Elixir Administration Tools

Release 3.0.0

Elixir Ambience

Elixir Technology Pte Ltd

Elixir Administration Tools: Release 3.0.0

Elixir Technology Pte Ltd

Published 2014

Table of Contents

List of Figures

List of Tables

Chapter 1
Elixir Administration Tools

Overview

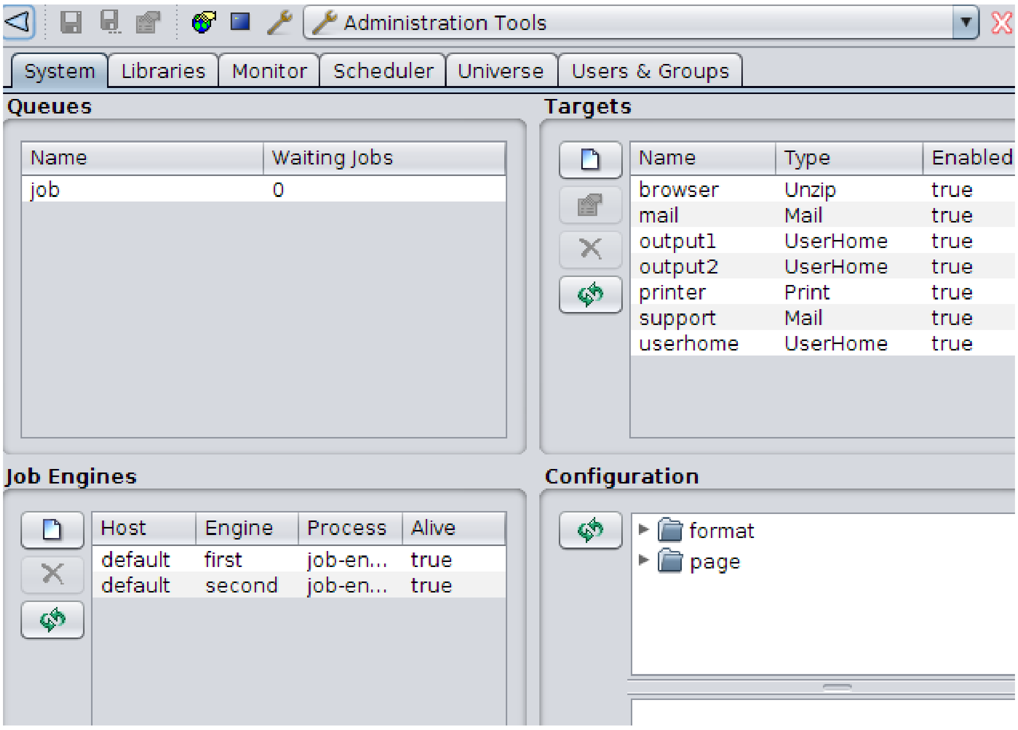
On clicking the Admin Tools... button on the Action Bar, you will see the Administration Tools panel. These tools include Anonymous, System, Libraries, Monitor, Scheduler and Users & Groups.

System

The System page is used to create job engines, monitor job queues, set targets, and configure date and time formats.

Jobs are run by many Job Engines according to their sequence in the queue. Each Job Engine takes a task from the queue, sends the result and writes the log into the Repository. Each Job Engine only processes one task at a time. Unwanted jobs can be deleted.

Figure 1.1. System



Name	Waiting Jobs
job	0

Name	Type	Enabled
browser	Unzip	true
mail	Mail	true
output1	UserHome	true
output2	UserHome	true
printer	Print	true
support	Mail	true
userhome	UserHome	true

Host	Engine	Process	Alive
default	first	job-en...	true
default	second	job-en...	true

Queues

The Queues page shows the history of tasks run by Job Engines.

Job Engines

The Job Engines page is designed for controlling the job engines on the pre-defined set of hosts. You have the option to add and remove job engines, as well as set the host, engine name and process. If you have multiple hosts configured, you can add and remove the engines across all the hosts.

Configuration

This allows you to perform common configurations that will apply across all Ambience modules. For example, you can set the uniform formats for date, datepicker, time, timepicker and timestamp. When a date is displayed by Java, it will use the /Configuration/format/date/ settings. When JavaScript is used, it will use the /Configuration/format/datepicker settings by default, unless there is a local override. Web-based modules will reflect the changes quickly, while other modules may show the changes after they are restarted.

Right-click **Configuration > format > date**, **datepicker**, **time**, **timepicker** or **timestamp**, the following menu option will display:

- **Edit contents...:** This enables you to edit the contents of the current node. You can also validate the XML.

Targets

After a job finishes, a target may pick up the result. Different targets may be chained for further processing. Target processing is also a task run by the Job Engine.

Target Constants

If you find yourself typing a string repetitively when configuring targets, you can define that string as a constant. Then you can refer to that constant in target property values like ${constant-name}. Target constants can be enabled/disabled. Only enabled constants can be used in target configurations. You can define multiple constants with the same name, but only one of them can be enabled at any time.

