser - Errors - Overview .....	14
4.16. Job Browser - DocBase Error Details .....	14
4.17. Job Browser - Error - Filter .....	15
4.18. Job Browser - Raw Data - Overview .....	15
4.19. Job Browser - DocBase Raw Data Details .....	16
4.20. Job Browser - Raw Data - Filter .....	16
4.21. Job Browser - Documents - Overview .....	17
4.22. Job Browser - DocBase Document Details .....	18
4.23. Job Browser - Document - Filter .....	19
4.24. Server Monitor - Overview .....	19
4.25. Server Monitor - Server Details .....	20
4.26. Server Monitor - Server Details - JVM Threads .....	21
4.27. Start DocBase Jobcard - Overview .....	22
4.28. DocBase Render Test - Overview .....	23
4.29. DocBase Job Scheduler - Scheduled Job Details .....	25
4.30. DocBase Job Scheduler - Overview .....	25
4.31. Promotion File Import Details .....	27
4.32. Scheduled Promotion File Import - Overview .....	27
5.1. Configuration - Attachment Meta-Information overview .....	30
5.2. Configuration - Create Attachment Meta-Information .....	30
5.3. Configuration - Channels overview .....	33
5.4. Configuration - Create Channel .....	34
5.5. Documents - Overview .....	34
5.6. Documents - Create Document Reference .....	35
5.7. Attachments overview .....	36
5.8. Create Attachments .....	37
5.9. Create Attachment Version - Main .....	38
5.10. Create Attachment Version - Override Attachment .....	39
5.11. Attachment Instructions overview .....	40
5.12. Modify Attachment Instruction - Main .....	41
5.13. Modify Attachment Instruction - Override Attachment (version) .....	43
5.14. Attachment Campaigns overview .....	44
5.15. Create Attachment Campaign - Main .....	44
5.16. Table of Attachment Instructions .....	45
5.17. Assign Attachment Instruction to Attachment Campaign .....	45
5.18. Validate Attachment Campaign .....	46

---

# Chapter

# 1

## Introduction

DocAdmin Console is a web application within DocFamily to monitor the overall setup and allows certain interventions in case of problems. This document describes the usage of this application and the tasks which can be performed.

### 1.1. Prerequisites

For using DocAdmin Console a supported operating system with Java is required. The application needs database access and is deployed into an application server. Run the installation as described in the document "DocFamily Installation Guide". Furthermore a running instance of DocBase is required to follow this document. Its installation is also described in "DocFamily Installation Guide". See the document "DocFamily Compatibility" to find out which environment is compatible with DocAdmin Console.

### 1.2. Audience

The typical users of the DocAdmin Console and this document are IT operators which are responsible for a proper working DocFamily setup. To use DocAdmin Console deep technical knowledge of the concepts in DocBase is necessary. More information about concepts and functionality of DocBase can be found in the documents:

- DocBase Cookbook
- DocBase Tasks Reference
- DocBase Post Processing Guide
- DocBase Configuring Formats
- DocBase Commandline Client.

---

# Chapter

# 2

## General principles

DocAdmin Console monitors a “DocFamily environment”. Such an environment may include various server and client applications of DocFamily (see Figure 2.1, “DocAdmin Console principles”). At least DocBase and DocAdmin Console must use the same database schema to store their data that DocAdmin Console can provide its full range of functionality to monitor and manipulate data.

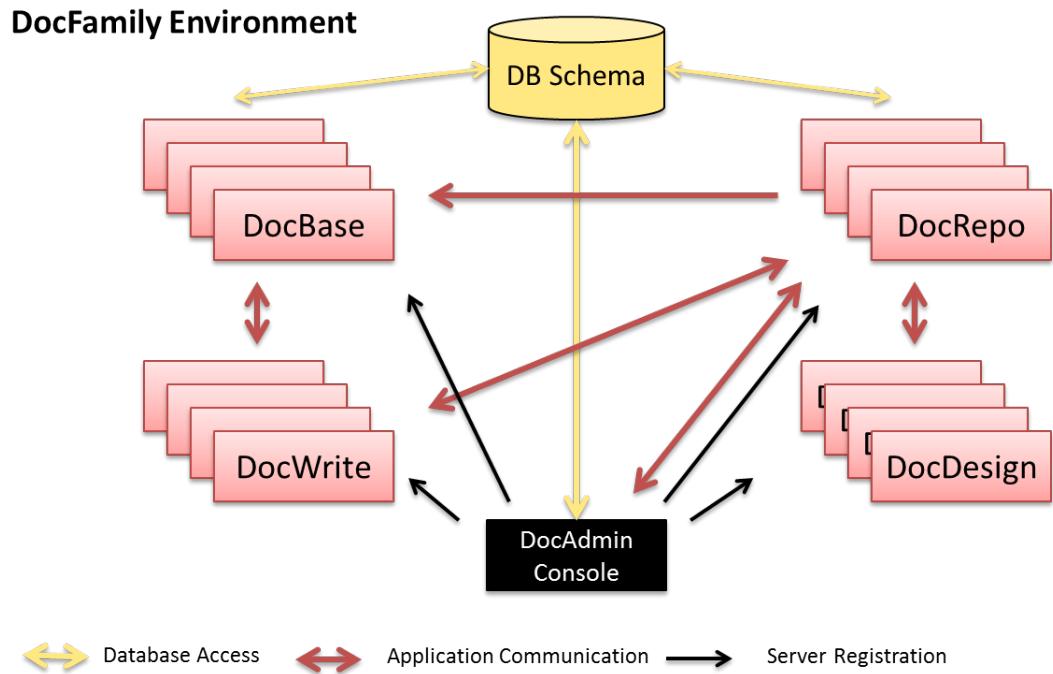
Within DocFamily different ways of retrieving and writing data to the database exist. The database stores resources and runtime data. In general resources are fetched via the DocRepo client, other data like runtime data are written and retrieved directly by the applications.

During startup of DocRepo on an empty database schema the initialization is done. It creates all tables that are needed by the various DocFamily applications. For DocAdmin Console tables are reserved to store its runtime data like promotion files or connection data of registered servers. DocBase uses separated tables to save the runtime data during document production. Another range of tables is exclusively used by DocRepo to store and organize resources like DocDesign Workspaces, deployment packages, Expressions, DocWrite Templates, images and test data. Also all relations between the resources are mapped in these tables. In general the tables are dedicated to a certain application and there exist only very few tables that are shared between all applications that use this database schema, e.g. the table “DBINFO”.

To get the correct and complete resources all DocFamily applications have to use the DocRepo client interface. E.g. the resource-creating and -manipulating applications like DocAdmin, DocDesign or DocWrite use this interface.

On the other hand DocBase writes its runtime data directly into its database tables even if each DocBase has defined a corresponding DocRepo that distributes the required resources via the client. DocAdmin Console also writes its data directly into the database, but also has defined a corresponding DocRepo that handles the authentication during login into DocAdmin Console. Since the functionality of DocAdmin Console is based on the runtime data of DocBase, it needs access to these data. This happens via direct read and write access to the database tables.

Figure 2.1. DocAdmin Console principles



The views and pages of DocAdmin Console are predefined and provide particular functionality. With the interface of the application it is possible to provided customized functionality. For more information see the public APIs of DocAdmin Console, DocRepo and DocBase. Also the documents "DocBase Commandline Client", "DocBase Post Processing Guide" and "DocBase Tasks Reference" are of interest.

## 2.1. Important terms

- **job:** A job is a process that is executed in DocBase.
- **job execution:** Depending on the defined cron expression a job will be executed regular. The run of a job at a certain time is called job execution.
- **promotion file:** This file contains resources that can be imported into DocRepo.
- **document:** In the Assentis context a document is information put into writing, whereas the writing is done electronically. This includes various types of information that can be stored in electronic systems, such as images, texts, charts and so on.
- **DocFamily environment:** A cluster of DocFamily applications that write into the same database schema or retrieve data from are called DocFamily environment.
- **database schema:** Various applications of Assentis DocFamily write their data into a database. Therefore a database schema of a dedicated structure is necessary. This structure is created during the first startup of DocRepo on an empty database schema. Even if it is possible to configure a separated database schema per application, it is recommended to use the same schema for all DocFamily applications of one DocFamily environment (see "DocFamily environment").
- **runtime data:** Runtime data are created during the execution of a program. They are consumed during this execution and will change their status while being processed. Even if runtime data are stored, they are time dependent and expire. Storing of such data is meant for safeguarding and recovery needs.

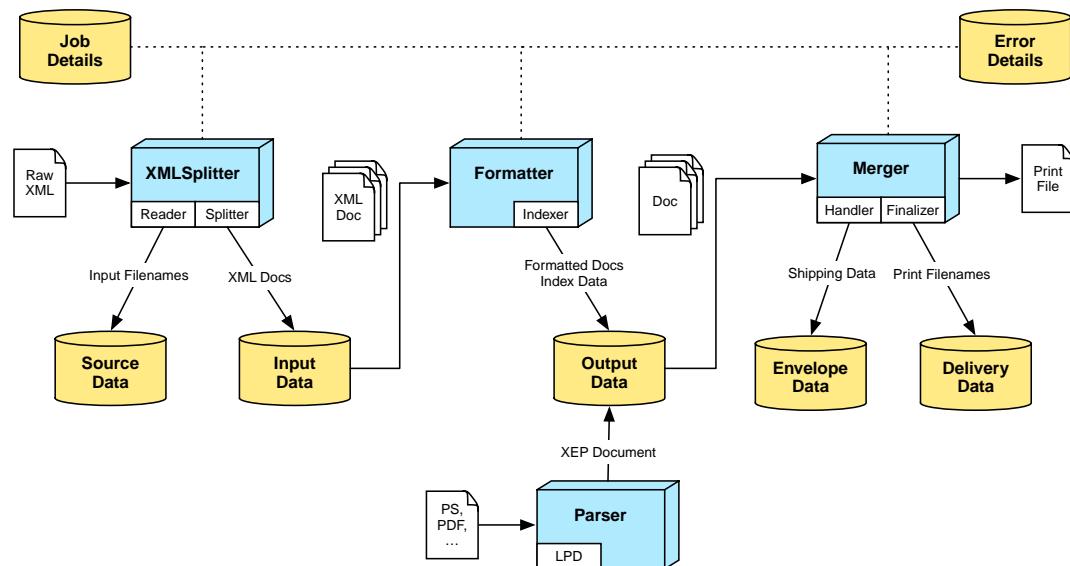
- **resources:** Resources are data that are created to be used by an application. Resources are not consumed by processing and do not expire. Resources are setup to be used in a defined status and do not change during the processing.
- **page:** A page in DocAdmin Console is a functionality that can be chosen in the task bar at the top of the application window.
- **tab:** A tab is a subpage of a functionality page. In general various tabs provide different views on the same data.
- **panel:** A panel is an area in a page or tab that provides information or manipulation options that belong together.

# Chapter 3

## Overview

As described before DocAdmin Console views data of the runtime tables of DocBase. This data is produced during the processing in the application. For further details see Figure 3.1, "DocBase processing - Overview".

Figure 3.1. DocBase processing - Overview



This graphic displays a typical DocBase process and which process step writes into which database table. Each database table in this graphic belongs to a certain page or tab of the “Job Browser”. For detailed information see Section 4.2, “Job Browser”.

# Chapter

# 4

## The graphical user interface

DocAdmin Console is a web application that is opened in a web browser. Therefore the DocAdmin Console server must be running. After a standard installation as described in "DocFamily Installation Guide" the web application can be accessed at the URL:

```
http://localhost:16002/DocAdminConsole
```

In this chapter first the general application elements are explained before it is described in particular how to use the application.

### 4.1. General application elements

Different elements can be found in various pages of the application or are global. They will be described once in this section. In case they occur in the application it will be referenced to this descriptions.

#### 4.1.1. Login page

The entry point for the application DocAdmin Console is the login page. It provides a drop down list of servers, entry fields for username and password and the button "Login".

Figure 4.1. Login Page - Overview

**assentus** 



The screenshot shows a login form with the following fields and button:

- 1 Repository:** A dropdown menu set to "DocAdmin Console Repository 12002".
- 2 Username:** An input field containing "uta".
- 3 Password:** An input field containing "•••".
- Login:** A button labeled "Login" located to the right of the password field.

1. **Repository:** DocAdmin Console can be configured to access various DocRepo. In this drop down list the user has to choose the current server he wants to monitor.

