

ReportServer

Administrator's Guide



ReportServer Administrator's Guide

Thorsten J. Krause Arno Mittelbach

datenwerke, 2015 http://www.datenwerke.net/ http://www.reportserver.net/



Copyright 2015 datenwerke – Jan Albrecht. All rights reserved.

This document is protected by copyright. It may not be distributed or reproduced in whole or in part for any purpose without written permission of datenwerke. The information included in this publication can be changed at any time without prior notice.

All rights reserved.

Contents

Co	ontents	I
1	Preamble	3
2	First Steps	7
	2.1 Configuration and installation	7
	2.2 Login	7
	2.3 Creating a data source	9
	2.4 Creating your First Report	9
	2.5 Importing a Graphical Report	10
	2.6 Creating users	11
	2.7 Terminal and FileServer	12
3	User and Permission Management	15
	3.1 The User Tree	15
	3.2 Permission Management	16
4	Data Sources	23
	4.1 Relational Databases	23
	4.2 Storage of Database Passwords	24
	4.3 The Internal Data Base	24
	4.4 Datasource Pool	25
	4.5 CSV Lists	25
	4.6 Script Data Sources	27
	4.7 BIRT Report Data Source	27
	4.8 Mondrian Datasource	28
	4.9 Data Source Bundle	29
	4.10 Configuration of a Standard Data Source	29
5	File System	33
	5.1 Configuration Files	34
	5.2 Filing of Scripts	34
	5.3 Accessing Resources by URL	34

6	Report Management																	37
	6.1 Fundamentals													•				38
	6.2 The Dynamic List													•				40
	6.3 Working with Parameters													• •				44
	6.4 JasperReports													•				56
	6.5 Eclipse Birt													•				57
	6.6 SAP Crystal Reports																	58
	6.7 Saiku / Mondrian Reports													•				59
	6.8 JXLS Reports																	60
	6.9 Script Reports																	61
	6.10 Grid Editor Reports																	64
	6.11 Executing Reports via the URL													•				66
	6.12 Report Metadata													•				69
	6.13 Drill Down Reports													• •				70
7	Global Constants																	73
8	User Variables																	75 75
	8.1 Defining User variables	•••		•	• •	•	• •	• •	• •	·	• •	·	•	• •	•••	·	• •	15
	8.2 Allocating User Variables	•••		•	• •	•	• •	• •	• •	·	• •	·	•	• •	•••	·	• •	70
	6.5 Using User variables in Reports	•••		·	• •	·	• •	• •	• •	·	• •	•	•	• •	•••	•	• •	70
9	Import and Export																	79
	9.1 Exporting													•				79
	9.2 Importing													• •				79
10	Scheduling of Reports																	83
10	10.1 Technical Backgrounds to Scheduler	· lot	5															83
	10.2 Filtering by the Status of a Job	500		•	• •	•	• •	• •	• •	·	• •	•	•	• •	• •	•	• •	84
	10.3 Notifications			•	• •	•	• •	• •	• •	•	•••	•	•	• •	•••	•	• •	84
	10.4 Conditional Scheduling	•••	•••	•	• •	•	• •	• •	• •	·	• •	•	•	• •	•••	•	• •	84
	10.5 Enhanced Scheduling			•	• •	•	• •	• •		•	• •	•	•	• •	•••	•	• •	85
				•		•		• •	• •	•	• •	•	•	• •		•		
11	Terminal																	87
	11.1 Using the Terminal	•••		•	• •	•	• •	• •	• •	·	• •	·	•	• •	•••	·	• •	δ <i>ι</i> 00
	11.2 Anging Aliance	•••		•	• •	•	• •	• •	• •	·	• •	·	•	• •	•••	·	• •	88
	11.3 Assigning Allases			·	• •	·	• •	• •	• •	·	• •	·	•	• •		·	• •	90
				•	• •	•	• •	• •		•	•••	·	•	• •	•••	·	• •	91
12	ReportServer Scripting																	93
	12.1 A first Hello World	•••		•	• •	•		• •	• •	·	• •	•	•	• •	•••	·	• •	94
	12.2 How to Handle Errors	• • •		•	• •	•	• •	• •	• •	·	• •	·	·	• •		·	• •	94
	12.3 Cancelling Scripts	•••		·	• •	·	• •	• •	• •	·	• •	•	•	• •	•••	·	• •	95
	12.4 Administrative Scripts	•••		·	• •	·	• •	• •	• •	·	• •	•	•	• •	•••	·	• •	95
	12.5 Changing the Data Model	•••		·	• •	·	• •	• •	• •	·	• •	•	•	• •	•••	·	• •	98
	12.0 Monitoring Script Executions	 -I		·	• •	·	• •	• •	• •	·	•••	·	•	• •		·	• •	98
	12.7 Starting Scripts in the Server Thread	a.,		·	• •	·	• •	• •	• •	·	• •	•	•	• •	•••	·	• •	100
	12.8 Enhancing ReportServer with Scripts	5.				•					• •	•						100