Target List

- Target Creation/Update/Deletion
  All manipulations can be done in the Target Wizard. For most of the targets, configuration is simple. You need to provide a name, enable the target and provide values for required target properties.

- Status
  Targets can also be enabled/disabled. Only enabled targets can be used in RenderReport task. You can define multiple targets with the same name, but only one of them can be enabled. If you make one target enabled, the rest targets with the same name will be disabled automatically.

  Targets can be be available/unavailable to anonymous users. When the **Available to anonymous users** check box is selected, the target is accessible in anonymous mode. By default, only the "browser" target is available.

- Target Properties and Parameters
  A target requires certain properties such as file name, folder name, port number and so on.

  When editing a target, the Target Wizard lists all required properties for the target. An administrator has several options when configuring those property values:

  - Provision of an exact string value for some properties.

  - Reference to target constants.
  - Definition of parameters in some properties if they should be provided by end users. For example, the parameter in property "filename" can be defined as "${file#report}_${date}". End users should provide values for those parameters when invoking targets.

- Output Types for Report Rendering
  A target provides various output types such as CSV, HTML, PDF, Microsoft formats and so on.

  After the properties are set, the Target Wizard enables you to choose among different output types for report rendering purpose. You can choose multiple types at a time for convenience. You can also print the report directly.

Table 1.1. Output Types for Report Rendering

CSV	Glint
HTML	Line Printer
Logical RML	Logical RML Tree
Microsoft DOCX	Microsoft Excel
Microsoft PPTX	ODF Spreadsheet
PCL	PDF
Postscript	Print
Rich Text Format	Simple HTML
XML	Zipped Bitmap
Zipped Jpeg	Zipped Png
Zipped Svg	Zipped Tiff
Zipped Wbmp	

Create New Target

When you create a new target using the Target Wizard, you should fill in values, for example parameters, into the property fields. Nested parameters are not supported. For example, you cannot specify a parameter like ${file##${reportname}}.

JDBC Target

A JDBC Target allows reports to be written directly into a database. This is useful if you have some subsequent program to pick them up or otherwise act on them - for example a document management system. Each report is written as a record into a specific table in the database. The report data itself is stored as a BLOB. Before you can use the JDBC target, you need to set up a database with a table that has the correct schema to accept a report file. An example as shown:

```
CREATE TABLE JOBOUTPUT (
  id INTEGER NOT NULL GENERATED ALWAYS AS
   IDENTITY (START WITH 1, INCREMENT BY 1),
  name VARCHAR(256) NOT NULL,
  lastModified BIGINT NOT NULL,
  content BLOB NOT NULL,
  CONSTRAINT JOBOUTPUT_PK PRIMARY KEY(id) )
```

Once the database is setup, configuration will write into a table called JobOutput in the Derby database that is built into the Elixir Repertoire Server:

Table 1.2. JDBC Target Configurations

Name	Value
Driver	org.apache.derby.jdbc.EmbeddedDriver
URL	jdbc:derby:/home/.../jdbctarget
Table	${update table##JobOutput}
User	${userid##Enter your userid}
Password	${password##Enter your password}
Filename	${filename##Job_Output_Report}
Overwrite	true

JMS Target

A JMS Target can be used for asynchronous messaging. JMS applications can use job messages as a form of managed request/response processing, to give remote feedback to the users on the outcome of their send operations and the fate of their messages. Examples of job messages are Exception, Expiration, Confirm on arrival (COA), Confirm on delivery (COD), etc.

Table 1.3. JMS Target Configurations

Name	Value
Destination	RQueue
User	${userid##Enter your userid}
Password	${password##Enter your password}
Filename	${filename##Pet_Store_User_Accounts_Report}
Reply required	true
Timeout	${reply timeout in secs##30}
Reply success keyword	${reply success keyword##OK}
Reply success pattern	${reply success pattern##^.*OK.*$}

Mail Target

A Mail Target allows the output to be sent by email. There are a number of parameters to specify, but remember that you can use substitutions to avoid hard-coding those that you decide need to be flexible. The report will be sent as an attachment by email, so you can choose the render format you prefer.

Table 1.4. Mail Target Configurations

Name	Value
To	${to##sam@elixirtech.com}
SMTP host	elixir.aspirin
From	${from##susan@elixirtech.com}
Cc	${cc##bob@elixirtech.com}
Subject	${subject##Pet Store User Accounts Report From Elixir Server}
Message	${message##Your report is attached.}
Filename	${filename##Pet_Store_User_Accounts_Report}

PDF Signer Target

PDF Signer Target is used when a PDF output needs to be signed digitally. It prints a "signature" in a PDF file when the file is rendered.