2. **Username:** The username of the user provided to this entry field has to be configured in the chosen DocRepo instance (1).
3. **Password:** The password of the user provided to this entry field has to be configured in the chosen DocRepo instance (1).
4. **Login:** The login process is executed. The user will be authenticated against the chosen DocRepo server. Afterwards the user can monitor all servers and processes that are registered or hosted in this instance.

## 4.1.2. Application page

The application window can be divided into different areas. The task bar at the top of the application window provides various buttons to open new pages that display one or another function area depending on their functionality.

- **Task bar**

The task bar is always present. It contains all links to the various pages where the user can access the functionalities of DocAdmin Console.

**Figure 4.2. General elements - Task bar**

■ Server Monitor ■ Job Browser ■ Start Job ■ Render Test ■ Job Scheduler ■ PROM Import ■ Attachment Management ■ Logout

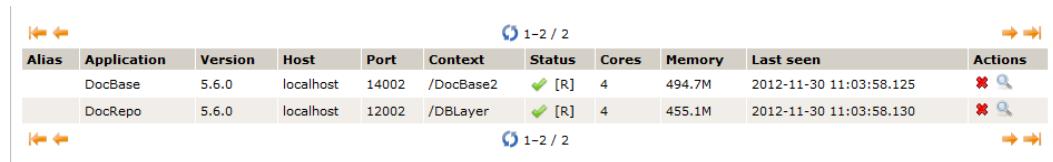
- **Tables**

Tables list the objects that belong to the currently opened functionality. The content of the columns depends by the displayed objects.

Each table is organized in pages, so only a limited number of objects is displayed at once. To browse the table pages, at the top and at the bottom on the left and right site the arrows allow to jump forth and back page by page or to the first respectively to the last page.

In the middle on top of the table a refresh button is located to reload the whole table. At the same place the counters indicate which items of how many available items are displayed.

**Figure 4.3. General elements - Table**



Alias	Application	Version	Host	Port	Context	Status	Cores	Memory	Last seen	Actions
DocBase	5.6.0	localhost	14002	/DocBase2	[R]	4	494.7M	2012-11-30 11:03:58.125		
DocRepo	5.6.0	localhost	12002	/DBLayer	[R]	4	455.1M	2012-11-30 11:03:58.130		
						1-2 / 2				

- **Selection**

For tables that come with a filter panel (see “Filter panel”) this column is displayed as first. Closely connected to enabling the check boxes per row are the functions that are offered by the table operations bar (see “Table operations bar”).

**Figure 4.4. General elements - Selection**



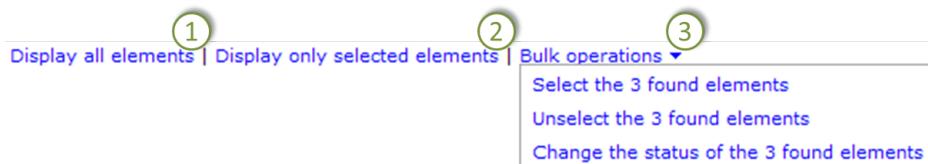
- **Table operations bar**

For the tables where the row selection (see “Selection) and the filter panel (see ‘Filter panel’) is displayed, this bar offers functions to manipulate the table rows. Thereby the number of effected rows depends by the usage of the filter. ”

1. **Display all elements:** Independently from the selection of rows, this link will display all rows of the table. In case, a filter has been applied this only the rows that match the filter are shown.
2. **Display only selected elements:** This link will display only the rows of the table that have been selected via the check box in the first column of each row. In case a filter has been applied to the table, only the selected rows that match the filter are shown.
3. **Bulk operations:** The downwards-arrow will open a menu. Per default the following options are available:
  - “Select the found elements”: Selects all elements that are currently available in the table.
  - “Unselect the found elements”: Unselects all elements that are currently available in the table.
  - “Change the status of the found elements”: Opens a dialog window where the user can change the status of all elements that are currently available in the table.

This menu is extendable by custom implementations. Therefore see the documents “DocBase Cookbook”, “DocBase Tasks Reference”, “DocBase Post Processing Guide” and the public API of DocAdmin Console to learn more about the possibilities to manipulate the DocBase runtime data and how to write a DocAdmin Console plug-in.

**Figure 4.5. General elements - Table operations bar**



- **Actions**

Most tables contain the column “Actions”. It contains buttons that will execute actions on the objects.

1. Removes the object from the table and completely from the database.
2. Opens the object in a detail page. This detail pages are specific for the displayed object. For further details see the description of the detail pages in the following chapters.

**Figure 4.6. General elements - Actions**



- **Filter panel**

For the pages with tables that show jobs or job related data, a filter panel can be displayed. In this panel the user can define various parameters to limit the displayed table objects. The filter criteria refer to the data that are available in DocRepo for each job.

Additionally, data can be filtered for custom properties. The result of the filtering is always displayed in a table (see "Tables").

**Figure 4.7. General elements - Filters**

Error Filter [show/hide]	
Database key	0
Selected	<input type="radio"/> yes <input type="radio"/> no <input checked="" type="radio"/> ignore
Database key of job	0
Error message	
Status	ALL
Created by	
Error Id	
Created from	
until	
<input type="button" value="Search ..."/> <input type="button" value="Reset Filter"/>	

- Parameter panel

Some functionality needs further configuration. Then a panel is displayed, where the required information can be inserted. Which input fields are displayed depends by the currently active functionality.

**Figure 4.8. General elements - Parameters**

Job Parameters	
DocBase server	http://localhost:14002/DocBase2
Jobcard ID	massProductionJobCard
Jobcard XML	massProductionJobCard.xml [ <a href="#">Clear</a> ]
<input type="button" value="Start"/>	

- Result panel

Certain functionalities produce a result which will be displayed in the result panel. It will provide further information to the executed function and it will show a link where the result can be reviewed.

**Figure 4.9. General elements - Result**

Job Status	
Job started successfully. JobID: 87ee0d6d-1a5b-4b32-a719-cc6676bc48f7	

- Navigation

In detail pages that have been opened from a table overview by executing the action "View details" the following control is displayed. It refers to the table in the overview the detail page has been opened from.

1. **Previous:** Jumps forward in the table - one row upwards - and displays the detail page with the details of the object in the row above the current one.
2. **Next:** Jumps to the next row of the table and displays the details of the regarding object.
3. **Back:** Clicking this button leads the user back to the page from where the detail page has been opened.

**Figure 4.10. General elements - Navigation**



## 4.2. Job Browser

The most important part of DocAdmin Console is the page “Job Browser”. It is supposed to provide all views and functions that enable a user to track and control the processes of DocBase servers that belong to the DocFamily environment DocAdmin Console is currently signed on.



### Tip

Also it is necessary that at least one process via DocBase has been executed and written some data into the environment database. To learn more about the setup of DocBase tasks and how to group them to processes see the documents “DocBase Cookbook”, “DocBase Post Processing Guide” and “DocBase Tasks Reference”. Also read the document “DocBase Post Processing Guide”.

Since the page “Job Browser” shows different views of the runtime data of a DocBase server, it can not display any data in case no DocBase is connected to the current DocFamily environment.

All sub pages of the job browser are devided into the filter panel and a table that shows the objects of current runtime table. As explained in Section 4.1.2, “Application page”, “Filter panel” the fields to filter depend by the runtime table that is active.

In the details view of a particular table entry further overview tables can be displayed. In general over the whole application, tables with the same name in various views display the same information. It is possible that the displayed entries are filtered by the previous view.

### 4.2.1. Jobs

This table lists all jobs that are available in the table “Job Details” (see Figure 3.1, “DocBase processing - Overview”).

**Figure 4.11. Job Browser - Jobs - Overview**

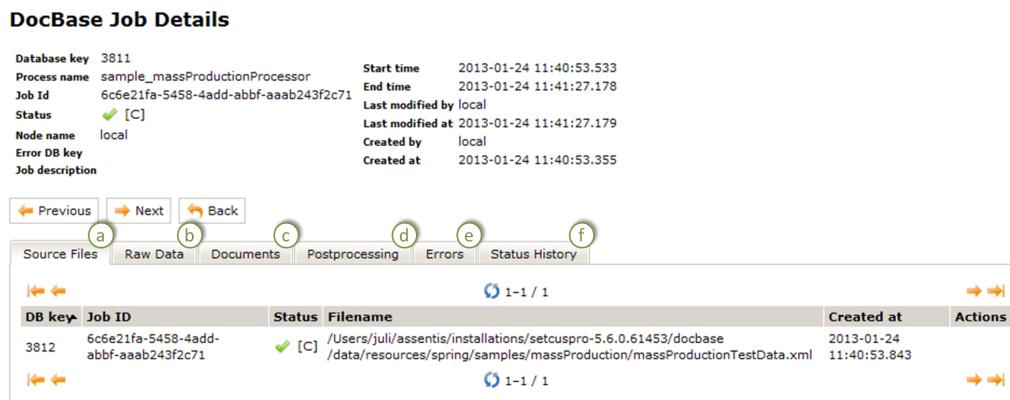
DocBase Job Browser																	
Jobs		Errors		Raw Data		Documents											
Job Filter [show/hide] (9)																	
Display all elements   Display only selected elements   Bulk operations																	
<span style="float: left;">◀ ▶</span> <span style="float: right;">◀ ▶</span> <span style="margin-left: 10px;">1-3 / 3</span>																	
DB key		Job ID		Status	Process name	Server	Node name	Created at	Actions								
<input type="checkbox"/>	3811	(1) 6c6e21fa-5458-4add-abbf-aaab243f2c	(2)  [C] sample_massProductionProcessor	(3)	(4) sample_massProductionProcessor	(5)	(6) local	2013-01-24 11:40:53.	(7)  (8)								
<input type="checkbox"/>	3839	c4c61a3e-fe3d-406d-a7d6-66d697d0bdd6	[C] adb_simpleRenderOnlySequenceProcessor				local	2013-01-24 11:42:38.866									
<input type="checkbox"/>	3840	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[P] splitRenderMergeProcess				local	2013-01-24 11:43:40.996									
<span style="float: left;">◀ ▶</span>		1-3 / 3															

1. **DB Key:** The unique database key of a job.

2. **Job ID:** The Id of a job execution is displayed. The job Id is a generated unique identifier. It is not only unique for one DocBase instance jobs are produced by but must be unique for the whole DocFamily environment since various DocBase instances can store jobs in it.
3. **Status:** This column displays the current status of the job. Predefined the status [W] for a job waiting for its first executing is known as well as the status [F] for an erroneous job execution. The status [R] marks a currently running job, a stopped or disabled job will display status [S].
4. **Process name:** The process definition a job definition is based on has a name that is displayed in this column. Since various job definitions can base on the same process definition and a job card can be executed various times, the table may contain numerous entries with the same process name.
5. **Server:** DocBase server a job has been executed. The name the DocBase instance has been registered in DocAdmin Console is displayed.
6. **Node name:** The node is the physical or virtual machine that can host various server applications and therefore various DocBase servers. Its name is displayed in this table column.
7. **Created at:** This time is the time stamp when the current job execution has been created. This should relate to the cron expression of the job that defines the regular execution of it.
8. **View details:** This action will open a detail view for the current job execution. It displays all the properties of a job that can be used for filtering a job execution out of the overview table of the job browser. For more information see "Job Filter".

**Figure 4.12. Job Browser - DocBase Job Details**

**DocBase Job Details**



Database key	3811	Start time	2013-01-24 11:40:53.533
Process name	sample_massProductionProcessor	End time	2013-01-24 11:41:27.178
Job Id	6c6e21fa-5458-4add-abbf-aaab243f2c71	Last modified by	local
Status	[C]	Last modified at	2013-01-24 11:41:27.179
Node name	local	Created by	local
Error DB key		Created at	2013-01-24 11:40:53.355
Job description			

Navigation bar with tabs labeled a-f:

- a: Source Files
- b: Raw Data
- c: Documents
- d: Postprocessing
- e: Errors
- f: Status History

Table below the navigation bar:

DB key	Job ID	Status	Filename	Created at	Actions
3812	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	/Users/juli/assentis/installations/setcuspro-5.6.0.61453/docbase /data/resources/spring/samples/massProduction/massProductionTestData.xml	2013-01-24 11:40:53.843	

Additionally the detail view displays various tables with data related to a job execution. For the entries of all tables in the job details view except for the table "Source Files" it is possible to open again a detail view of the table entry. In case the object contains any data it is mostly possible to view and download the data as files.