	16.2 Logging	119
16	Maintenance 16.1 Testing User Specific Settings (su) 16.2 Logging	119 119
15	SFTP Server	117
14	Dashboards and Dadgets14.1 Static HTML Dadgets14.2 Embedding Dashboards	111 112 114
13	Terminal Commands	103
	12.9 Scheduling of Scripts	101 101

Chapter 1

Preamble

Business Intelligence

Business Intelligence (BI) describes the ability to jointly analyze all of a company's data, distilling relevant information to be used to foster better business decisions. The foundation of any BI solution is the careful preprocessing of existing data, for example, in a data warehouse.

ReportServer acts as the gateway between end-users and the collected data, allowing users to efficiently access and analyze the available data. From camera-ready evaluations to fine-grained ad-hoc reporting; ReportServer provides you with the tools to support your daily work.

Target Audience

This document is designed for future administrators of ReportServer. Administrators are the persons who are entrusted with the daily operation of the reporting platform. Operation means for example to create and manage users and user permissions, to organize reports as well as to monitor the complete system. Separate manuals and instructions illustrate the various aspects of ReportServer. You can purchase the various guides from http://www.reportserver.net.

- **ReportServer Configuration Guide:** Describes the installation of ReportServer as well as the basic configuration options. The configuration guide is freely available.
- **ReportServer User Guide:** The user guide describes ReportServer from the point of view of the ultimate user. It includes an in-depth coverage of dynamic lists (ReportServer's adhoc reporting solution), execution of reports, scheduling of reports, and much more.
- **ReportServer Administrator Guide:** The administrator guide describes ReportServer from the point of view of administrators that are tasked with maintaining the daily operation of the reporting platform including the development of reports, managing users and permissions, monitoring the system state, and much more.
- **ReportServer Scripting Guide:** The ReportServer scripting guide covers the scripting capabilities of ReportServer which can be used for building complex reports as well as for extending the functionality of ReportServer or performing critical maintenance tasks. It extends the introduction to these topics given in the administrator guide.

In addition to the available manuals we regularly publish tutorials and best practices on our blog http://blog.datenwerke.net.

System State

We proceed on the assumption that the basic configuration has been completed as de- scribed in the installation and configuration instructions. All examples given in this book are based on the available demo data.

ReportServer Demo Content

The demo data used in this book are part of the standard delivery scope of ReportServer and can be automatically loaded when starting the system. Additionally we provide a demo content package that preconfigures ReportServer for a fictitious company called "1 to 87". This includes setting up various users including permissions, TeamSpaces where these users have access and can collaborate as well as a number of demo reports.