Table 1.5. PDF Signer Target Configuration

Name	Value
Keystore	${key.url##C:/Java/jre7/bin}
Keystore type	${keystore-type}
Store encrypted password	false
Store password	${key-password##keyStorePassword}
Key alias	${key-alias}
Signer appearance	${signer-appearance##self-signed}
Info reason	${info-reason##Reason for signature}
Info location	${info-location##bottom-left}
Sign width	${sign-width##100}
Sign height	${sign-height##60}
Sign page	sign-last-page
Sign image	${signature-image-url##D:/image/logo.png}
Certification level	${certificate-level##certificate-no-changes}
Filename	Specify file name here

PDF Signer Target Properties

Signer appearance

There are two types of signing mode. They are *self-sign* and *wincer-sign*.

self-sign signer can be generated using Java key generator (keytool.exe). An example of a signature command is as follows :

```
keytool -genkey -keyalg RSA -alias QAkey -keypass mypassword
  -keystore keystore.ks -dname "cn=My Key Name, c=SG"
```

wincer-sign is recommended if higher security is required. Certificate obtained is installed to the web browser and needs to be exported to be used in PDF signing.

Keystore type

For *self-sign*, no keystore type is required. For *wincer-sign*, the keystore type will depend on the respective vendor. For example, a VeriSign keystore type will be `pks12`.

Keystore

The directory of the keystore is entered here. *repository* is prohibited for this parameter.

Key alias

This parameter is optional. It is to define the alias used by the keystore file.

Store password

key-password is the password entered when creating the keystore file. The password can either be in encrypted or unencrypted form.

Store encrypted password

This parameter holds a boolean value, `true` if the key password is encrypted and `false` if otherwise.

Sign page

This parameter is to specify the page that the signature is placed. *sign-no-page* will mean that the signature will not be visible. *sign-first-page* and *sign-last-page* implies that the signature will be placed on the first and last page of the PDF file respectively. If you wish to place the signature on a specific page, enter the page number as the value for the parameter. For example, *${sign-page##5}*, for placing the signature on the 5th page of the PDF document.

Sign width and Sign height

These two parameters are for user to specify the height and width of the signature rectangle size.

Sign image

The directory of the image that is to be used with the signature. When no value is entered for this parameter, there will still be a PDF Signature (self-signed) automatically generated.

Info reason and Info location

The reason for placing a signature in the PDF document and the position of the reason.

Certification level

certificate-not
: has no visible signature in the PDF document. The document is digitally counter signed.

certificate-no-changes
: displays the signature in the PDF document and no changes can be made to the document.

certificate-form-filling
: also displays the signature in the PDF document, but only the filling in of forms, signing and page adding is allowed for the document.

certificate-form-filling-annotation
: displays the signature in the PDF document, but only commenting, form fill-in, signing and page adding is allowed.

Print Target

A Print Target allows you to send a report to a named printer. The only option is the name of the printer. If you have multiple alternate printers, you could use a separate target for each printer so that you could control access by different groups. In most cases, the printer names will be fixed and not include substitutions. You can also leave the printer name blank as it will route automatically to the default printer defined on the server.

Table 1.6. Print Target Configurations

Name	Value
Printer name	Canon iR C3220 PCL5c

Repository Target

A Repository Target writes the report to the filesystems. You can identify a target folder in the repository and provided it is writable, files will be written there. This works regardless of whether the target filesystem is of type local, secure or dbfs. You should use Repository Targets when you want to allow users to view the reports through their browser as the repository will automatically update to show the latest files.

During the configuration, the "folder" property must refer the "dir##" parameter to an existing target filesystem in the repository. The "folder##" parameter need not as it will create a new folder after its name if it is not found in the repository.

Table 1.7. Repository Target Configurations

Name	Value
Folder	${dir##ElixirSamples}/${folder##repository}
Filename	${filename##Pet_Store_User_Accounts_Report}
Overwrite	true

SFTP Target

A SFTP Target allows the report to be transferred to a user's secured FTP Server. The available parameters are user, password, host, port, dir and filename. The port is optional and will default to 22, which is the default SFTP port, if not specified.

The parameter in "dir" property must be an existing directory found in the target ftp server.

Table 1.8. SFTP Target Configurations

Name	Value
Host	${sftp host##domain_name.com}
Port	${port##22}
Directory	${dir##dept1}/${folder##Pet_Store}
Filename	${filename##Pet_Store_User_Accounts_Report}
User	${userid##Enter userid to access sftp client}
Password	${password##Enter password to access sftp client}

Socket Target

A Socket Target sends the report to a program which is listening, typically on another machine. For example, a program can be written that listens on a company.com port 6000 and writes any data it receives to a database, or to a fax etc. It is up to the receiving program what it does with the data. The server opens a connection to the listening program, using the host and port information required by the socket target and streams the data across the network to the listening socket.

Table 1.9. Socket Target Configurations

Name	Value
Host	company.com
Port	6000

Unzip Target

This target enables you to specify a file name for unzipping purpose. You can enable/disable this target, and set this target available/unavailable to anonymous users.

User Home Target

A Repository User Home Target writes the report to the invoking user's folder in the repository. Users can share jobs and they can keep separate output without overwriting one another. In the example, once the user "Jon" signs in to the repository and runs the job, the report will be written to /User/Jon/Pet_Store.

