**FUND FINDER USER**

**INSTALLATION MANUAL**

**Table of contents**

Introduction 3

Delivery package 3

Installation 3

Database 3

Web Sphere 3

Environment variables 3

JDBC providers 4

JAAS - J2C authentication data 5

Data source 6

Shared libraries 9

Deploying application 10

Starting application 13

# Introduction

This document describes how to install **Fund Finder User** application on WebSphere AS.

Important note: Fund Finder Admin application must be **installed and started** prior to installation of Fund Finder User application.

# Delivery package

Fund Finder delivery package includes the following:

|  |  |
| --- | --- |
| **[docs]**  **installation-manual.docx**  **functional-specification** | Folder that contains Fund Finder related documents. |
| **[libs]**  **javassist-3.20.0-GA.jar**  **sqljdbc4.jar** | Folder that contains JAR files required during the installation.  - Java Programming Assistant  - Microsoft JDBC drivers for SQL Server |
| **[shared]**  **application.properties**  **application-prod-mssql.properties**  **logback.xml** | Folder that should be configured in WebSphere as shared library for Fund Finder User application.  It contains externalized application properties and logback configuration.  Note: All customer specific properties should be defined in application-prod-mssql.properties. |
| **fundfinder.war** | Deployable unit in form of web archive. |

# Installation

## Database

Database is created by Fund Finder Admin application during the initial startup by *FlyWay* mechanism from provided SQL scripts.

## Web Sphere

In the following subchapters we give an overview of all the steps required to successfully deploy Fund Finder User application to WebSphere AS.

Please note that described installation steps were performed on **WebSphere 8.5.5.10**.

Also, some steps might not be necessary as already done before (e.g. during the installation of Fund Finder Admin application).

### Environment variables

Note: This step is probably already done during the installation of Fund Finder Admin application, and therefore **can be safely skipped**.

In **Administrative console** go to **Environment / Web Sphere variables** and set the following variables:

