

# Migration from 6.x.x to 7.x.x

## Introduction

Repertoire Server 7 has greatly improved the configurability of the system environment such as repository filesystem and user access security. A web interface is provided for administration, repository browsing, triggering of reports, and accessing to reports, data sources and dashboards.

The JMX HTTP console is no longer required as system monitoring is supported via standard Java JMX console, available as part of Java Development Kit.

Repertoire Server 7 provides new REST APIs to access the entire range of server services. The **Java ERSClient API** for rendering reports and data sources is retained for backward compatibility.

Perspective is now part of Repertoire 7 Server component. The Perspective Servlet web archive file or a separate J2EE server is no longer required. You are advised to retest all dashboards designed in Repertoire 6 with Repertoire 7 Designer and Server. Minor modifications may be needed to support new features implemented such as a fixed-sized card dimension and the use of borders with margin and padding for aesthetic and precision layout control.

## Installation

Do NOT attempt to install version 7.x over on the same location as an existing version 6.x installation as this prevents ERS 7.x from starting up correctly.

The minimum Java Run Time requirement for Repertoire Server version 7.x is Java version 5\_08 and above.

The minimum Java Run Time requirement for Repertoire Server version 6.x is Java version 1.4.

## Configuration

Do NOT copy all configuration files in `<ERS6_installation>/config` over to `<ERS7_installation>/config` as this prevents ERS 7.x from starting up correctly.

## Core Engine

The default port used for ERS 7.x is 8080. Port 8089 used in version 6.x is no longer required.

## External Jars

Copy all existing jars in *ERS6.x \_ext\_ folder* to the new *ERS7.x \_ext\_ folder*.

## Repository

`<server_installation>/config/repository.xml` ( in version 6.x ) is no longer applicable in 7.x server. Hence, you need to add all the filesystems used in version 6.x to version 7.x through the Administration web interface. All the repository settings are stored in the database in version 7.x.

## Users/Roles

`<server_installation>/config/user.properties` and `<server_installation>/config/role.properties` (in version 6.x) is no longer applicable in 7.x server. Hence, add all the users/roles configuration used in version 6.x to version 7.x through the Administration web interface. All the repository settings are stored in the database in version 7.x.

User "Anonymous" (in version 6.x) needs to be explicitly created in version 7.x, if required.

## API

In version 6.x, the following jar files are provided:

- Glint.jar
- jdom.jar
- log4j.jar
- mx4j-rimpl.jar
- mx4j-rjmx.jar
- RepertoireClient.jar
- servlet-api.jar

In version 7.x, the following jar files are provided:

- Glint.jar
- log4j.jar
- RepertoireClient.jar

For an existing Java application, using the ERS Client to access server-side resources, remove all 6.x jar files as mentioned above and make available the new 7.x jars in your Java Environment.

## Known Differences between version 6.x and version 7.x

1) In version 7.x,

- `com.elixirtech.ers2.client.ERSClient.getFile()`
- `com.elixirtech.ers2.client.ERSClient.getReport()`

does not include the filesystem name when the value is returned.

e.g

- In version 6.x the `getFile()` or `getReport()` method will return `"/ElixirSamples/Report/Charting/3D Visualization.rml"`
- In version 7.x the `getFile()` or `getReport()` method will return `"/Report/Charting/3D Visualization.rml"`

As a workaround, the filesystem name needs to be appended programmatically.

e.g.

```
_String test = "/" + "MyFileSystem" _  
  
_if (!test.startsWith(reportpath)) // Test if the reportpath return contain the filesystem name _  
  
_ { _  
  
_reportpath = test + reportpath // If not, append the filesystem name to the report path _  
  
_ } _
```

2) In version 7.x, **ReplyTarget** is implicit. Therefore, the keyword **me** is not needed to set the target.

e.g.

- In version 6.x - `client.renderReport(rptName, "application/pdf", fos, props, "me");`
- In version 7.x - `client.renderReport(rptName, "application/pdf", fos, prop);`

## Deprecated Methods

- `com.elixirtech.ers2.client.ERSClient.renderReport(java.lang.String report, java.lang.String mimeType, java.io.OutputStream os, java.util.Properties properties, java.lang.String target)`
- `com.elixirtech.ers2.client.ERSClient.setConnectionMode(short mode)`
- `com.elixirtech.ers2.client.ERSClient.setAutoEscape(boolean b)`

- `com.elixirtech.ers2.client.ERSClient.isAutoEscape()`
- `com.elixirtech.ers2.client.ERSClient.isConnected()`

Download:

- [Migration Guide](#)
- [Migration Checklist](#)