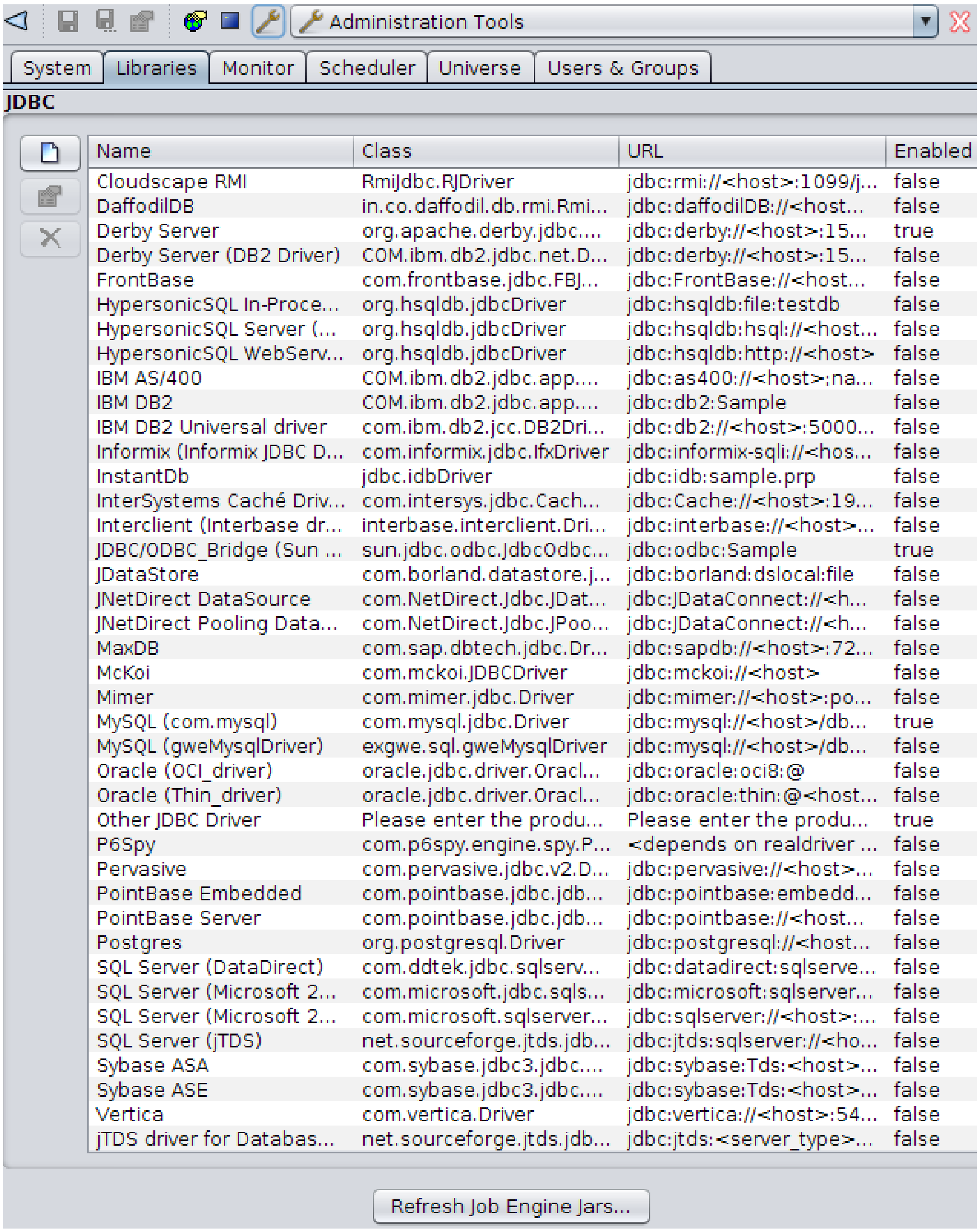
Table 1.10. Repository User Home Target Configurations

Name	Value
Folder	${folder##Pet_Store}
Filename	${filename##Pet_Store_User_Accounts_Report}
Overwrite	true

Libraries

Libraries include configuration information for a full range of JDBC drivers. Repertoire 9 enables you to configure JDBC drivers more easily. Any JDBC drivers that require native components will usually require the Ambience system to be stopped and restarted. Currently, most drivers are JDBC-Net pure Java drivers or native protocol pure Java drivers. If there is no native code, then you can transfer the JAR files to the job engine machines by putting them in /Public/lib in the Repository Tree and clicking the Refresh Job Engine Jars... button. This will cause the job engines to stop at the end of their current task. The Process Managers will then start them up again, with the updated classpath (including any new JDBC drivers).