Loading the demo warehouse. ReportServer comes with the option to install a demo database, which is the base for the demo report shipped with ReportServer. The demo data will be installed in an internal database (we will later see the significance of this internal database). To load the demo data on startup change the configuration file /etc/datasources/internaldb.cf to include the tag <installdemodata>true</installdemodata>. You will find this configuration file, if you log into ReportServer as root, go to the administration module.¹ For this choose Administration from the module bar at the top of the screen, and then choose File Server from the aspects on the left. Navigate to file /etc/datasources/internaldb.cf in the tree. To change the file choose the tab *edit* on the bottom. A sample configuration could, for example, look like:

```
<configuration>
<internaldb>
<location>dbtmp</location>
<encryption>
<disable>false</disable>
<password>SecretPassphrase</password>
</encryption>
<installdemodata>true</installdemodata>
</configuration>
```

Note that this change will only take effect after a restart of ReportServer.

Loading the demo content. To install the demo content, log into ReportServer as root.After logging in, press CTRL+ALT+T to open the terminal. The terminal is powerful tool to administer ReportServer. We will see various use cases for it throughout this manual. Type

pkg list

and confirm the command by pressing enter. You should see

¹Look at the next section "First Steps" for an introduction to ReportServer and a guide to logging in, the user interface and the various interfaces.

baseconfig-RS2.1.1-5517-2013-11-06-18-02-40.zip demobuilder-RS2.1.1-5517-2013-11-06-18-02-40.zip reportserver\$

The example content is supplied as a ReportServer package, which is basically a groovy script with some additional files. The pkg list terminal command lists all the available packages. In the example above, there were to packages. The *baseconfig* package is automatically installed, whenever you conduct a fresh install of ReportServer. It installs the default configuration files in the etc directory of ReportServer's internal file server. The *demobuilder* package contains the example data, we are going to install. The precise filename may differ from the one pictured here, depending on your ReportServer version.

Warning, the installation of the demo content, will remove any existing content. To install the example data use the following command:

pkg install -d <packagename>

where packagename denotes the name of the package as returned by the pkg list command. Note that you don't actually have to type the whole name, but can only enter the first few characters and press the tab-key to have ReportServer autocomplete the filename for you. After you issued the pkg install command ReportServer unzips and installs the example data. This might take a minute or two, but after a while you will be presented with the *ok*-confirmation. If you reload your browser you can now use the example data.

Chapter 2

First Steps

This section will provide you with a first impression of how an administrator will work with Report-Server. On the basis of the demo data provided we will guide you step by step through the various sections of the Administration interface. We will use examples to explain the basic concepts that you will meet throughout ReportServer.

2.1 Configuration and installation

You will find a detailed description in the configuration and installation instructions in the freely available ReportServer configuration guide. As we will frequently refer to configuration files and options, we list below the most important locations where to configure ReportServer:

- **reportserver.properties** To be found in the directory WEB-INF/classes; it includes basic configuration settings which have to be rarely adapted. This applies to login control settings, or passwords to encrypt sensitive data, etc.
- **persistence.xml** To be found in the directory WEB-INF/classes/META-INF, it includes the configuration of the database connection.

All further settings will be made within ReportServer. ReportServer has an internal file system which you find in the administration module. There, in the spirit of UNIX systems, you will find the configuration files in the folder /etc. For further information on the configuration of ReportServer please refer to the configuration manual.

2.2 Login

Use your web browser and open the ReportServer home page in order to log into ReportServer. By default you will find it under the URL http://SERVERNAME:PORT/reportserver/. If you are on the server where the program is installed, the address is normally http://127.0.0.1: 8080/reportserver/. In order to log into ReportServer for the first time, it must be preconfigured so that users can log-in with their user name and password (this is the default setup). A description of the various authentication procedures is given in the ReportServer configuration guide. Now, log in by entering the user name "root" and password "root".

After login you get to the Dashboard module. The user interface is structured as follows: At the top screen margin you will find the module bar. Here you can switch between the different ReportServer modules. At the right hand side you will find the option to log off from the system as well as your user profile (click on your name) and global search. For further information on your profile and ReportServer's search functionality, please refer to the ReportServer user guide. As root you have access to the following modules:

- **Dashboard** The dashboard enables the user to get important information at a glance. For further information, please refer to the ReportServer user guide as well as to the Dashboard section.
- **TeamSpace** The TeamSpace module enables users or user groups to organize their individual work areas. The users may attend to nearly all of their settings themselves. For further information on TeamSpace, please refer to ReportServer user guide.
- **Scheduler** The Scheduler module lists all scheduled reports. For further information on the Scheduler module, please refer to section Scheduling of reports as well as to the ReportServer user guide.
- **Administration** The administration module provides access to a collection of various sub modules used by administrators. The administration module is split into the following sub modules.