- a. **Source Files:** Lists all XML input files that are used by this job execution.
- b. **Raw Data:** This table lists all data units that are used to produce documents. This raw data result from the possible splitting of the XML input files.
- c. **Documents:** Lists the output data of the formatting step of the job. This table also will list additional documents that will be inserted during postprocessing.
- d. **Postprocessing:** Lists the postprocessing tasks that have been executed for the job. It is possible to open a detail view for the task.

**Figure 4.13. Job Browser - DocBase Job Details - Postprocessing Details**

**Postprocessing Execution Details**

Job DB key	3826
Process name	adb_postprocessingTask
Job Id	6c6e21fa-5458-4add-abbf-aaab243f2c71
Status	✓ [C]
Start time	2013-01-24 11:41:25.100
End time	2013-01-24 11:41:27.174
Execution DB key	3827

Buttons: Previous, Next, Back

Tabs: Documents, Envelopes, Streams

Stream Filter: [show/hide]

Table Headers: DB key, Name, Status, Created at, Actions

Table Data:

DB key	Name	Status	Created at	Actions
3829	UNKNOWN	✓ [C]	2013-01-24 11:41:25.493	[Search]

The three tables of the details view present the three object types a postprocessing handles. It gets the documents, composes them to envelopes and sends them to various streams.

- e. **Errors:** Table of error entries for this job execution.
- f. **Status History:** Lists the status the job execution has passed until so far.
- 9. **Job Filter:** Opening the filter panel provides various entry fields to insert values to filter the jobs entries of the overview table.

**Figure 4.14. Job Browser - Jobs - Filter**

**Job Filter** [show/hide] q

a Database key	0	h Selected	<input type="radio"/> yes <input type="radio"/> no <input checked="" type="radio"/> ignore
b Process name		i Start time from	until
c Job Id		j End time from	until
d Status	ALL	k Last modified by	
e Node name		l Last modified from	until
f Error database key	0	m Created by	
g Job description		n Created from	until

Buttons: Search ..., Reset Filter

- a. **Database key:** Each element in the DocRepo database has its unique database key. A search with this criteria will deliver one dedicated job as result.
- b. **Process name:** Each job is based on a process definition. This process has a name which can be used in this entry field as search criteria. A search with this criteria can deliver a set of jobs as result.
- c. **Job Id:** Each job has an over the DocFamily environment unique job Id that can be used for filtering the job entries. In this entry field wildcards can be used. A search without wildcards will deliver one single job as result.
- d. **Status:** All status known in the DocFamily environment DocAdmin Console is connected to, can be selected from the drop down list. A search with this criteria can deliver a set of jobs as result.

- e. **Node name:** The node name of the physical or virtual machine that hosts the executing server application can be used as filter criteria. A search with this criteria can deliver a set of jobs as result.
- f. **Error database key:** As every other object in the DocAdmin Console error entries also have database keys. As soon as an error is related to a job execution its database key can be used as filter criteria. A search with this criteria will deliver a dedicated job as result.
- g. **Job description:** Searches the description of the job that is stored in a table of the Doc-Family environment.
- h. **Selected:** Depending on the selected radio button only the selected (**yes**), the not selected (**no**) or all (**ignore**) rows of the table are filtered.
- i. **Start time from/ until:** Time range that will deliver as result all jobs that were started in this range.
- j. **End time from/ until:** Time range that will deliver as result all jobs that ended in this range.
- k. **Last modified by:** This entry field requires the name of a user that can authenticate against the currently connected DocRepo instance. The search result will contain all jobs that were modified last by this user. It is possible to use wildcards in this entry field.
- l. **Last modified from/ until:** Time range that will deliver as result all jobs that were modified last in this range.
- m. **Created by:** This entry field requires the name of a user that can authenticate against the currently connected DocRepo instance. The search result will contain all jobs that were created by this user. It is possible to use wildcards in this entry field.
- n. **Created from/ until:** Time range that will deliver as result all jobs that were created in this range.
- o. **Search ...:** Pressing this button will execute the search regarding the configured filter. The search result will be displayed in the overview table below the filter panel.
- p. **Reset Filter:** With this button the result of the filtering will be resetted. Afterwards the overview table below the filter panel will display again all entries available in the database.
- q. **show/hide:** This link will show or hide the filter panel. In case a filter has been configured it will be still active even if the panel is hidden.

## 4.2.2. Errors

This table lists all errors that are available in the table “Error Details” (see Figure 3.1, “DocBase processing - Overview”).

Figure 4.15. Job Browser - Errors - Overview

DB key	Error ID	Status	Created by	Actions
3932	ADB-XS-0002	[W]	local	(a)

1. **DB Key:** Every object in DocRepo has a unique database key that is displayed here.
2. **Error ID:** Depending on the type, DocBase errors provide an error Id. So it is possible to filter for an error type.
3. **Status:** Displays the status the error provoked. After the error occurred it has the status [W] until the status will be changed by a housekeeping process or manually by an administrator.
4. **Created by:** Displays the name of the node where an erroneous job run and that therefore created an error entry in the database.
5. **Created at:** Displays the time the error entry has been created in the database.
6. **View details:** This action will open a details page for the error entry. It also shows various tables with objects that belong to the error, like jobs, source files, raw data documents and the status history. See the descriptions of this tables also in this chapter.

Figure 4.16. Job Browser - DocBase Error Details

DB key	Job ID	Status	Process name	Server	Node name	Created at	Actions
3931	6b3a2bb2-a019-479c-8fbd-ffff5f47ab021	[F]	sample_massProductionProcessor	local	sample_massProductionProcessor	2013-01-24 21:42:19.586	

- a. **Status:** In this view the link enables the user to change the status of the error. This might be the case when he has reviewed it and taken taken measures, so he can set the status to [C] (completed). The possible status can be selected from a drop down list. The selection has to be saved afterwards.

- b. **Error data:** Clicking this link opens a separate page and displays the stack trace that has been produced when the job execution threw the error.
7. **Error Filter:** Opening the filter panel provides various entry fields to insert values to filter the error entries of the overview table. The entry fields refer to the columns of the overview table.

**Figure 4.17. Job Browser - Error - Filter**

Database key	0	Selected	<input type="radio"/> yes <input type="radio"/> no <input checked="" type="radio"/> ignore
Database key of job	0	Error message	
Status	ALL	Created by	
Error Id		Created from	until
<input type="button" value="Search ..."/> <input type="button" value="Reset Filter"/>			

### 4.2.3. Raw Data

This table lists all data that are available in the table “Input Data” (see Figure 3.1, “DocBase processing - Overview”). This data must not be confused with the input files. From an input file various entries in the table “Input Data” can be produced by splitting the input file into smaller units. The input data will be used later on to produce documents. Per input data entry one or more documents are created. Per input file various documents may result.

**Figure 4.18. Job Browser - Raw Data - Overview**

DB key	Job ID	Status	Content type	Created at	Actions
3813	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xml	2013-01-24 11:40:53.931	(1)  (2)  (3)  (4)  (5)  (6)
3814	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xml	2013-01-24 11:40:54.377	
3842	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[C]	xml	2013-01-24 11:43:41.008	
3843	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[C]	xml	2013-01-24 11:43:41.012	
3844	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[C]	xml	2013-01-24 11:43:41.064	

1. **DB Key:** Every object in DocRepo has a unique database key that is displayed here.
2. **Job ID:** The Id of a job execution is displayed. The job Id is a generated unique identifier. It is not only unique for one DocBase instance jobs are produced by but must be unique for the whole DocFamily environment since various DocBase instances can store jobs in it.
3. **Status:** This column displays the current status of production of the raw data. The status [C] marks raw data which production is completed.
4. **Content type:** This displays the content type of the raw data. This usually matches the file ending of the uploaded source file the raw data were produced from.
5. **Created at:** The column displays the time, the raw data have been created and stored in the database.

6. **View details:** This action opens the details page of a raw data table entry. It also shows various tables with objects that belong to the raw data like the status history. See the descriptions of this table also in this chapter.

The custom attributes overview table lists the custom attributes that are available for the raw data files. For each custom attribute the name and the value are displayed.

**Figure 4.19. Job Browser - DocBase Raw Data Details**

The screenshot shows the 'DocBase Raw Data Details' page. At the top, there is a summary table with columns for Database key, Status, Document alias, Format, Content type, and Content. Below this is a navigation bar with 'Previous', 'Next', and 'Back' buttons. A tab bar at the bottom includes 'Custom Attributes' (which is selected) and 'Status History'. The main content area displays a table of custom attributes with columns for Name and Value. There are also navigation arrows for the table.

Database key	3813	Job ID	6c6e21fa-5458-4add-abbf-aaab243f2c71								
Status	<input checked="" type="checkbox"/> [C] <a href="#">Edit</a>	Error DB key									
Document alias	sample_massProduction	Last modified by	local								
Format	xep	Last modified	2013-01-24 11:41:05.630								
Content type	xml	Created by	local								
Content	<a href="#">Show content</a>	Created at	2013-01-24 11:40:53.931								
# of pages	-1	validity	2013-02-13 11:40:53.931								
<a href="#">Previous</a> <a href="#">Next</a> <a href="#">Back</a>											
<a href="#">Custom Attributes</a> <a href="#">Status History</a>											
<table border="1"> <thead> <tr> <th colspan="2">1-1 / 1</th> </tr> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>ID</td> <td>01</td> </tr> <tr> <td colspan="2"> <a href="#">1-1 / 1</a> </td> </tr> </tbody> </table>				1-1 / 1		Name	Value	ID	01	<a href="#">1-1 / 1</a>	
1-1 / 1											
Name	Value										
ID	01										
<a href="#">1-1 / 1</a>											

- a. **Status:** In this view the link enables the user to change the status of the error. This might be the case when he has reviewed it and taken measures, so he can set the status to [C] (completed). The possible status can be selected from a drop down list. The selection has to be saved afterwards.
  - b. **Content:** Clicking this link opens a separate page and displays the content of the raw data entry.
7. **Raw Data Filter:** Opening the filter panel provides various entry fields to insert values to filter the raw data entries of the overview table. The entry fields refer to the columns of the overview table.

**Figure 4.20. Job Browser - Raw Data - Filter**

The screenshot shows the 'Raw Data Filter' panel. It contains two columns of input fields. The left column includes fields for Database key (with value 0), Job id, Status (set to ALL), Error database key (with value 0), Document alias, Format, Content type, Created by, Created after, Last modified by, Last modified after, and Valid after. The right column includes fields for Selected (radio buttons for yes, no, ignore), Document id, Custom attribute name 1, Custom attribute value 1, Custom attribute name 2, Custom attribute value 2, Custom attribute name 3, Custom attribute value 3, Custom attribute name 4, Custom attribute value 4, Custom attribute name 5, and Custom attribute value 5. At the bottom are 'Search ...' and 'Reset Filter' buttons.

Raw Data Filter <a href="#">[show/hide]</a>			
Database key	0	Selected	<input type="radio"/> yes <input type="radio"/> no <input checked="" type="radio"/> ignore
Job id		Document id	
Status	ALL	Custom attribute name 1	
Error database key	0	Custom attribute value 1	
Document alias		Custom attribute name 2	
Format		Custom attribute value 2	
Content type		Custom attribute name 3	
Created by		Custom attribute value 3	
Created after	before	Custom attribute name 4	
Last modified by		Custom attribute value 4	
Last modified after	before	Custom attribute name 5	
Valid after	before	Custom attribute value 5	
<a href="#">Search ...</a> <a href="#">Reset Filter</a>			

## 4.2.4. Documents

This table lists all data that are available in the table “Output Data” (see Figure 3.1, “DocBase processing - Overview”). This data must not be confused with the final print files. One entry in the table contains one document. A print file can be composed from various documents by merging them.