Figure 1.2. Libraries



Name	Class	URL	Enabled
Cloudscape RMI	RmiJdbc.RJDriver	jdbc:rmi://<host>:1099/j...	false
DaffodilDB	in.co.daffodil.db.rmi.Rmi...	jdbc:daffodilDB://<host...	false
Derby Server	org.apache.derby.jdbc....	jdbc:derby://<host>:15...	true
Derby Server (DB2 Driver)	COM.ibm.db2.jdbc.net.D...	jdbc:derby://<host>:15...	false
FrontBase	com.frontbase.jdbc.FBJ...	jdbc:FrontBase://<host...	false
HypersonicSQL In-Proce...	org.hsqldb.jdbcDriver	jdbc:hsqldb:file:testdb	false
HypersonicSQL Server (...	org.hsqldb.jdbcDriver	jdbc:hsqldb:hsql://<host...	false
HypersonicSQL WebServ...	org.hsqldb.jdbcDriver	jdbc:hsqldb:http://<host>	false
IBM AS/400	COM.ibm.db2.jdbc.app....	jdbc:as400://<host>;na...	false
IBM DB2	COM.ibm.db2.jdbc.app....	jdbc:db2:Sample	false
IBM DB2 Universal driver	com.ibm.db2.jcc.DB2Dri...	jdbc:db2://<host>:5000...	false
Informix (Informix JDBC D...	com.informix.jdbc.IfxDriver	jdbc:informix-sqli://<hos...	false
InstantDb	jdbc.idbDriver	jdbc:idb:sample.prp	false
InterSystems Caché Driv...	com.intersys.jdbc.Cach...	jdbc:Cache://<host>:19...	false
Interclient (Interbase dr...	interbase.interclient.Dri...	jdbc:interbase://<host>...	false
JDBC/ODBC_Bridge (Sun ...	sun.jdbc.odbc.JdbcOdbc...	jdbc:odbc:Sample	true
JDataStore	com.borland.datastore.j...	jdbc:borland:dslocal:file	false
JNetDirect DataSource	com.NetDirect.Jdbc.JDat...	jdbc:JDataConnect://<h...	false
JNetDirect Pooling Data...	com.NetDirect.Jdbc.JPoo...	jdbc:JDataConnect://<h...	false
MaxDB	com.sap.dbtech.jdbc.Dr...	jdbc:sapdb://<host>:72...	false
McKoi	com.mckoi.JDBCDriver	jdbc:mckoi://<host>	false
Mimer	com.mimer.jdbc.Driver	jdbc:mimer://<host>:po...	false
MySQL (com.mysql)	com.mysql.jdbc.Driver	jdbc:mysql://<host>/db...	true
MySQL (gweMysqlDriver)	exgwe.sql.gweMysqlDriver	jdbc:mysql://<host>/db...	false
Oracle (OCI_driver)	oracle.jdbc.driver.Oracl...	jdbc:oracle:oci8:@	false
Oracle (Thin_driver)	oracle.jdbc.driver.Oracl...	jdbc:oracle:thin:@<host...	false
Other JDBC Driver	Please enter the produ...	Please enter the produ...	true
P6Spy	com.p6spy.engine.spy.P...	<depends on realdriver ...	false
Pervasive	com.pervasive.jdbc.v2.D...	jdbc:pervasive://<host>...	false
PointBase Embedded	com.pointbase.jdbc.jdb...	jdbc:pointbase:embedd...	false
PointBase Server	com.pointbase.jdbc.jdb...	jdbc:pointbase://<host...	false
Postgres	org.postgresql.Driver	jdbc:postgresql://<host...	false
SQL Server (DataDirect)	com.ddtek.jdbc.sqlserv...	jdbc:datadirect:sqlserve...	false
SQL Server (Microsoft 2...	com.microsoft.jdbc.sqls...	jdbc:microsoft:sqlserver...	false
SQL Server (Microsoft 2...	com.microsoft.sqlserver...	jdbc:sqlserver://<host>:...	false
SQL Server (jTDS)	net.sourceforge.jtds.jdb...	jdbc:jtds:sqlserver://<ho...	false
Sybase ASA	com.sybase.jdbc3.jdbc....	jdbc:sybase:Tds:<host>...	false
Sybase ASE	com.sybase.jdbc3.jdbc....	jdbc:sybase:Tds:<host>...	false
Vertica	com.vertica.Driver	jdbc:vertica://<host>:54...	false
jTDS driver for Databas...	net.sourceforge.jtds.jdb...	jdbc:jtds:<server_type>...	false

Configuring JDBC Drivers

To illustrate the JDBC driver configuration procedure, here are the steps:

1. Click the Administration Tools... button on the workspace toolbar. Click the Libraries tab. The JDBC drivers display.

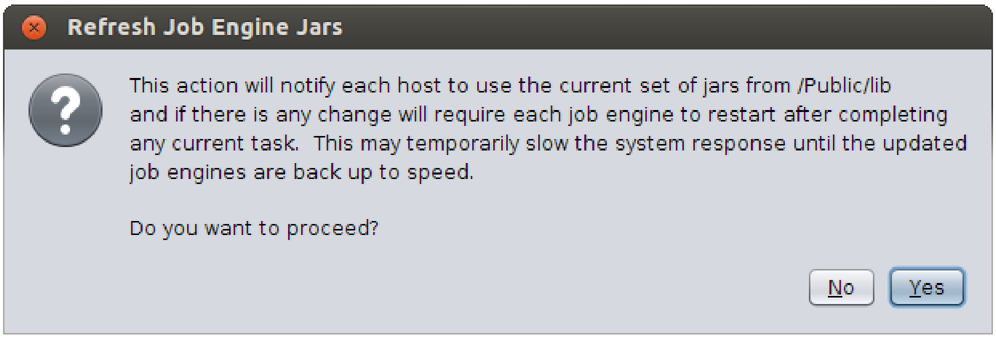
2. Find and select the desired driver from the list. Click the Edit button. The JDBC Driver dialog displays. Fill in the host machine name to replace <host> in the URL field. Leave the class information as it is.