User management Here users and user groups may be maintained.

- Report management Enables to manage report objects.
- **Dadget library** Dadgets (abbreviation for Dashboard Gadget) are used by the users to assemble their dashboard. In the Dadget library, administrators can provide pre-assembled dadgets which the users may simply select.
- **File system** ReportServer uses the File system in many different ways. You will find here common configuration files or ReportServer scripts. But you also can provide resources (e.g. pictures) for reports.
- **Datasources** This section enables to manage database connections which may be used as a basis for reports.
- **Global constants** Global constants may be used in reports to swap out configuration parameters so that they can be managed centrally.
- **Import** The import module enables to import ReportServer objects that had been exported before.
- **Permission management** The permission management defines the access permissions for the varying ReportServer modules/functions.
- **Scheduler** As compared to the Scheduler main module, the Administration module provides insight into any scheduler job made by the users. Now we are going to get acquainted with ReportServer in a quick resumé. We try to introduce important concepts and functions by presenting examples. If you haven't done so, you should now load the demo data. For the upcoming introduction the demo content is not necessary. See Section 1 on page 4.

2.3 Creating a data source

The basic requirement for running reports is the configuration of a data source. Data sources are organised hierarchically in a tree structure like many other objects. The ReportServer trees are organised in a similar way as you may know from files and folders of common file systems. The hierarchic structure enables you to retain the overview even when dealing with a large number of objects. In addition, the hierarchic structure enables to map even the most complicated access rules in a compact and comprehensible manner.

You will work with ReportServer trees just in the same way as you are used to from other programmes.You will create new objects by using the context menu (right-clicking an object), and with drag and drop you can move objects. Now, select the Data sources section within the Administration module. Below the Root folder, the folder "internal data sources" should already be located. Beside the existing folder, create another one: Right click on "Data source Root" -> Insert -> Folder. Now, select the newly created folder "unnamed". In the right part of the window you can edit the properties of the currently selected object. Rename the folder to "Demo data" and click on the *Apply* button.

Now, we will configure a data source which enables to access the internal demo data. Below the newly created folder, create then the data source type "relational database". Enter "Demo data" as the name for the new database connection. Select H2 as database type. User name and password are "demo". Enter the following JDBC connection URL: jdbc:h2:dbtemp/rsdemodb. The dbtemp path refers to the directory of the internal database as defined in configuration file etc/datasources/internaldb.cf in the internal file system of ReportServer. By default this is the dbtemp directory relative to the ReportServer directory. See the configuration guide on information on how to change this location.

Apply the chosen settings and run a test whether the connection can be established ("Test connection" button in the tool bar). Be aware to only test the connection after saving your changes.

2.4 Creating your First Report

In the following we will create a first basic report. To do so, switch to the section Report management in the Administration module. Here you see that reports are managed in a tree structure as well. As you did before, create a new folder with the name "demo reports". The provided demo data represents a small datamart for the fictitious model-making manufacturer "1 to 87". You will find the corresponding database scheme in the appendix. First, we want to create a customer list. For this purpose the reporting type "dynamic list" is suitable. Below the demo reports folder insert a report of type "dynamic list". Rename the report to **customer list** and select the data source (click on the magnifying glass) **demo data** that you have just created.