**Figure 4.21. Job Browser - Documents - Overview**

DocBase Job Browser									
Jobs	Errors	Raw Data	Documents						
<b>Document Filter [show/hide] (7)</b>									
Display all elements   Display only selected elements   Bulk operations ▾									
DB key	Job ID	Status	Format	Created at	Actions				
3816 (1)	6c6e21fa-5458-4add-abbf-aaab243f2c71 (2)	[C] (3)	xep (4)	2013-01-24 11:41:05.059 (5)	(6)				
3817	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.059					
3818	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.570					
3819	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.570					
3820	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.583					
3821	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.584					
3822	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.606					
3823	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.611					
3824	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.621					
3825	6c6e21fa-5458-4add-abbf-aaab243f2c71	[C]	xep	2013-01-24 11:41:05.624					
3846	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[W]	xep	2013-01-24 11:43:41.804					
3847	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[W]	xep	2013-01-24 11:43:41.862					
3848	1f291f83-c59a-401a-93cb-ccc4a37d31d6	[P]	xep	2013-01-24 11:43:42.183					

- 1. DB Key:** Every object in DocRepo has a unique database key that is displayed here.
- 2. Job ID:** The Id of a job execution is displayed. The job Id is a generated unique identifier. It is not only unique for one DocBase instance jobs are produced by but must be unique for the whole application landscape since various DocBase instances can store jobs in a single DocRepo instance.
- 3. Status:** This column displays the current status of production of the document. The status [C] marks documents which production is completed. Documents with status [W] await their processing, while the status [P] indicates that this document is currently processed.
- 4. Format:** This displays the format of the document. Only two formats are possible: xep and xsl-fo. They are the base for the production of different print files. See the document “Doc-Family Compatibility” for more information about the valid format combinations.
- 5. Created at:** The column displays the time, the documents have been created and stored in the database.
- 6. View details:** This action opens the details page of a document table entry. It also shows various tables with objects that belong to the document like the status history. See the descriptions of this table also in this chapter.

The custom attributes overview table lists the custom attributes that are available for the document. For each custom attribute the name and the value are displayed.

Figure 4.22. Job Browser - DocBase Document Details

**DocBase Document Details**

Database key	3825	Job ID	6c6e21fa-5458-4add-abbf-aaab243f2c71
Status	[C] [Edit]	DB key	(a)
Input data DB key	3813	Last modified by	local
	Show raw data content	last modified	2013-01-24 11:41:05.624
Format	xep	created by	local
Content	Show content	created at	2013-01-24 11:41:05.624
	Display as PDF	entity	2013-02-13 11:41:05.624
# of pages	1	Customer ID	-1
# of sheets	1	Resource ID	-1
Print file name			

[Previous](#) [Next](#) [Back](#)

Custom Attributes Status History

Name	Value
ID	05

- a. **Status:** In this view the link enables the user to change the status of the error. This might be the case when he has reviewed it and taken measures, so he can set the status to [C] (completed). The possible status can be selected from a drop down list. The selection has to be saved afterwards.
- b. **Input data DB key - Show raw data content:** Clicking this link opens a separate page and displays the content of the raw data file that has been used to produce the document entry.
- c. **Content - Show content:** Clicking this link opens a separate page and displays the content of the document entry.
- d. **Content - Display as PDF:** Clicking this link opens a separate page and displays the content of the document as PDF file. Therefore a running DocBase instance is necessary.
7. **Document Filter:** Opening the filter panel provides various entry fields to insert values to filter the document entries of the overview table. The entry fields refer to the columns of the overview table.

Figure 4.23. Job Browser - Document - Filter

**Document Filter [show/hide]**

Database key	0	Selected	<input type="radio"/> yes <input type="radio"/> no <input checked="" type="radio"/> ignore
Job id		Print filename	
Status	ALL	Resource Id	0
Error database key	0	Customer Id	
Format		Custom attribute name 1	
Content type		Custom attribute value 1	
Pages (min)	0	Custom attribute name 2	
Sheets (min)	0	Custom attribute value 2	
Last modified by		Custom attribute name 3	
Last modified after	before	Custom attribute value 3	
Created by		Custom attribute name 4	
Created after	before	Custom attribute value 4	
Valid after	before	Custom attribute name 5	
Input data id	0	Custom attribute value 5	

**Buttons:** Search ..., Reset Filter

## 4.3. Server Monitor

In the server monitor the user can register other DocFamily server applications. Therefore he has to enter the complete URL into the input field and click the "Add" button (see Figure 4.24, "Server Monitor - Overview", (12)).

Figure 4.24. Server Monitor - Overview

**DocFamily Server Monitor**

**Applications**

Alias	Application	Version	Host	Port	Context	Status	Cores	Memory	Last seen	Actions
(13) DocBase (1)	5.6.0 (2)	localhost (3)	14002 (4)	DocBase2 (5)	✓ [R] (6)	4 (7)	482.5M (8)	2012-11-28 16:43:12.286 (9)	(10)  (11)	
DocRepo	5.6.0	localhost	12002	/DBLayer	✓ [R]	4	455.1M	2012-11-28 16:43:12.294		

**Add Application**

Server URL:  Add (12)

All registered server applications are listed in a table (see Figure 4.24, "Server Monitor - Overview") which displays various columns with the following information:

1. **Application:** The column displays what kind of Assentis product component is running as server application, e.g. DocRepo, DocBase, DocWrite or even another DocAdmin Console instance. This information is detected automatically.
2. **Version:** Displays the version of the server application.
3. **Host:** The host shows the IP or the DNS name the server application has been registered in DocAdmin Console.

4. **Port:** The column shows the port the server application has been registered in DocAdmin Console.
5. **Context:** Here the application context is displayed the server application has been registered in DocAdmin Console.
6. **Status:** DocAdmin Console detects the status of each registered server application. The status [R] points out a running server, the status [S] indicates a stopped server. The status [W] indicates a server for which never until so far a status could be evaluated after its registration. For a server for which already a status has been evaluated after its registration but for which the last status evaluation failed, the status [F] is displayed.
7. **Cores:** The column displays the cores of the machine (physical or virtual) the JVM that runs the server application has access to.
8. **Memory:** Here the total memory of the base JVM is displayed. This is the total size of the allocated memory for the JVM since it has been started. The memory can increase until the maximal defined memory size is reached, but never decrease.
9. **Last seen:** Timestamp of the last connection and status evaluation of the server application.
10. **Delete:** Action button. The server application is removed from the table and won't be monitored anymore.
11. **View details:** Action button. Opens a separate page that displays the details of the server application and the machine it is running.

### 4.3.1. View details

Figure 4.25. Server Monitor - Server Details



The screenshot shows the 'Server Details' page with the following sections and data:

- Server Details (Top Panel):**
  - Application: DocRepo
  - Version: 5.6.0
  - Alias: [\[Edit\]](#) (1)
  - Scheme: http
  - Host: localhost
  - Port: 12002
  - Context: /DBLayer
  - Status: [R]
  - Last seen: 2012-12-03 15:10:32.861
- Navigation:** Previous, Next, Back
- Overview (Panel 2):**
  - Application: DocRepo ([Status Page](#))
  - Build version: 5.6.0.61041
  - Java version: 1.6.0\_35
  - JVM vendor: Sun Microsystems Inc.
  - Operating system: Windows 7 6.1
  - Architecture: amd64
  - Webapp container: Apache Tomcat 7.0.30
  - Available processors: 4
  - Memory (used): 114.8M ([Run GC](#)) (7)
  - Memory (free): 99.1M
  - Memory (assigned): 455.1M
- Environment (Panel 3), JVM Threads (Panel 4), Configuration (Panel 5):** These tabs are visible at the top of the overview panel.

The action button "View details" opens a separate view of the server details. As displayed in Figure 4.25, "Server Monitor - Server Details", in the upper panel all information from the server table are displayed. Here the user can insert an alias for the server (1). This alias then is displayed in the server overview (Figure 4.24, "Server Monitor - Overview", (13)) and in the places, where servers can be selected, e.g. in the page "Start DocBase Jobcard" (Section 4.4, "Start Job"). This alias has no functional aspect, it is just for visualization and better recognition.

The tabs in the lower panel (2, 3, 4, 5) display various information about the server application and the machine (physical or virtual) the server runs at, grouped by topic.

The tab “Overview” (2) details the global server application properties and provides two actions:

6. **Status Page:** Clicking this link, will open the application internal status page in a new browser tab or window. The functionality offered there depends by the application. It runs completely independ from DocAdmin Console.
7. **Run GC:** This will run the garbage collection of the JVM that runs the server application to free used memory.

All properties of the system and of the environment are listed in the tab “Environment” (3).

In the tab “JVM Threads” (4) all active threads of the JVM running the server application are displayed (see Figure 4.26, “Server Monitor - Server Details - JVM Threads”).

**Figure 4.26. Server Monitor - Server Details - JVM Threads**

ID	Name	Group	Prio	State	Actions
20	http-bio-12002-exec-4	main	5	WAITING	Show stacktrace
21	http-bio-12002-exec-5	main	5	WAITING	Show stacktrace
18	http-bio-12002-exec-2	main	5	WAITING	Show stacktrace
4	Signal Dispatcher	system	9	RUNNABLE	Show stacktrace
5	Attach Listener	system	5	RUNNABLE	Show stacktrace
3	Finalizer	system	8	WAITING	Show stacktrace
19	http-bio-12002-exec-3	main	5	WAITING	Show stacktrace
23	http-bio-12002-exec-7	main	5	WAITING	Show stacktrace
2	Reference Handler	system	10	WAITING	Show stacktrace

8. **Show stacktrace:** Opens the current stacktrace of the regarding thread in a new browser tab or window.

The tab “Configuration” (5) displays the server application internal configuration, e.g. the content of the file “DocBaseConfig.xml” or “RepositoryConfig.xml”.

## 4.4. Start Job

Within this page the user can start a DocBase job card. This executes a job immediately. Thereby the user has always to provide the Id of a job card.

This Id may refer to a job card that, depending on the configuration of DocBase, is already available in the DocFamily environment the DocBase instance belongs to, or that is available in the `spring` folder DocBase has configured as job card resources directory.

The user also can select a job card definition file from the local machine, the DocAdmin Console GUI is opened. This job card will be uploaded to the DocAdmin Console server application immediately after it has been chosen. It will be available as long as the server application is not shut down and the user does not close the application browser window.

1. **DocBase server:** From the servers that have been registered in the DocAdmin Console one has to be chosen that executes the job card
2. **Jobcard ID:** As displayed in the code sample below, each job card contains an Id that has to be provided in this entry field. The Id has to be inserted in any case, independent if the job card already exists in the DocFamily environment or if the job card is uploaded for this execution (see (3)).

```
<bean class="com.assentis.db.starter.IOCWorkflowJobCard" id="massProductionJobCard"
      scope="prototype">
```

**Figure 4.27. Start DocBase Jobcard - Overview**

**Start DocBase Jobcard**

<b>Job Parameters</b>	DocBase server: http://localhost14002/DocBase2 Jobcard ID: massProductionJobCard Jobcard XML: Choose a file... <input type="button" value="Start"/>	 
<b>Job Status</b>	Job started successfully. JobID: d3c267c1-cb86-4641-9e73-39d1aa578866 	

3. **Jobcard XML:** The user can upload here a job card description file. In the opening dialog a file from the local system the DocAdmin Console GUI is operated can be chosen. This file will be uploaded to the DocAdmin Console server. The Id in this file has to match the job card Id provided in (2). For more information about job card description files see the documents "DocBase Cookbook" and "DocBase Tasks Reference".

**Jobcard XML**  massProductionJobCard.xml [[Clear](#)]

Once the file is uploaded it can be removed again by pressing the "Clear" link. Afterwards a new file can be uploaded.

4. **Start:** Klicking this button will start the execution of the job card.
5. **Job Status:** After starting the execution of the job, the status of the job is displayed in this area.

## 4.5. Render Test

This page provides functionality to check the rendering of documents. An XML data input file can be used that will be processed by an deployment package.

1. **DocBase server:** From the servers that have been registered in the DocAdmin Console one has to be chosen that executes the job card.
2. **Input type:** DocBase is able to process different input types. The input type of the source file has to be chosen in this drop down list.
3. **Output format:** DocBase is able to produce various output formats. The output format of the result file has to be chosen in this drop down list. See the document "DocFamily Known Limitations" for more information about existing limitations of output formats. Not all output formats can be produced from all input types. See the document "DocFamily Compatibility" to find the available input type - output format combinations.
4. **Document alias:** To run a render test, a deployment package has to be selected. Each deployment package is identified by its alias which has to be provided in this entry field. This alias has been given during design time of the document.

**Figure 4.28. DocBase Render Test - Overview**

5. **Source File:** The user can upload here an XML data file. In the opening dialog a file from the local system the DocAdmin Console GUI is operated can be chosen. This file will be uploaded to the DocAdmin Console server. It is available as long as the server application is not shut down or the user closes the browser application window. The XML data file has to provide a structure matching the needs of the deployment package (4). This structure has been defined during design time of the document.