3. If the desired driver cannot be found in the list, click the Add button in the Libraries tab page. Fill in JDBC driver name, class and URL information.

4. Select the Enabled checkbox and click OK.

5. If there is no native code for the driver, download the driver’s JAR file from the vendor's Web site. Copy the driver's JAR file into the /Public/lib folder in the Repository Tree. Click the Refresh Job Engine Jars... button. A message may pop up, as shown in Figure 1.3, “Refresh Job Engine Jars”:

Figure 1.3. Refresh Job Engine Jars



This action will notify each host to use the current set of jars from /Public/lib and if there is any change will require each job engine to restart after completing any current task. This may temporarily slow the system response until the updated job engines are back up to speed.

Do you want to proceed?

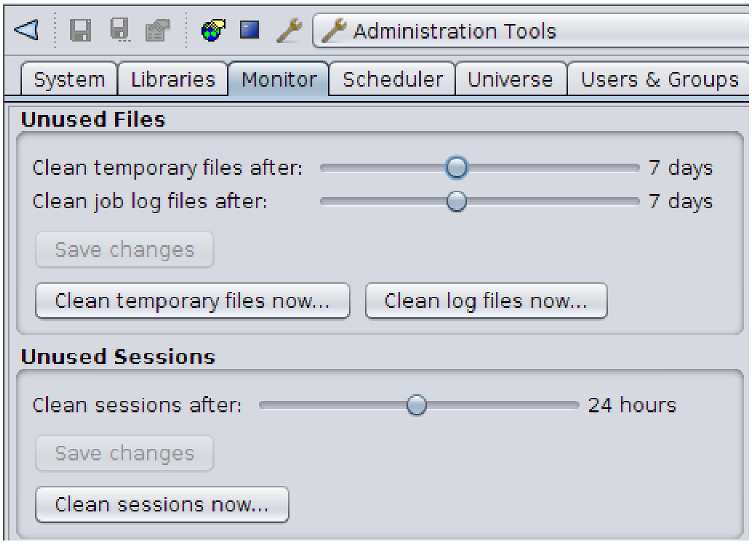
Monitor

The Monitor interface is used to control the regularly scheduled cleanup and other monitoring tasks. Set the slider to the appropriate position to specify the time frame. These time frame values will control which files should be removed in each task.

There are three options available:

- **Clean temporary files now...:** After specifying the time period (between 2 and 14 days) that the temporary files have not been accessed, you can manually clean those unused temporary files by clicking this button.
- **Clean log files now...:** After specifying the time period (between 2 and 14 days) that the log files have not been accessed, you can manually clean those unused log files by clicking this button.
- **Clean sessions now...:** After specifying the time period (between 1 and 48 hours) that the sessions have been inactive, you can manually clean those sessions by clicking this button.

Figure 1.4. Monitor



Clean temporary files after: 7 days

Clean job log files after: 7 days

Clean sessions after: 24 hours

Scheduler

Through the Scheduler interface, Calendars and Triggers can be created and modified.

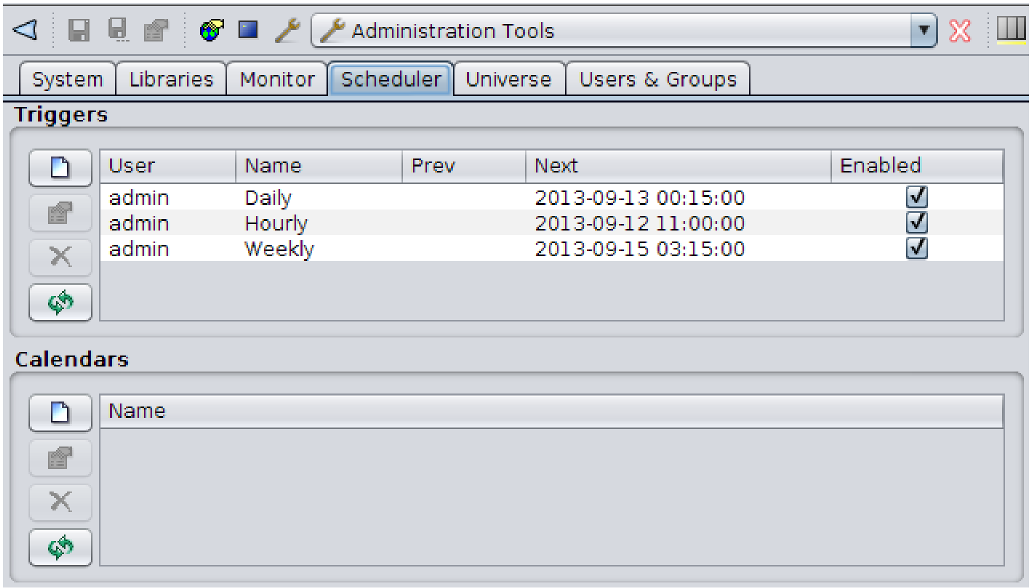
Calendars

Calendar is used to mark daily, weekly, monthly events. Specific dates and CRON expressions can also be used. When the Calendar file is added to a Trigger, the Trigger will not run on those marked dates. For more information, refer to Elixir Scheduler User Manual.