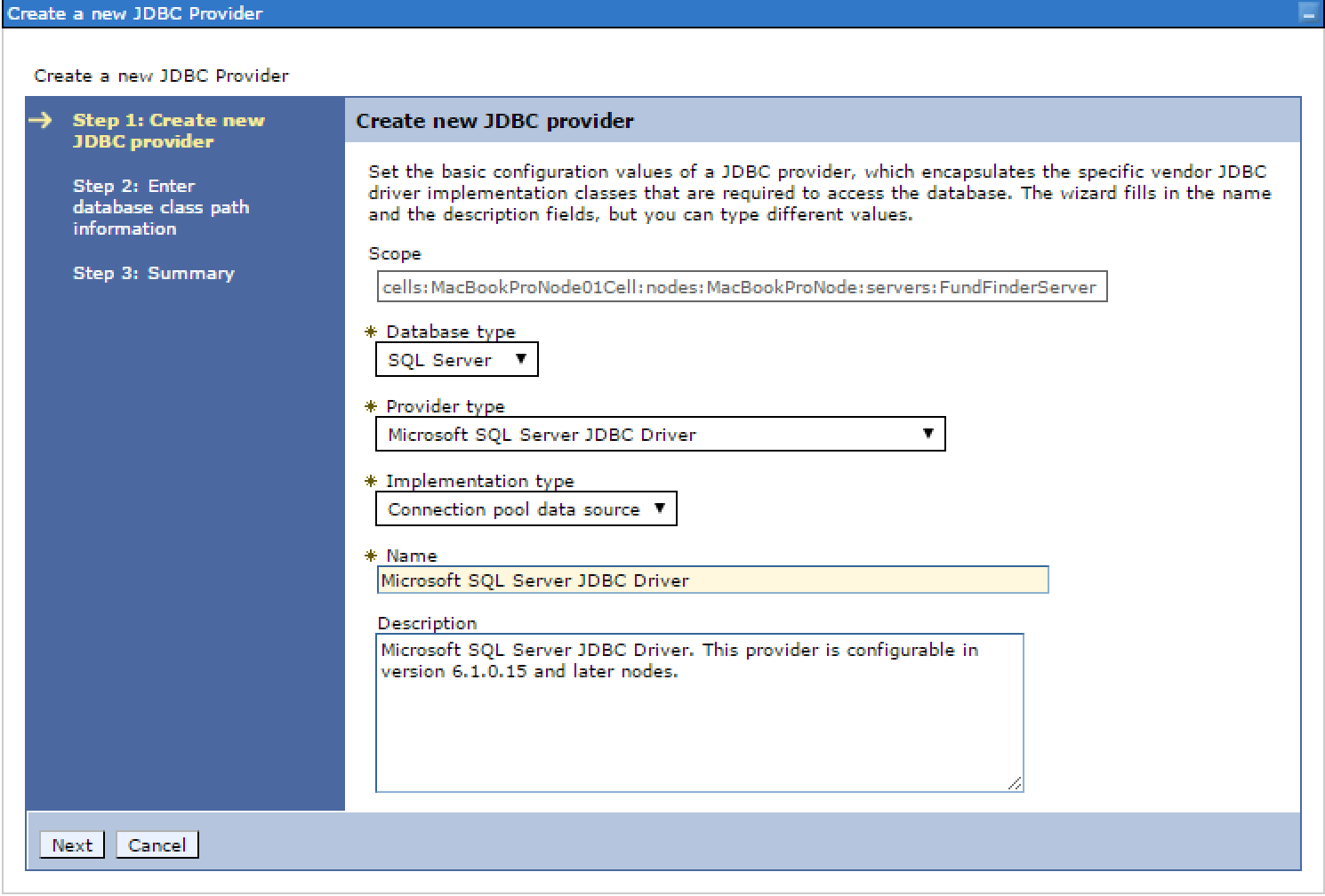
|  |  |
| --- | --- |
| **Variable** | **Description** |
| JAVA\_HOME | As Fund Finder is build with Java 7 this variable should point to **IBM SDK Java 7** installation folder (e.g. ${WAS\_INSTALL\_ROOT}/java\_1.7.64). |
| MICROSOFT\_JDBC\_DRIVER\_PATH | This variable should point to folder where Microsoft JDBC driver (sqljdbc4.jar) can be found. |