**Source file**  docs.xml [[Clear](#)]

Once the file is uploaded it can be removed again by pressing the "Clear" link. Afterwards a new file can be uploaded.

6. **Render:** This button renders the document using the deployment package and in case it has been uploaded, the XML data file. The file is stored in the DocBase runtime tables of the DocFamily environment.
7. **Render Result:** This area displays the result of the render test. In case the rendering has been executed successfully a link is displayed where the render result can be reviewed. Clicking this link will open a new browser page or window, or it will open a download dialog in case the output format can not be displayed by the browser.

## 4.6. Job Scheduler

Jobs usually are scheduled time based. They are supposed to run on certain time. This application page gives the operator the opportunity to schedule a job manually. Either the job definition already exists in the DocFamily environment or in the `spring resources` directory of the DocBase instance, or it is uploaded during the configuration into a dedicated DocAdmin Console table inside of the environment.



### Warning

To execute the scheduled job it is necessary that not only the target servers like DocRepo and DocBase are up and running, but DocAdmin Console as well.

The application page "Job Scheduler" is divided into a table that lists all scheduled jobs and a parameter panel where the user can insert all necessary data to schedule new jobs. To explain the functionality in detail, see Figure 4.30, "DocBase Job Scheduler - Overview":

1. **Name:** For each job a name can be defined in the parameter panel (1a). This name is used for better identification and has no functional significance.

2. **Server:** From the servers that have been registered the user has to select a DocBase instance in the parameter panel (2a) that will execute the job.
3. **Next run at:** Displays the time the job will run the next time. It is connected to the cron expression (3a) that defines a timer pattern. For more information how to define the expression, click the "Help" link.

Usually jobs are supposed to be scheduled by this cron expressions to be executed regularly repeated. But it is also possible to define a single run.

4. **Status:** Displays the current state of a job execution. The following job status can occur:

W	Waiting for processing.
BW	Waiting for processing. This status is only used for BigDocs.
W-REPRINT	Waiting for processing. This status is only used for reprint jobs.
P	Currently being processed.
C	Completed.
S	Stopped.
F	Failed.
M	Job results are merged to another execution.
R	Ready.
WX	Waiting for processing. This status is set by an external system.
PX	Currently being processed. This status is set by an external system.
CX	Completed. This status is set by an external system.
FX	Failed. This status is set by an external system.
UNDEFINED	The job status is undefined.
W-JOBONLY	Not yet being processed. This status is only used for BigDocs.

5. **Last run at:** As soon as a job has been executed for the first time, the time last execution time is displayed here. This time should correlate with the timer which is defined by the cron expression (3a).
6. **View details:** Clicking this action will open a separate page that displays the details of scheduled job.

Figure 4.29. DocBase Job Scheduler - Scheduled Job Details

### Scheduled Job Details

Name	Nightly Render <a href="#">[Edit]</a>	(a)
Cron expression	0 59 23 ? * MON-FRI	
Next run at	2013-01-01 23:59:00	
Enabled	<input checked="" type="checkbox"/>	(b)
Server	http://localhost:14002/DocBase2	
Jobcard ID	renderonly	
Jobcard	<a href="#">Download Jobcard</a>	(c)
Status	 [W]	
Last run at		
Message		

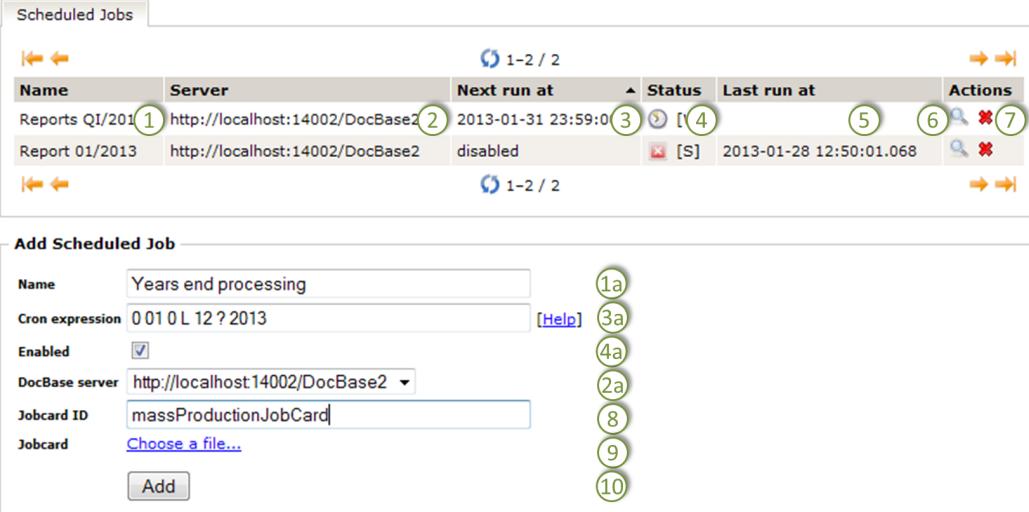
- a. The user can edit the name of the scheduled job here.
- b. In the details view the user can configure if the scheduled job is active or not. This will influence the state of the job which will turn in to "Stopped" ([S]) in case the job is disabled.
- c. Since the job card description file is stored in the DocFamily environment, the user has the option to download it later on.

7. **Delete job:** Action button that will remove the scheduled job and all execution data from the table and from the DocFamily environment.
8. **Jobcard ID:** As displayed in the code sample below, each job card contains an Id that has to be provided in this entry field. The Id has to be inserted in any case, independent if the job card already exists in the DocFamily environment or if the job card is uploaded for this execution (see (9)).

```
<bean class="com.assentis.db.starter.IOCWorkflowJobCard" id="massProductionJobCard"
      scope="prototype">
```

Figure 4.30. DocBase Job Scheduler - Overview

### DocBase Job Scheduler



The screenshot shows the DocBase Job Scheduler interface with two main sections:

- Scheduled Jobs:** A table listing scheduled jobs. The columns are: Name, Server, Next run at, Status, Last run at, and Actions. The table contains two rows:
  - Row 1: Reports Q1/2013, http://localhost:14002/DocBase2, 2013-01-31 23:59:00, [I] (disabled), 2013-01-28 12:50:01.068, Actions (edit, search, delete).
  - Row 2: Report 01/2013, http://localhost:14002/DocBase2, disabled, [S], 2013-01-28 12:50:01.068, Actions (edit, search, delete).
- Add Scheduled Job:** A form for adding new scheduled jobs. The fields are: Name (Years end processing), Cron expression (0 0 1 12 ? 2013), Enabled (checked), DocBase server (http://localhost:14002/DocBase2), Jobcard ID (massProductionJobCard), and Jobcard (Choose a file...). There is also an 'Add' button.

9. **Jobcard:** The user can upload a job card description file. In the opening dialog a file from the local system the DocAdmin Console GUI is operated can be chosen. This file will be uploaded

into a table of DocAdmin Console inside of the DocFamily environment. The Id in this file has to match the job card Id provided in (8). For more information about job card description files see the documents “DocBase Cookbook” and “DocBase Tasks Reference”.

**Jobcard**  massProductionJobCard.xml [[Clear](#)]

Once the file is uploaded it can be removed again by pressing the “Clear” link. Afterwards a new file can be uploaded.

- 10 **Add**: This button will add the job scheduler configuration to the DocAdmin Console table of the DocFamily environment. The job then is displayed in the overview table of scheduled jobs.

## 4.7. PROM Import

This page offers functionality to upload a promotion file to one of the registered DocRepo instances. The import is two-stage. The user defines a timer-based promotion file import job. Immediately when adding the job, the selected promotion file will be uploaded to DocRepo as a file (first stage). It is stored there until the import job is triggered, the promotion file from the database is expanded and the single files are pushed into DocRepo (second stage).



### Warning

To import the promotion file it is necessary that not only the target servers like DocRepo are up and running, but DocAdmin Console as well.

The user interface of this functionality is separated into a table that lists all promotion file imports and into a parameter panel where the user can insert all data to configure a new promotion file import job. An import is always timer based what means the user has to define a time when the job is executed. To explain the functionality in detail, see Figure 4.32, “Scheduled Promotion File Import - Overview”:

1. **Name**: For better identification the user can provide a name for the import configuration in the regarding input field (1a) of the import job configuration.
2. **Server**: Shows the DocRepo server application the promotion file is supposed to be imported. This server can be chosen from the drop down list in the lower panel (2a).
3. **Import at**: The time at which the promotion file import should start is set in the input field (3a) in the lower panel automatically to the time the page is opened. In case the user wants to define another time he has to provide a date-time-combination in a Java standard conform pattern.
4. **Status**: Current status of the promotion file import job. The status [W] indicates that the import job is scheduled but has not been executed yet. During the execution of the job, the status [P] (in Progress) will be set. In case the job has been executed successfully the status [C] is displayed, in case of an error during the job execution status [F] occurs in the table.
5. **Imported at**: In case the job execution has been finished here the time is displayed when the job effectively has finished. In case of big promotion files a significant difference between the field “Import at” and the field *Imported at* might occur.
6. **View details**: Action button that displays in a separate page the details of the import job.

Figure 4.31. Promotion File Import Details

### Promotion File Import Details

Name	Test Import [Edit]	a
Server	http://localhost:12002/DBLayer	
User	uta	
Import at	2012-12-06 10:25	
Promotion File	Download PROM b	
Status	[C]	
Imported at	2012-12-06 10:25:05.531	
Message	Imported successfully	

- a. The user can edit the name of the promotion file import job.
  - b. Since the promotion file to upload is stored in DocRepo, the user has the option to download it later on.
7. **Delete import:** Action button that will remove the import job configuration and all execution data from the table and from the repository.

Figure 4.32. Scheduled Promotion File Import - Overview

### Scheduled Promotion File Import

The screenshot shows the 'Scheduled Promotion File Import' interface. At the top, there's a header bar with tabs like 'PROM Imports' and 'Promotion File Import'. Below is a table titled 'PROM Imports' with columns: Name, Server, Import at, Status, Imported at, and Actions. A single row is shown: 'Nightly deployment' with server 'http://localhost:12002/DBLayer', imported at '2013-01-28 23:59', status 'W', and actions buttons. Below the table is a section titled 'Add PROM Import' with fields: Name ('Resource deployment'), Import at ('2013-01-29 12:00'), PROM file ('Choose a file...'), DocRepo server ('http://localhost:12002/DBLayer'), Username ('uta'), and Password ('\*\*\*'). There are also 'Add' and 'Cancel' buttons. Numbered circles (1-11) are overlaid on various elements: 1 (Name), 2 (Import at), 3 (PROM file), 4 (DocRepo server), 5 (Username), 6 (Actions), 7 (Actions), 8 (Choose a file...), 9 (Password), 10 (Add), and 11 (Cancel).

8. **PROM file:** The user has to choose a promotion file that should be imported. The file chooser dialog will display a view of the local machine. When clicking the button "Add" the file immediately is uploaded to DocRepo that is used by DocAdmin Console.

**PROM file** postproc-addr-sheet.prom [[Clear](#)]

Once the file is uploaded it can be removed again by pressing the "Clear" link. Afterwards a new file can be uploaded.

9. **Username:** The DocRepo instance the promotion file should be imported to, will request an authentication. In this input field the user has to provide the username of a user that has the permission to execute promotion file imports.
10. **Password:** The password of the user that is used for the promotion file import job (see above number 9.).

11. **Add:** The “Add” button will save the configured promotion file import job into the DocRepo database. The scheduled job will be displayed in the overview table.

---

# Chapter

# 5

## Attachment Management

This chapter describes the UI for Attachment Management. This functionality allows to add Attachments to documents and envelopes during production with Post Processing. More information about the principles and concepts of Attachment Management can be found in the document "DocFamily Introduction". Their impact on document production is described in the document "DocBase Post Processing Guide".

The UI for Attachment Management is designed for the creation of Attachment Instructions.



### Attachment Instruction

An Attachment Instruction defines for a document in which case which Attachment will be added during Post Processing.