Triggers

Trigger is used to set the configurations to decide when a job is invoked. For more information, refer to Elixir Scheduler User Manual.

Figure 1.5. Scheduler



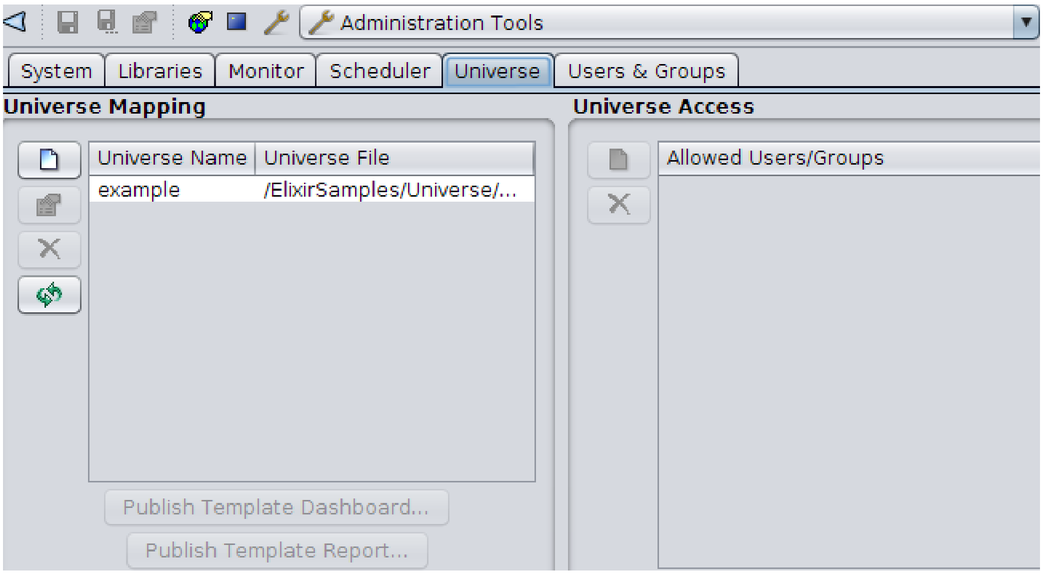
User	Name	Prev	Next	Enabled
admin	Daily		2013-09-13 00:15:00	☑
admin	Hourly		2013-09-12 11:00:00	☑
admin	Weekly		2013-09-15 03:15:00	☑

Name

Universe

Through the Universe interface, you can map a Universe name with its path, publish template dashboard or report, and define which users and groups will be able to access the Universe.

Figure 1.6. Universe



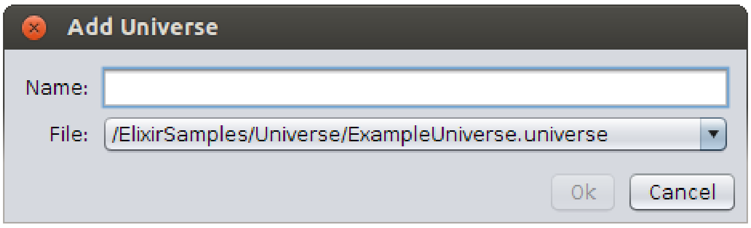
Universe Name	Universe File
example	/ElixirSamples/Universe/...

Universe Mapping

A Universe can exist anywhere in the Repository, but are only accessible if its path is mapped to a Universe name. Only administrators can set or alter the Universe mapping.

Complete the following steps to map a Universe:

Figure 1.7. Add Universe



Name:

File: /ElixirSamples/Universe/ExampleUniverse.universe

1. In the Administration Tools, click the **Universe** tab.

2. In the Universe Mapping pane, click the **Add Universe** button.

3. In the Add Universe window, enter a name and select the desired universe from the File list.

4. Click OK. The Universe name and path are successfully mapped.

Note

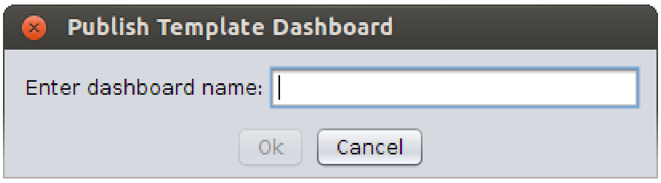
Unwanted Universe(s) can be easily removed from the Universe Mapping pane by a mouse click.