The configuration displayed varies with the data source selected. Relational databases require the setting of the appropriate SQL statement. We want to create a list showing any and all information about the customers of the company "1 to 87". In the demo data the T_AGG_CUSTOMER table has already been prepared for this. The following statement selects all data records from the table:

SELECT * FROM T_AGG_CUSTOMER

Apply the data and open the report by double clicking on the object in the tree. Now, the screen that opens should show the report execution area. This section is detailed in the ReportServer user manual. To get a quick overview of the "1 to 87" customers, select from *Select columns* all available columns and then on the left select *Preview* from the aspects. The first 50 data records should now display. If you wish to export the complete list, for instance, to an Excel file, select the button *Excel export* from the tool bar.

Then close the report by clicking on the button \mathbf{x} on the right hand side above the tool bar to return to the Administration section. Alternatively, select the Administration module from the module bar.

2.5 Importing a Graphical Report

In the next step we want to import a graphical report from the ReportServer sample projects. Within your ReportServer installation you should find a folder called "pkg". Within there should be a file called demobuilder-xx.zip where xx denotes your ReportServer version. Copy that file to a temporary directory and there unzip the file.You should find a tmp directory containing various files of the type "export_...zip".

Now, select the section **Import** in the Administration module and then click on the **Start import** button. Select the file export_birt_sample_reports.zip and click on **Submit**. In the left part of the screen the sections Datasources, User management and Report management should display. This indicates that objects from these sections were exported to the export file. Click on Report management and select the folder **demo reports** as the import target. At the bottom click on the **objects** tab and select the **sales invoice** report.

Now switch to Datasources and choose the **demo data** data source that you created as default data source. For the import it will be entered in the report as data source. Then at the top, right click on **Finalize import** and answer the question whether you want to reset the configuration with **Yes**. Now select **Cancel job** from the tool bar to show ReportServer that no other objects are to be imported from this file.

Switch back to Report management and open the **Demo reports** folder. You might need to reload the tree to see the freshly imported report. For this click the reload button in the tree's toolbar. The imported report **sales invoice** should be located here. Open the report to ensure that the correct data connection has been set.

If you run the report you will first be directed to the parameters page of the report. Parameters enable the user to limit the data basis in particular for graphical reports (such as JasperReports or Eclipse-BIRT reports) according to its needs. However, parameters can be applied for all reports. For instance, enter 10167 as invoice number (order number) and click on **Preview**. The report will run for exactly this invoice number. When you return to the Administration section and select the **sales invoice** report, you can view the application of this parameter. Beside the general report settings you may switch to other aspects by clicking on the tabs at the bottom margin. Select **Parameter management**. The report shows the parameter **Order Number**. If you double click on the icon of the parameter, a dialogue box opens showing the parameter settings. Close the dialogue by clicking on the **Cancel** button.

2.6 Creating users

In the following we will create the **Report Management** user group as well as the **jondoe** user. So far you have been working under the user **root**. You should use this account only in an emergency case as the user **root** operates completely isolated from the management of permissions. It is reasonable to set up user accounts in such a way that the users will always be granted only those permissions they actually need for their work. In the Administration section now switch to User management. Like reports, users will be managed in ReportServer in a hierarchic structure. This enables to easily map, for instance, company hierarchies and to create users with similar or equal permissions in a common folder (or as we call them organisational unit). Beside hierarchic structuring you can additionally organise users by means of groups. We will get back to the various structuring options in more detail in later sections. First create the organisational unit **IT** and then the user Jon Doe in a sub-folder with jondoe as user name and password. Jon Doe shall be given the permissions to create and manage reports. Here it is recommendable to create a group which will be granted the respective permissions and to add jondoe to this group. To do this, create the organisational unit roles below the root directory as well as the group **Report management**. Add the user **Jon Doe** to the group either by using the corresponding button, or directly by Drag and Drop from the user tree (drag the user to the respective member list). Don't forget to save your changes by clicking on **Apply**.