Since the creation of Attachment Instructions requires foregoing configuration and creation of objects the following flow of work can be sketched:

1. Creation of variables of type Custom Attributes in DocAdmin. See the description in "DocAdmin Online Help".
2. Create Attachment Conditions in DocWrite. See the description in "DocWrite User Guide".
3. Create channels and meta-information in the UI tab "Configuration" (Section 5.1, "Configuration").
4. Create documents in the UI tab "Documents" (Section 5.2, "Documents").
5. Create Attachments and versions in the UI tab "Attachments" (Section 5.3, "Attachments").
6. Create Attachment Instructions in the UI tab "Attachment Instructions" (Section 5.4, "Attachment Instruction").

The points 1 and 2 are not part of this documentation. The mentioned terms will be defined in the corresponding chapters.

### 5.1. Configuration

In the tab "Configuration" necessary base elements can be configured.

## 5.1.1. Attachment Meta-Information



### Attachment Meta-Information

Attachment Meta-Information are attributes that are available for Attachments.

**Figure 5.1. Configuration - Attachment Meta-Information overview**

The screenshot shows a table titled "Attachment Meta-Information" with four rows of data. The columns are labeled "Name", "Type", "Field type", and "Mandatory". The "Actions" column contains icons for search and edit. A header row indicates there are 1-4 / 4 items. A "Add new" link is located at the top left of the table area.

Name	Type	Field type	Mandatory	Actions
Mandatory	System	Checkbox	Yes	
Priority	System	Number	Yes	
Allow usage of previous version	System	Checkbox	Yes	
Allow usage of following version	System	Checkbox	Yes	

This table lists all available Attachment Meta-Information. It displays the main properties of the objects and the available Actions (1):



Opens a page, displaying the details of the Attachment Meta-Information.



Opens a page, where the details of the Attachment Meta-Information can be edited. This page corresponds to the page where Attachment Meta-Information can be created. It is explained in detail in Figure 5.2, "Configuration - Create Attachment Meta-Information".



Deletes an Attachment Meta-Information. This action can not be executed successfully as long as an Attachment has an active reference to this Attachment Meta-Information. In this case an error message is displayed on top of the overview table.

Clicking the link Add new (2) opens the page to create a new Attachment Meta-Information.

**Figure 5.2. Configuration - Create Attachment Meta-Information**

The form has the following fields and controls:

- Name:** Input field (1)
- Type:** Select dropdown set to "Custom" (2)
- Field type:** Select dropdown set to "Text" (3)
- Mandatory:** Checkbox (4)
- Create custom attribute:** Checkbox
- Initial text:** Input field (7a)
- Buttons:** "Submit" (5) and "Cancel" (6)

1. Display name, that will be visible in the overview table and later on in the page during Attachment creation.

2. Type of the Attachment Meta-Information. Currently two types are available:

- **System:** The system provides the following predefined Attachment Meta-Information, which are always to be defined for each Attachment that will be created. These Attachment Meta-Information can not be removed and also their editing options are limited:
    - **Mandatory:** Checking or unchecking this checkbox sets if the Attachment has to be added to a document in any case.
    - **Priority:** Number that defines the priority of the Attachment. This priority relates to factors like page limits, which must not be exceeded but could in case of adding an Attachment.
    - **Allow usage of previous version:** Attachments have time based version. In case for a certain time frame no version is defined, this checkbox allows to use the previous available version of an Attachment to be added to the document.
    - **Allow usage of following version:** Attachments have time based version. In case for a certain time frame no version is defined, this checkbox allows to use the following available version of an Attachment to be added to the document.
  - **Custom:** Customized Attachment Meta-Information that can be created individually. Attachment Meta-Information of this type can be removed in case they are not used in active Attachments.
3. Defines if the Attachment Meta-Information has to be defined mandatory for all Attachments, Attachment versions or Attachment Instructions.
4. If this checkbox is set, this Attachment Meta-Information will be stored in the resulting custom attribute in the runtime database of DocBase, presenting the Attachment information.
5. Creates the Attachment Meta-Information. Returns to the overview table where the new Attachment Meta-Information is displayed.
6. Cancels the Attachment Meta-Information creation and returns to the overview table.
7. For Attachment Meta-Information of Type “Custom” a field type can be defined. Depending on the selection in the dropdown list, further options can be adjusted (7a).

- **Text:** The Attachment Meta-Information is represented by a simple text field.

<b>Field type</b>	<input type="button" value="Text"/>
<b>Initial text</b>	<input type="text"/> (a)

a. An initial text can be defined here. When creating Attachments with this Attachment Meta-Information the text can be overwritten.

- **Checkbox:** A checkbox is displayed for this Attachment Meta-Information.

<b>Field type</b>	<input type="button" value="Checkbox"/>				
<b>Initial selection</b>	<input type="checkbox"/> (b)	<b>State</b>	<input checked="" type="checkbox"/> Checked (a)	<b>Custom attribute value</b>	<input type="text"/> (c)
	<input checked="" type="checkbox"/>		<input type="checkbox"/> Unchecked		

a. A checkbox usually covers two status - checked and unchecked.

- b. Defines, which of the available status the checkbox has initially during Attachment creation. In this column only one selection can be made.
- c. In case a Custom Attribute for this Attachment Meta-Information is saved in the DocBase runtime database, this value is stored corresponding to the selected status during Attachment creation.
- **Number:** The Attachment Meta-Information allows the insertion of a number. All numbers can be inserted that are covered by the range of the type defined in the Java class `java.lang.Number`.

Field type	Number
Initial value	<input type="text" value="0"/> (a)
Value between (incl.) min	<input type="text"/> (b)
(incl.) max	<input type="text"/> (c)

- a. An initial number can be defined here. When creating Attachments with this Attachment Meta-Information the number can be overwritten.
- b. Defines the minimal value for the Attachment Meta-Information. This value is optional and can define together with point c an input range.
- c. Defines the maximal value for the Attachment Meta-Information. This value is optional and can define together with point b an input range.
- **List:** An dropdown list is displayed for an Attachment Meta-Information of this type, where the user can choose one of various selection options.

Field type	List
Add new	
Initial selection	<input checked="" type="checkbox"/> (a)
Label	<input type="text" value="b"/> (b)
Custom attribute value	<input type="text"/> (c)
	<input type="button" value="d"/> <input type="button" value="e"/> <input type="button" value="x"/>
	<input type="button" value="d"/> <input type="button" value="e"/> <input type="button" value="x"/>
	<input type="button" value="d"/> <input type="button" value="e"/> <input type="button" value="x"/>

- a. Defines, which of the available selection options is chosen initially during Attachment creation. In this column only one selection can be made.
- b. Label, that will be displayed in the dropdown list.
- c. Value that is saved for the selection option in the DocBase runtime database in case a Custom Attribute for this Attachment Meta-Information is stored in the DocBase runtime database.
- d. Buttons to order the selection options, how they are displayed in the dropdown list. Moves the table entry either up or down, like the option will move up or down in the list.
- e. Removes the table entry and the corresponding selection option.
- **Date:** In case the Attachment Meta-Information is of this type an date input field will be displayed during Attachment creation.

Field type	Date
Initial date	<input type="text"/>
Date valid	<b>from</b> <input type="text"/> <b>to</b> <input type="text"/>

- a. An initial date can be defined here. When creating Attachments with this Attachment Meta-Information the date can be overwritten.
- b. Defines the earliest possible date for the Attachment Meta-Information. This date is optional and can define together with point c a time frame.
- c. Defines the latest possible date for the Attachment Meta-Information. This date is optional and can define together with point b a time frame.

## 5.1.2. Channels



### Channel

A Channel is a target where a Post Processing stream has to be delivered. It has to match a Custom Attribute from the DocBase runtime database.

Figure 5.3. Configuration - Channels overview

The screenshot shows the 'Attachment Management' interface with the 'Configuration' tab selected. Below it, the 'Channels' tab is active. A table lists two channels: 'shipping' and 'archive'. Each channel row includes a 'Name' column, an 'Actions' column with three icons (search, edit, delete), and a small table showing '1-2 / 2' items. Navigation arrows are present at the top and bottom of the main table.

This table lists all Channels that relate to delivery targets for documents with Attachments. It displays the main properties of the objects and the available Actions (1):



Opens a page, displaying the details of the Channel.



Opens a page, where the details of the Channel can be edited. This page corresponds to the page where Channels can be created. It is explained in detail in Figure 5.4, "Configuration - Create Channel".



Deletes a Channel. This action can not be executed successfully as long as an Attachment has an active reference to this Channel. In this case an error message is displayed on top of the overview table.

Clicking the link Add new (2) opens the page to create a new Channel.

Figure 5.4. Configuration - Create Channel

**Create Channel Information**

Name  (1)

Custom attribute

Key  (2)

Value  (3)

(4) Submit (5) Cancel

1. Display name, that will be visible in the overview table and later on in the page during Attachment creation.
2. The name of a Custom Attribute, that exists on runtime, has to be provided here. This Custom Attribute then will be looked up in the runtime database of DocBase.
3. The string inside of the field should match a possible value of the Custom Attribute mentioned in point 2. During runtime this value will be compared with the content of the Custom Attribute in the runtime database of DocBase.
4. Creates the Channel. Returns to the overview table where the new Channel is displayed.
5. Cancels the Channel creation and returns to the overview table.

## 5.2. Documents



### Document Reference

A Document Reference defines to which object an Attachment has to be added.

Figure 5.5. Documents - Overview

**Attachment Management**

Attachments				Documents	Configuration																								
<input type="checkbox"/> Document Filter [show/hide] (3)																													
Add new (2) <a href="#">Play all elements</a>   <a href="#">Display only selected elements</a>   <a href="#">Bulk operations</a> ▾																													
<table border="1"> <thead> <tr> <th colspan="2"></th> <th colspan="2">1-1 / 1</th> <th colspan="2"></th> </tr> <tr> <th colspan="2"></th> <th>Name</th> <th>Type</th> <th colspan="2">Actions (1)</th> </tr> </thead> <tbody> <tr> <td colspan="2"><input type="checkbox"/></td> <td>Sample1</td> <td>DocDesign template</td> <td></td> <td></td> </tr> <tr> <td colspan="2"></td> <td colspan="2">1-1 / 1</td> <td></td> <td></td> </tr> </tbody> </table>								1-1 / 1						Name	Type	Actions (1)		<input type="checkbox"/>		Sample1	DocDesign template					1-1 / 1			
		1-1 / 1																											
		Name	Type	Actions (1)																									
<input type="checkbox"/>		Sample1	DocDesign template																										
		1-1 / 1																											

This table lists all Document References. It displays the main properties of the objects and the available Actions (1):



Opens a page, displaying the details of the Document Reference.



Opens a page, where the details of the Document Reference can be edited. This page corresponds to the page where Document References can be created. It is explained in detail in Figure 5.6, "Documents - Create Document Reference".



Looks up all Attachment Instructions that are built up on this Document Reference.



Deletes a Document Reference. This action can not be executed successfully as long as an Attachment Instruction; has an active reference to this Document Reference. In this case an error message is displayed on top of the overview table.

For this table a filter panel can be displayed (3) and clicking the link Add new (2) opens the page to create a new Document Reference.

**Figure 5.6. Documents - Create Document Reference**

**Create Document Reference**

Name	<input type="text"/>	1
Type	<input type="text" value="DocDesign template"/>	4
Repository object	<input type="text"/>	4a
Repository object ID	<input type="text"/>	
<input type="button" value="Submit"/> <input type="button" value="Cancel"/> 3		

1. Display name, that will be visible in the overview table and later on in the page during Attachment Instruction; creation.
2. Creates the Document Reference. Returns to the overview table where the new Document Reference is displayed.
3. Cancels the Document Reference creation and returns to the overview table.
4. To various object types am Attachment can be added during Post Processing. This type will be selected for the Document Reference. Depending on the type, further input fields or selection options appear (4a).
  - DocDesign template, Textsystem template, Textsystem template with copies, Textsystem document, Event definition: Selecting one of these types requires only the selection of the corresponding resource from the repository.

Type	<input type="text" value="DocDesign template"/>
Repository object	<input type="text"/> a
Repository object ID	<input type="text"/> b c

The button (a) opens a dialog to choose a resource of the selected type from the repository. The dialog will display the structure of the repository but only objects of the selected type. After selecting the resource the fields b and c are filled in automatically.

- Custom Attribute: When defining a Document Reference with this type an Attachment will be added to each object that has a reference to the corresponding Custom Attribute (a) with the inserted value (b).