For a Repository Universe, once its name and path have been mapped, the name will display as an option within Ad-hoc Dashboard. After setting the Primary Key, you can view the Ad-hoc Dashboard via the following URL. Fill in your domain, host machine and dashboard names to proceed:

```
http://<host>:8080/elx/do/<domain>/ei/accolades/dashboard/Sample/
Demo/Ad%20Hoc/Dashboard/<dashboard>.dashboard?mode=file
```

You also have the option to publish template dashboard or report, as shown in the figures below:

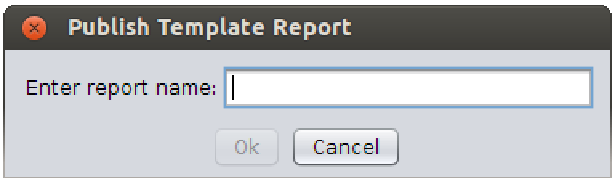
Figure 1.8. Publish Template Dashboard



Publish Template Dashboard

Enter dashboard name:

Figure 1.9. Publish Template Report



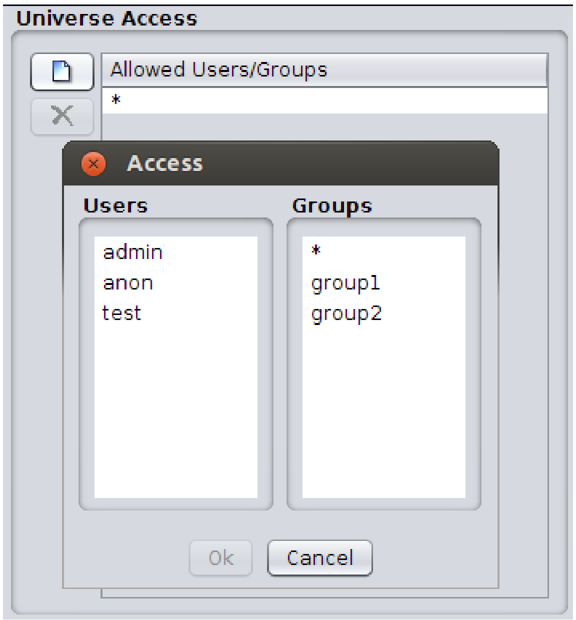
Publish Template Report

Enter report name:

Universe Access

Complete the following steps to control the access to the Universe:

Figure 1.10. Universe Access



Universe Access

Allowed Users/Groups

*

Access

Users	Groups
admin	*
anon	group1
test	group2

1. In the Administration Tools, click the **Universe** tab.

2. In the Universe Access pane, click the **Add Access** button.

3. In the Access window, select the desired users/groups from the list.

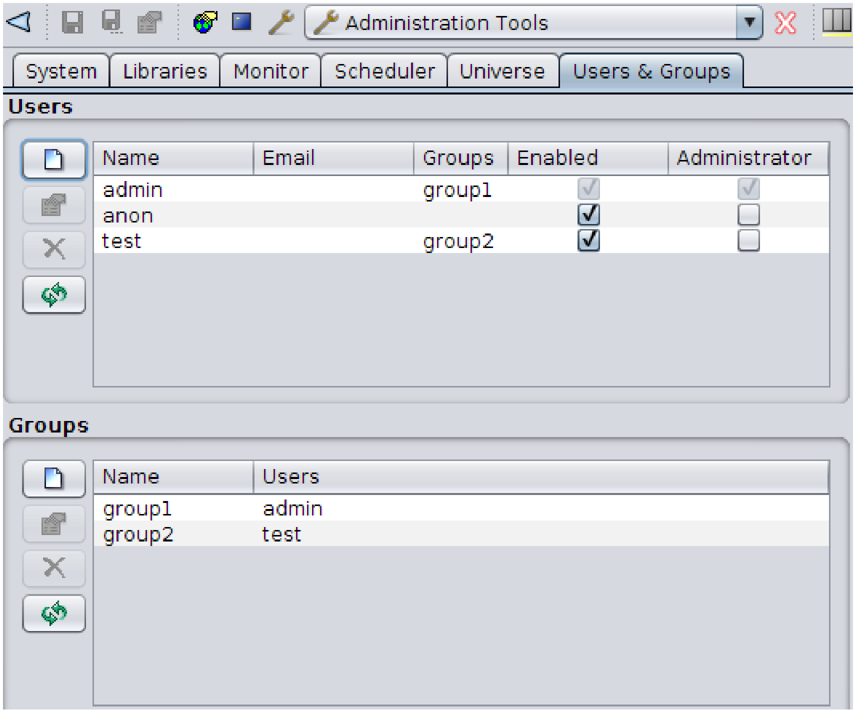
4. Click OK. The access rights are successfully assigned.

Only users who can read or write the Universe file can open it to view and edit the contents. However, those users in the Access list can only use the Universe, without being able to open or edit the Universe file. The Universe service will access the file for those users.

Users & Groups

The Users & Groups interface is designed for administrators to create and modify users and groups.

Figure 1.11. Users & Groups



Users

Name	Email	Groups	Enabled	Administrator
admin		group1	☑	☑
anon			☑	□
test		group2	☑	□

Groups

Name	Users
group1	admin
group2	test

Users

Administrators can create, enable and disable user accounts, assign administration privileges, specify user groups, as well as set anyone's password and email address. The users without administration privileges can change only their own password and email address.

Groups

Administrators can create new groups and assign group members.