jondoe needs the following permissions to manage reports: He must be able to access ReportServer and to use the Administration module, there, however, only the Report management. Within Report management we would like to allocate all rights to him. First, switch to module Permission **management** in the Administration section. Various subsections will now display. Now, select the subsection **Administration**. In ReportServer permissions are granted based on the ACL security model (for more information on ACLs, please refer to the section on user management or to http://de.wikipedia.org/wiki/Access_Control_List). Click on Insert to create a further permission entry. The entry will be added to the list. To edit the entry, double click on it. From the mask in column **Folk** (the receiver of rights) select the group **Report management** and allocate reading rights (r). To grant jondoe ReportServer access, go to the subsection **ReportServer Access** and create an ACE (an access control entry; a permission entry) that grants jondoe (or the group **Report management**) the execute (x) right. Now, when you log off and log on as jondoe, you will see that he will be allowed to see the Administration section, however, he will have no further access rights. Therefore, log in again with **root** and return to Permission management. Select subsection Report management and here as well add an access right by granting the group Report **management** reading access. Now Jon Doe is granted the rights to use the Administration module and there the Report management sub-module. However, he is lacking any rights with regard to single report objects. In ReportServer access rights can be granted fine-tuned down to the level of single objects. As jondoe is to account for the report management, we will assign him the rights so that he may access all report objects. To do this, go to Report management and select the root node. Switch to the permissions view of the root object by clicking on the tab below the expanded view. Add a further permission entry and double click on the newly created entry. The dialogue is identical to the already known permission dialogue box except for the option inherit. The option **inherit** controls the use of permission entries in trees. We wish to assign to jondoe any and all rights for all objects. To do so, under Folk we select again the group **Report management**. Keep the default settings for access type and inherit. In addition, we either set a tick at all rights individually, or we use the Quick assign and select Full access. If you now sign in as jondoe

you should be granted access to the complete Report section. When you are logged in with the user name **root** and you wish to quickly change the user you may also use the SU (switch user) command. You can do this by using the keyboard shortcut CTRL+L. In the dialogue box popping up select the user jondoe and click on **Submit**. The users who may execute the SU command may also be controlled in the Permission management.

2.7 Terminal and FileServer

Now log off and log in again as the user **root**. In the last part of the introduction we will get acquainted with two important Administration tools enabling to perform complex actions to ReportServer: the Terminal and ReportServer Scripts.

The Terminal follows the familiar Unix terminal and uses various commands to enable the set-up of a database connection, to move objects or retrieve system information. Open the Terminal by entering the keyboard shortcut CTRL-ALT-T. A window opens showing a prompt. Enter the command ls and confirm with ENTER. The ls command shows the objects in the current folder. All ReportServer trees (e.g. Report management or User management) are accessible from the Terminal, and are integrated as virtual file systems.

In the following we describe how to create a simple ReportServer script which goes into the internal **File System** (a sub-module of the administration module). The File System can be accessed via the Administration module or via the Terminal. Now enter the command cd fileserver to change the working directory to the root folder of the file system. The terminal supports the auto-completion function to enable quicker navigation within the Terminal. You can activate it by pressing the tab key. If you want to make sure which folder you are currently in, you can use the pwd command.

To verify whether the bin directory exists in FileServer, use the ls command. If it fails to exist create it by entering the mkdir bin command. With cd bin switch to the newly created directory. Now, we will create our first ReportServer script. In ReportServer, scripts have manifold tasks and are written in the programming language Groovy (http://groovy.codehaus.org/). For working with ReportServer, scripts are not necessarily required, however, you can use them to accomplish many special tasks (such as integrating complex data sources or importing of reports from third-party systems) and small enhancements. In this manual we will time and again meet with scripts. You will find a detailed introduction in the ReportServer scripting guide. Now create the script hello.rs by entering the command createTextFile hello.rs. A text editor opens. Enter the following line:

"Hello ReportServer Admin"

Be sure to enter the quotation marks. Click on Submit to save the script. To edit the file use the command editTextFile hello.rs. You can run the script with the command exec hello.rs. You should see the line

```
Hello ReportServer Admin reportserver$
```

on the terminal.

With this we would like to finish our quick ReportServer resumé. Up to this point we have only en-

countered a fractional part of the ReportServer options. In the sections to come we will specifically outline individual ReportServer sections and discuss them in detail.