Type

**Custom attribute**

Key	<input type="text"/>	(a)
Value	<input type="text"/>	(b)

- **Document in event definition:** For this type of Document Reference it is necessary to identify a dedicated document inside of an event. Since events are runtime data driven, in this case during Post Processing the event is identified by a Custom Attribute that depends on the runtime data, as well as the document.

For event and document each, the Custom Attribute key (a) and the value to match (b) has to be inserted. The checkbox (c) ensures that an Attachment is added only once per event in case there exist multiple documents that match the document-identifying Custom Attribute.

Type

**Custom attribute identifying event**

Key	<input type="text"/>	(a)
Value	<input type="text"/>	(b)

**Custom attribute identifying document**

Key	<input type="text"/>	(a)
Value	<input type="text"/>	(b)

Remove duplicate attachments on event level  (c)

## 5.3. Attachments

Attachments are the objects that will be added to documents during Post Processing. Per attachment at least one version has to be created.

Figure 5.7. Attachments overview

Attachment Management													
Attachment Instructions		Attachments		Documents		Configuration							
<input type="checkbox"/> Attachment Filter [show/hide] (3)													
Add new (2) <a href="#">play all elements</a>   <a href="#">Display only selected elements</a>   <a href="#">Bulk operations</a> ▾													
◀◀			1-1 / 1				▶▶						
Name	Type	Format	Mandatory	Priority	Notes	Versions	Actions (1)						
<input type="checkbox"/> FlyerProductsPrivatePension	Meta-Information	Custom attribute	No	0	1								
◀◀		1-1 / 1					▶▶						

This table lists all Attachments. It displays the main properties of the objects and the available Actions (1):

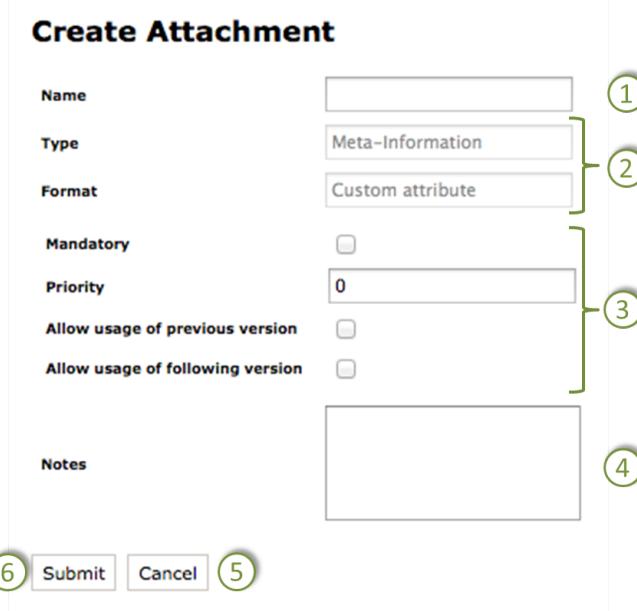


Opens a page, displaying the details of the Attachment.

-  Opens a page, where the details of the Attachment can be edited. This page corresponds to the page where Attachments can be created. It is explained in detail in Figure 5.8, "Create Attachments".
-  Looks up all Attachment Instructions that are built up on this Attachment.
-  Deletes an Attachment. This action can not be executed successfully as long as an Attachment Instruction; has an active reference to this Attachment. In this case an error message is displayed on top of the overview table.

For this table a filter panel can be displayed (3) and clicking the link [Add new](#) (2) opens the page to create a new Attachment.

**Figure 5.8. Create Attachments**



The screenshot shows the 'Create Attachment' form. At the top is the title 'Create Attachment'. Below it are several input fields and buttons:

- Name:** An input field with a green circle containing the number 1 next to it.
- Type:** A dropdown menu showing 'Meta-Information' with a green circle containing the number 2 next to it.
- Format:** A dropdown menu showing 'Custom attribute' with a green circle containing the number 2 next to it.
- Mandatory:** A checkbox.
- Priority:** An input field containing '0' with a green circle containing the number 3 next to it.
- Allow usage of previous version:** A checkbox.
- Allow usage of following version:** A checkbox.
- Notes:** A large text area with a green circle containing the number 4 next to it.
- Buttons:** At the bottom are two buttons: 'Submit' (with a green circle containing the number 6 next to it) and 'Cancel'.

1. Display name, that will be visible in the overview table and later on in the page during Attachment Instruction creation.
2. Since the current implementation covers only the attachment of meta-information in the form of Custom Attributes the fields "Type" and "Format" are read-only.
3. This group shows the Attachment Meta-Information that are provided by the system and the ones that were configured before (see Section 5.1.1, "Attachment Meta-Information").
4. Field that allows to enter notes, e.g. for description. They will be displayed in the overview table.
5. Cancels the creation of a new Attachment and returns to the overview table.
6. Clicking this button creates the Attachment. The button label changes and the screen is extended by an overview table (1), that allows the creation of multiple versions of an Attachment (2).

Attachment Versions						
Add new	(2)					
1-3 / 3						
Version	Valid from	Valid to	Allow to use previous version	Allow to use next version	Name	Actions
1	2014-01-21 15:40:00	2014-01-31 12:00:00	Yes	Yes	FlyerProductsPrivatePension2014Jan	
2	2014-02-01 12:00:00	2014-02-28 12:00:00	Yes	Yes	FlyerProductsPrivatePension2014Feb	
3	2014-03-01 12:00:00	2014-03-31 12:00:00	No	No	FlyerProductsPrivatePension2014Mar	

The versioning of Attachments is time related. A version is valid for a certain time period. Adding a new Attachment version opens the corresponding input form.

Figure 5.9. Create Attachment Version - Main

### Create Attachment Version

Main      Override Attachment

Version	2
Name	(1)
Valid from	(2)
Meta-Information	
Key	(3) VALID-FLYPRODPRIVPEN
Value	(4)
Sheet equivalent	(5) 1
Notes	(6) <input type="text"/>
<input type="button" value="Submit"/> (7) <input type="button" value="Cancel"/>	

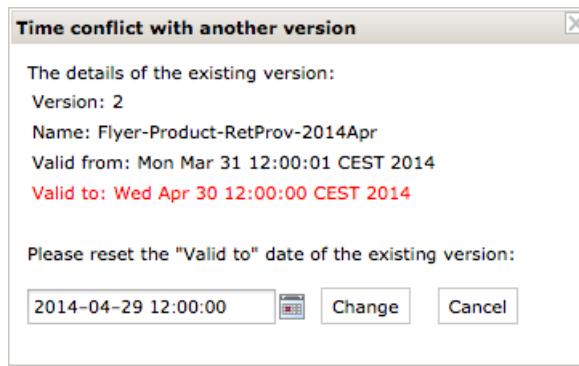
1. Display name, that will be visible in the overview table.
2. Here a time frame can be defined, when the Attachment version is active. In case only the "Valid from" value is set, the Attachment version is valid from the defined point in time for an unlimited period.

## Caution



In general versions can not be defined with overlapping validity times. Depending on how many other versions the newly created one covers, either a validation message is displayed or a dialog. The message is displayed in case more than one other Attachment version is affected and will state which versions need to be checked.

In case only one Attachment version collides with the new one, a dialog is opened displaying the details of the existing version and an input field for the correction of the interfering date. This field is pre-filled in with the next/ previous date, that would not conflict with the time period the new Attachment version should be valid for.



3. This Meta-Information will be saved in the runtime database of DocBase. It can contain any information that is required to identify the correct Attachment and its version by an external system. See further instructions how to integrate Attachment Management with external systems in the document “DocBase Post Processing Guide”.
4. An Attachment will consist of document pages. For the calculation of envelope and stream sizes in Post Processing the sheet number of an Attachment has to be added here.
5. In this field description notes for the current Attachment version can be added.
6. Takes over all changes made at the Attachment version and returns to the Attachment modification page with the Attachment version overview table.
7. Discards all changes and returns to the Attachment modification page with the Attachment version overview table.

In the tab “Override Attachment” (Figure 5.10, “Create Attachment Version - Override Attachment”) for this particular version the standard Attachment Meta-Information, defined on the Attachment can be overwritten. The area lists read-only in the right column the standard configuration for the Attachment Meta-Information (2) and displays checkboxen in the left column (1). Enabling a checkbox renders the standard configuration editable. All changes are taken over by clicking the button “Submit” (3) or discarded by clicking the button “Cancel” (4). In both cases the current view is closed and the Attachment creation page is displayed again.

**Figure 5.10. Create Attachment Version - Override Attachment**

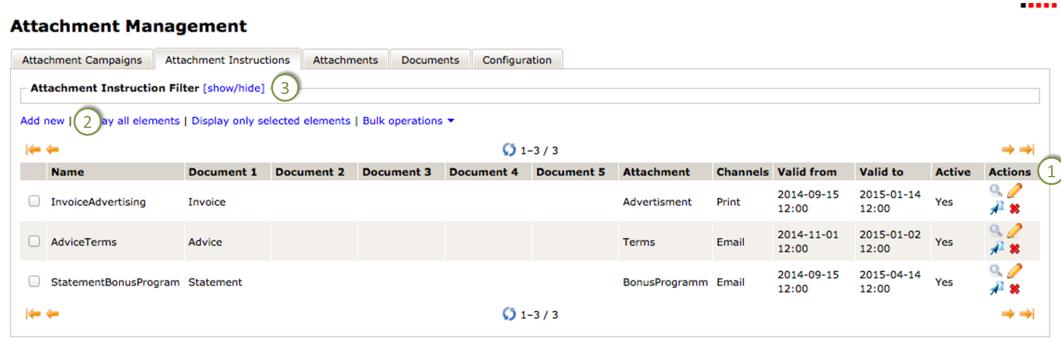
**Create Attachment Version**

Main	Override Attachment																			
<table border="1"> <thead> <tr> <th colspan="2">Override setting from attachment</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>(1)</td> <td><input checked="" type="checkbox"/></td> <td>(2)</td> </tr> <tr> <td><input type="checkbox"/></td> <td></td> <td>0</td> <td></td> </tr> <tr> <td><input type="checkbox"/></td> <td></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td><input type="checkbox"/></td> <td></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> </tbody> </table>		Override setting from attachment		Value	<input type="checkbox"/>	(1)	<input checked="" type="checkbox"/>	(2)	<input type="checkbox"/>		0		<input type="checkbox"/>		<input checked="" type="checkbox"/>		<input type="checkbox"/>		<input checked="" type="checkbox"/>	
Override setting from attachment		Value																		
<input type="checkbox"/>	(1)	<input checked="" type="checkbox"/>	(2)																	
<input type="checkbox"/>		0																		
<input type="checkbox"/>		<input checked="" type="checkbox"/>																		
<input type="checkbox"/>		<input checked="" type="checkbox"/>																		
(3)	Submit	Cancel	(4)																	

## 5.4. Attachment Instruction

The construction of an Attachment Instruction puts together all resources and objects that have been created so far, like Channels, Attachment Meta-Information, Document References, Attachments and Attachment Conditions. It can be used in Attachment Campaigns.

**Figure 5.11. Attachment Instructions overview**



The screenshot shows a table titled "Attachment Management" with a header row containing tabs: Attachment Campaigns, Attachment Instructions, Attachments, Documents, and Configuration. The "Attachment Instructions" tab is selected. Below the header is a search/filter bar with a magnifying glass icon and the text "Attachment Instruction Filter [show/hide]". To the right of the search bar is a circled number "3". Below the search bar are buttons for "Add new" (circled "2"), "Show all elements", "Display only selected elements", and "Bulk operations". The main area is a data grid with the following columns: Name, Document 1, Document 2, Document 3, Document 4, Document 5, Attachment, Channels, Valid from, Valid to, Active, and Actions. There are three rows of data:

Name	Document 1	Document 2	Document 3	Document 4	Document 5	Attachment	Channels	Valid from	Valid to	Active	Actions
InvoiceAdvertising	Invoice					Advertisement	Print	2014-09-15 12:00	2015-01-14 12:00	Yes	
AdviceTerms	Advice					Terms	Email	2014-11-01 12:00	2015-01-02 12:00	Yes	
StatementBonusProgram	Statement					BonusProgramm	Email	2014-09-15 12:00	2015-04-14 12:00	Yes	

At the bottom of the grid are navigation icons for first, previous, next, last, and a page number indicator "1-3 / 3".

This table lists all Attachment Instructions. It displays the main properties of the objects and the available Actions (1):



Opens a page, displaying the details of the Attachment Instruction.