### JDBC providers

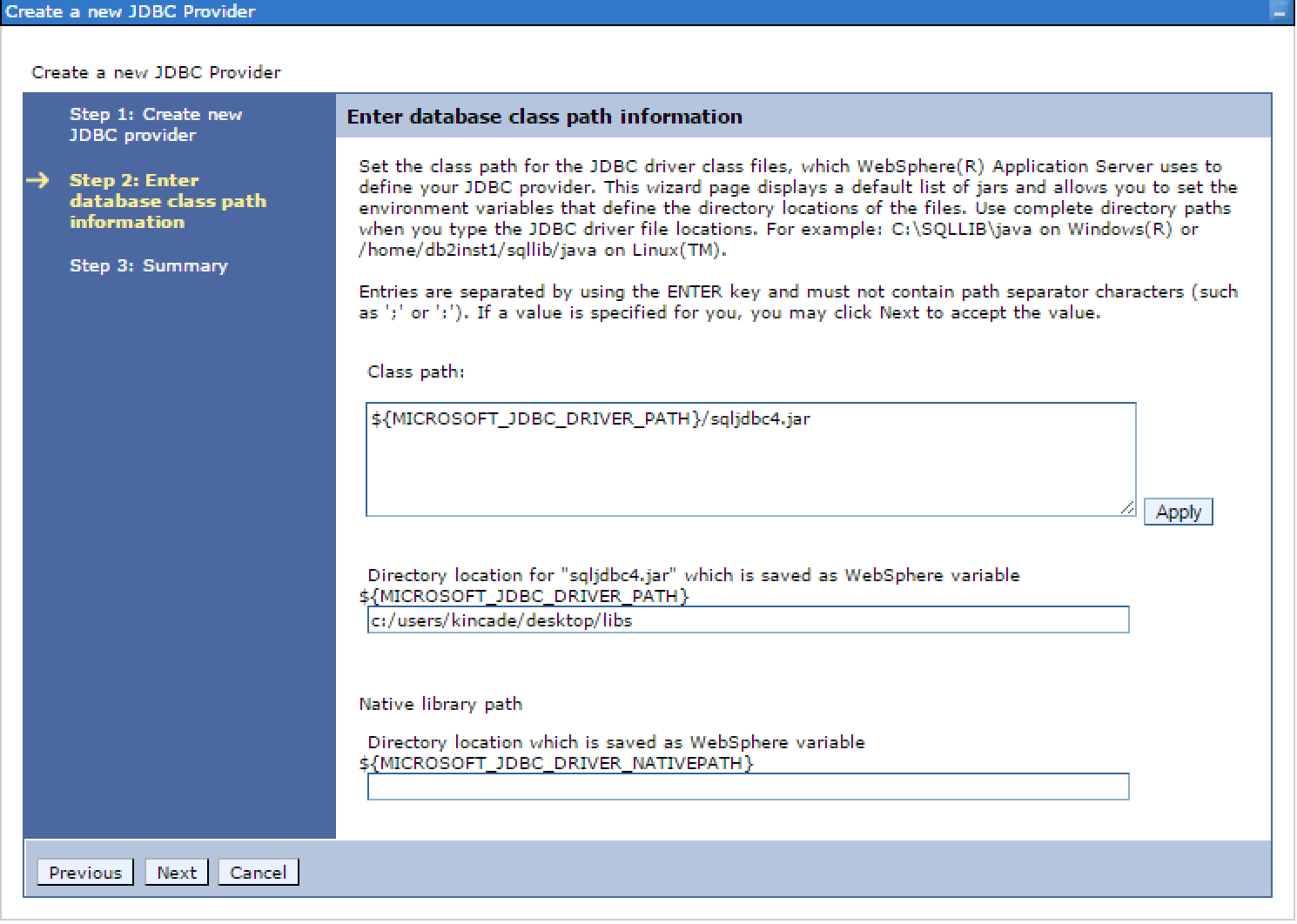
Note: This step is probably already done during the installation of Fund Finder Admin application, and therefore **can be safely skipped**.

In **Administrative console** go to **Resources / JDBC / JDBC providers** and add new JDBC provider:

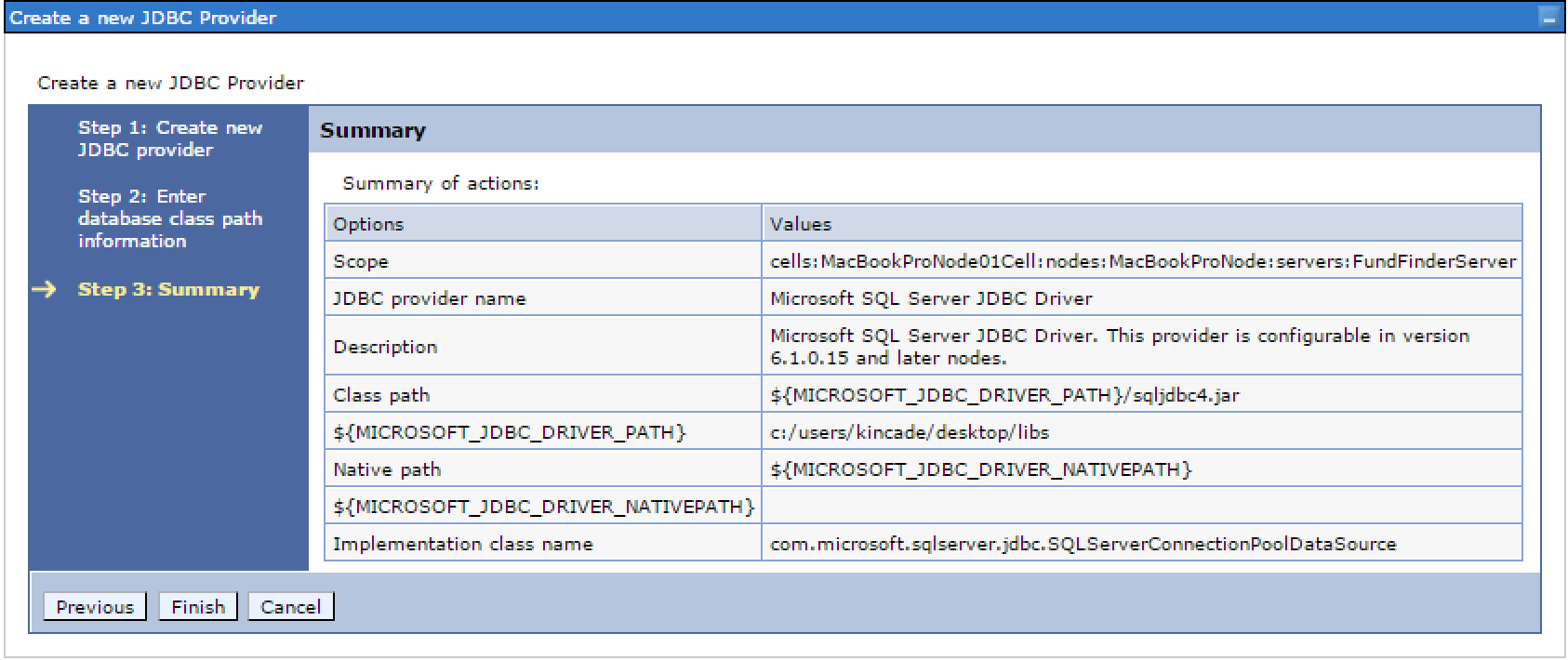
In step 1, select **SQL Server** for **Database type**, **Microsoft SQL Server JDBC Driver** for **Provider type** and **Connection pool data source** for **Implementation type**.



In step 2, set classpath to JAR with Microsoft JDBC drivers for SQL Server.



Last step gives summary. If everything is OK, press *Finished* button.