Opens a page, where the details of the Attachment Instruction can be edited. This page corresponds to the page where Attachment Instructions can be created. It is explained in detail in Figure 5.12, "Modify Attachment Instruction - Main".



Looks up all Attachment Campaigns that have assigned this Attachment Instruction.



Deletes an Attachment Instruction.

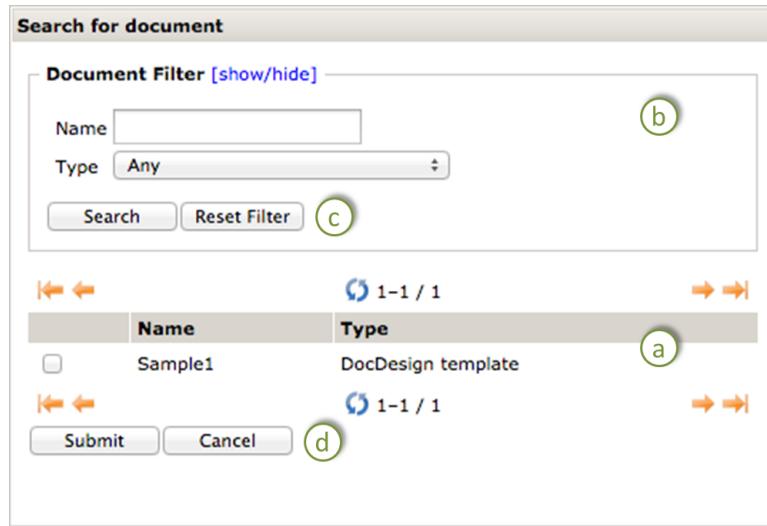
For this table a filter panel can be displayed (3) and clicking the link Add new (2) opens the page to create a new Attachment Instruction.

Figure 5.12. Modify Attachment Instruction - Main

**Modify Attachment Instruction**

Main	Override Attachment (version)
Name ①	StatementFlyerProductsPrivate
Document 1 ②	statement-A1-portrait-WOS
Document 2	<input type="text"/>
Document 3	<input type="text"/>
Document 4	<input type="text"/>
Document 5	<input type="text"/>
Documents connected by	<input checked="" type="radio"/> AND <input type="radio"/> OR ③
Attachment ④	FlyerProductsPrivatePension
Channels ⑤	archive  shipping
Condition 1 ⑥	docType-WOS
Condition 2	<input type="text"/>
Condition 3	<input type="text"/>
Condition 4	<input type="text"/>
Condition 5	<input type="text"/>
Conditions connected by	<input checked="" type="radio"/> AND <input type="radio"/> OR ⑦
Valid from ⑧	2014-01-21 15:54:00  to 2014-01-22 12:00:00
Active ⑨	<input checked="" type="radio"/> Yes <input type="radio"/> No
<b>⑩ Modify    ⑪ Cancel</b>	

1. Display name, that will be visible in the Attachment Instruction overview table.
2. Up to 5 Document References can be selected for which the user creates the Attachment Instruction. Clicking an magnifying glass icon opens a dialog with a view on the Document References table.
  - a. Table, which lists the available Document References.
  - b. Filter panel that allows to limit the result set displayed in the table.
  - c. Button set that applies the configured filter or resets the Document Reference search.
  - d. Button set to either take over the chosen Document Reference ("Submit") or to discard any selections ("Cancel"). In any case the view returns to the Attachment Instruction creation page.



3. Defines the logical relation between the Document References, in case more than one has been selected.

In case the **OR** relation has been chosen, a document which is checked by the Attachment Instruction has to match only one of the selected Document References. Still, the process will check all Document References since one of the matching ones might contain the information to remove duplicates. As soon as this is set for one of the matching Document References, duplicated Attachments will be removed in from an event.

In case the **AND** relation has been chosen, a document which is checked by the Attachment Instruction has to match all selected Document References. This induces some restriction for the combination of selected Document References, since e.g. two Document References of type "DocDesign template" would create an unrealisable Attachment Instruction because a document can base only on one DocDesign Template

The following table gives an overview which types can be combined:

AND	Textsystem template	Textsystem template with copies	Textsystem document	DocDesign template	Event definition	Document in event definition	Custom attribute
Textsystem template		X	X	X			X
Textsystem template with copies	X		X	X			X
Textsystem document	X	X		X			X
DocDesign template	X	X	X				X
Event definition						X	X
Document in event definition					X		X
Custom attribute	X	X	X	X			X

4. Like for Document References in point 2, via an separate dialog an Attachment has to be selected here.
5. In this control at least one Channel has to be selected. It is possible to select multiple or even all available Channels.
6. Up to 5 Attachment Conditions can be chosen from the repository by clicking the folder icon. The opening dialog limits the displayed objects to folders and Attachment Conditions. The selection of Attachment Conditions is optional, an Attachment Instruction can be created without.
7. Defines the logical relation between the Attachment Conditions, in case more than one has been selected.
8. By selecting a "Valid from" date and a "to" date, a time frame can be selected, when the Attachment Instruction can be active. If only the "Valid from" date is defined, the Attachment Instruction is activated from the given point in time for an undefined period.
9. Activates or desactivates the Attachment Instruction. This property overrules the time based activation of point 8.
10. Takes over all changes and returns to the overview table of Attachment Instructions.
11. Discards all changes and returns to the overview table of Attachment Instructions.

In the tab “Override Attachment (version)” (Figure 5.13, “Modify Attachment Instruction - Override Attachment (version)”) for this Attachment Instruction the standard Attachment Meta-Information, defined on the chosen Attachment can be overwritten. Selecting a particular version of the Attachment version (3), the area lists read-only in the right column the standard configuration for the Attachment Meta-Information (2) and displays checkboxen in the left column (1). Enabling a checkbox renders the standard configuration editable. All changes are taken over by clicking the button “Modify” (4) or discarded by clicking the button “Cancel” (5). In both cases the current view is closed and the Attachment Instruction overview table is displayed again.

**Figure 5.13. Modify Attachment Instruction - Override Attachment (version)**

Override setting from attachment (version)		Value
(3)	1	
Mandatory	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Priority	<input type="checkbox"/>	0
Allow usage of previous version	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Allow usage of following version	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

(4) Modify    (5) Cancel

## 5.5. Attachment Campaigns

Attachment Campaigns evaluate that for a set of Attachment Instructions the number of active Attachment versions is not exceeded.

**Figure 5.14. Attachment Campaigns overview**

The screenshot shows a table titled "Attachment Campaigns" with the following columns: Name, Threshold, Attachment instructions, Valid from (min), Valid to (max), and Actions. There is one entry: "MonthlyCampaign" with a threshold of 2, attachment instructions of 3, valid from 2014-09-15 12:00, and valid to 2015-04-14 12:00. The "Actions" column contains icons for search, edit, and delete. A filter panel is visible at the top right, and a toolbar with buttons for "Add new" (circled 2), "Attachment Campaign Filter [show/hide]" (circled 3), and other operations.

This table lists all Attachment Campaigns. It displays the main properties of the objects and the available Actions (1):



Opens a page, displaying the details of the Attachment Campaign;.



Opens a page, where the details of the Attachment Campaign can be edited. This page is similar to the page where Attachment Campaigns can be created. It is explained in detail in Figure 5.15, "Create Attachment Campaign - Main".



Deletes an Attachment Campaign.

For this table, clicking the link Add new (2) opens the page to create a new Attachment Campaign and a filter panel can be displayed (3).

When creating a new Attachment Campaign the following page is opened:

**Figure 5.15. Create Attachment Campaign - Main**

The form has fields for Name (1), Threshold (max. attachments) (2), Notes (3), and buttons for Submit (4) and Cancel (5). The "Name" field contains "MonthlyCampaign", and the "Threshold" field contains "2".

1. Display name, that will be visible in the overview table of Attachment Campaigns.
2. Maximum number of different Attachments, that are allowed to be added during document production.
3. Description notes field for the current Attachment version can be added.
4. Clicking this button creates the Attachment Campaign. The button label changes and the screen is extended by an overview table (1), that allows the assignment of multiple Attachment Instructions to the Attachment Campaign (2). For this table a filter panel can be displayed (3):

**Figure 5.16. Table of Attachment Instructions**

The screenshot shows a table titled "Attachment Instruction Filter [show/hide]". The table has columns: Name, Valid from, Valid to, Active, and Actions. There are three rows of data:

Name	Valid from	Valid to	Active	Actions
InvoiceAdvertising	2014-09-15 12:00	2015-01-14 12:00	Yes	
AdviceTerms	2014-11-01 12:00	2014-11-30 12:00	Yes	
StatementBonusProgram	2014-09-15 12:00	2015-04-14 12:00	Yes	

On the left side of the Attachment Campaign page, besides the "Submit" and "Cancel" buttons, a new "Validate" button will be displayed (see Figure 5.15, "Create Attachment Campaign - Main").

5. Discards all changes and returns to the overview table.

At least one Attachment Instruction has to be assigned to an Attachment Campaign. Clicking the link "Assign new" opens a dialog with the list of available Attachment Instructions (see Figure 5.16, "Table of Attachment Instructions", point 4).

**Figure 5.17. Assign Attachment Instruction to Attachment Campaign**

The screenshot shows a dialog titled "Search for Attachment Instructions". It contains a search filter with fields: Name, Document, Attachment (circled with a green circle), Valid from, Valid to, and Active. Below the filter is a table with columns: Name, Document 1, Document 2, Document 3, Document 4, Document 5, Attachment, Channels, Valid from, Valid to, and Active. The table has three rows of data:

Name	Document 1	Document 2	Document 3	Document 4	Document 5	Attachment	Channels	Valid from	Valid to	Active
<input checked="" type="checkbox"/> InvoiceAdvertising	Invoice					Advertisement	Print	2014-09-15 12:00	2015-01-14 12:00	Yes
<input type="checkbox"/> AdviceTerms	Advice					Terms	Email	2014-11-01 12:00	2014-11-30 12:00	Yes
<input type="checkbox"/> StatementBonusProgram	Statement					BonusProgramm	Email	2014-09-15 12:00	2015-04-14 12:00	Yes

1. Table with all existing Attachment Instructions.
2. Search filter to limit the displayed Attachment Instructions. The filter criteria reflect the properties that are configured for each Attachment Instruction.
3. Check boxes to select the Attachment Instructions that should be assigned to the Attachment Campaign. For Attachment Instructions that are already assigned to the Attachment Campaign the check box can not be ticked. By clicking the "Submit" button the Attachment Instructions are assigned and the dialog is closed.

**Validation.** The validation of the Attachment Campaign takes place when clicking on the "Validate" button in the main Attachment Campaign page (Figure 5.15, "Create Attachment Campaign - Main"). A new page as in the following screenshot is opened:

Figure 5.18. Validate Attachment Campaign

**Validate Campaign 'MonthlyCampaign'**

**Conflicts found for the following Attachment Instructions:**

Period from 2014-11-01 12:00 to 2014-11-30 12:00 (1)

Name	Valid from	Valid to	Actions
AdviceTerms	2014-11-01 12:00	2015-01-02 12:00	
StatementBonusProgram	2014-09-15 12:00	2015-04-14 12:00	
InvoiceAdvertising	2014-09-15 12:00	2015-01-14 12:00	

Period from 2014-12-24 12:00 to 2014-12-31 12:00

Name	Valid from	Valid to	Actions
AdviceTerms	2014-11-01 12:00	2015-01-02 12:00	
StatementBonusProgram	2014-09-15 12:00	2015-04-14 12:00	
InvoiceAdvertising	2014-09-15 12:00	2015-01-14 12:00	

(5) Back Re-validate (4)

1. Period for which the active Attachment versions of the Attachment Instructions assigned to the Attachment Campaign exceeds the maximal allowed number of Attachments.
2. List of Attachment Instructions that have Attachment versions configured, which are active in the displayed period.
3. Actions to edit an Attachment Instruction. Clicking the "Edit details" button, the "Modify Attachment Instruction" page will be opened (Figure 5.12, "Modify Attachment Instruction - Main"). Leaving the page by the "Modify" or "Cancel" button will return to the "Validate" page.
4. After changing the Attachment Instructions the Attachment Campaign can be re-validated. This is not executed automatically.
5. The page can be left by clicking the "Back" button. The user will be led back to the "Modify Attachment Campaign" page.



### Note

The validation is a check that is manually done. Please note that Attachment Campaigns, that are not valid can still be saved.