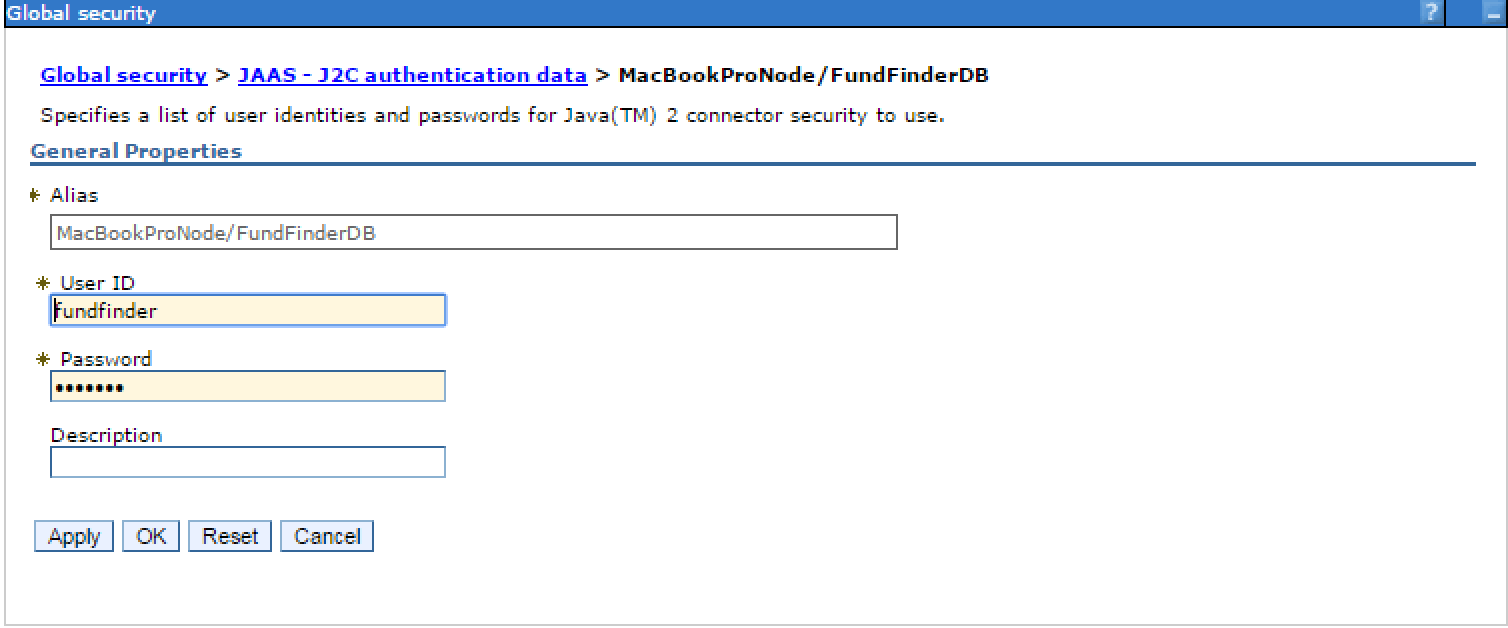


### JAAS - J2C authentication data

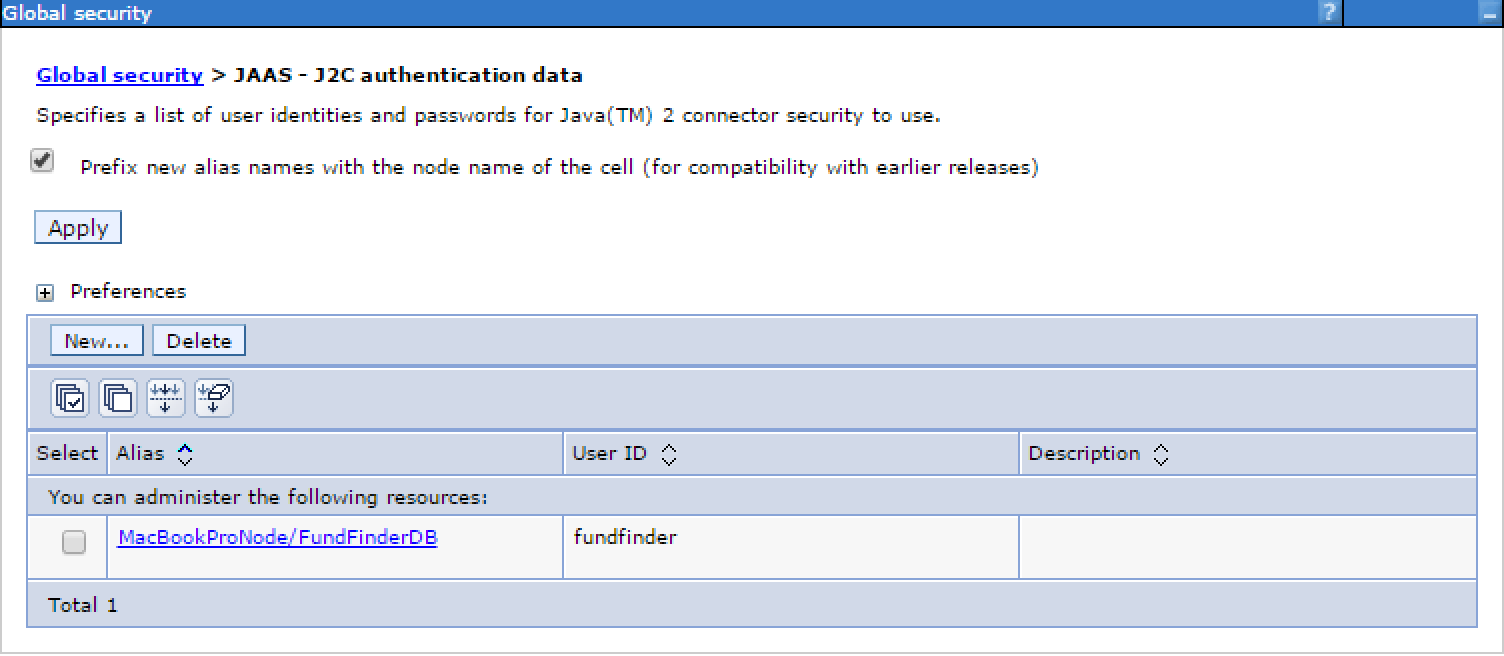
Note: This step is probably already done during the installation of Fund Finder Admin application, and therefore **can be safely skipped**.

In **Administrative console** go to **Security / Global security** and there in **Authentication** section expand **Java Authentication and Authorization Service** and click on **J2C authentication data**.

Here we create new J2C authentication alias that will be required later on when configuring Data source. For **User ID** and **Password** use database authentication credentials.



After the alias is created, you should see it in list of available aliases.

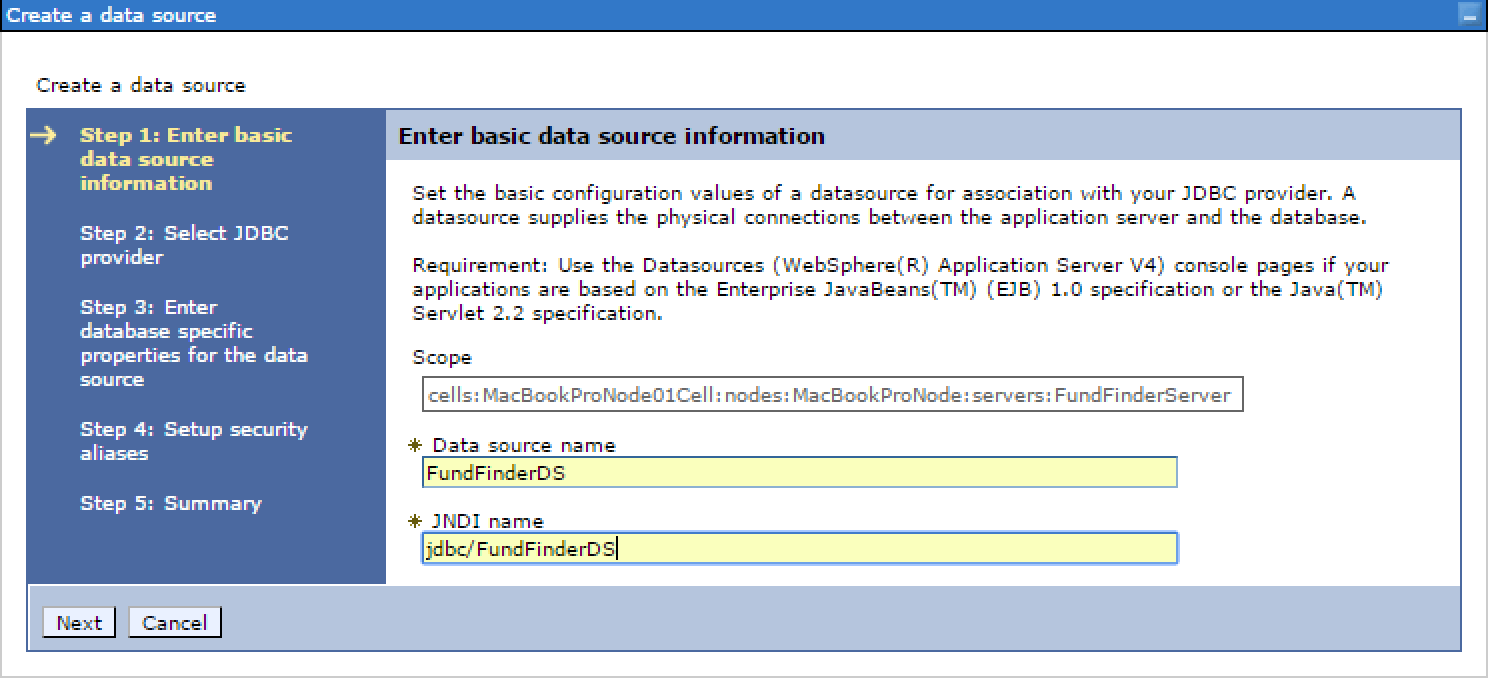


### Data source

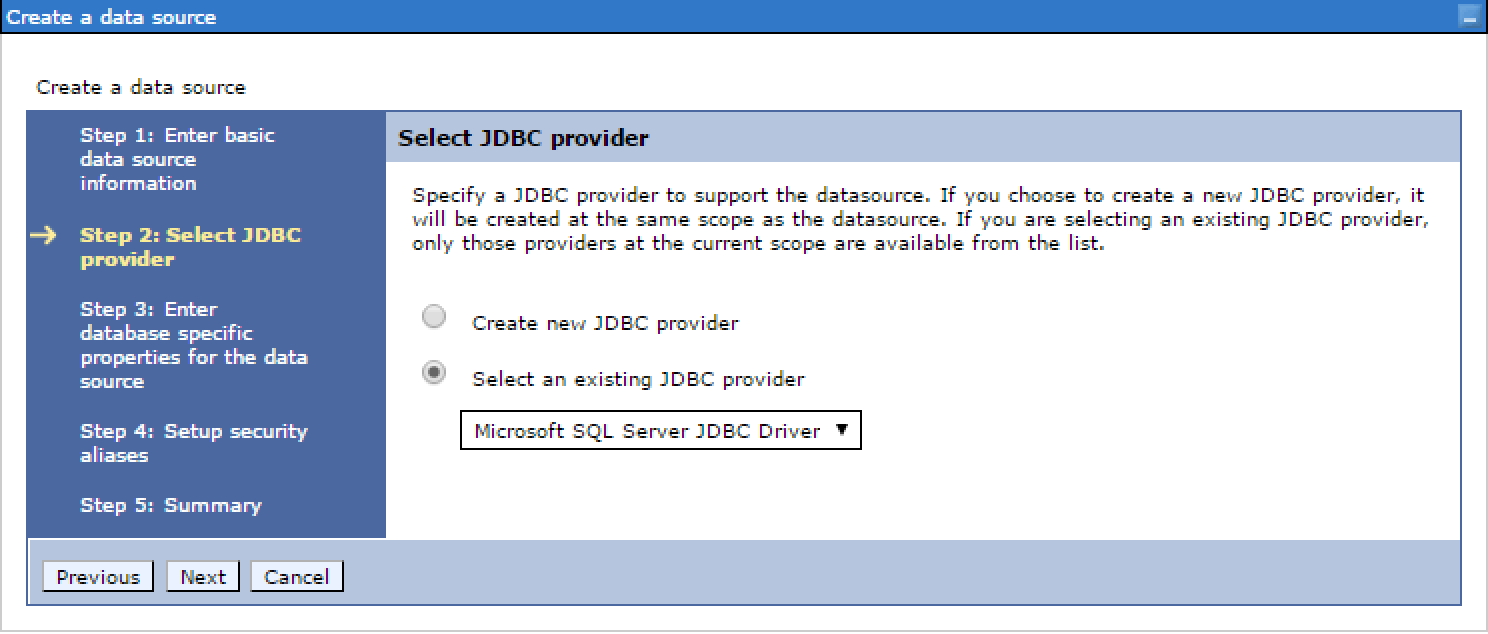
Note: This step is probably already done during the installation of Fund Finder Admin application, and therefore **can be safely skipped**.

In **Administrative console** go to **Resources / JDBC / Data sources** and create new data source.

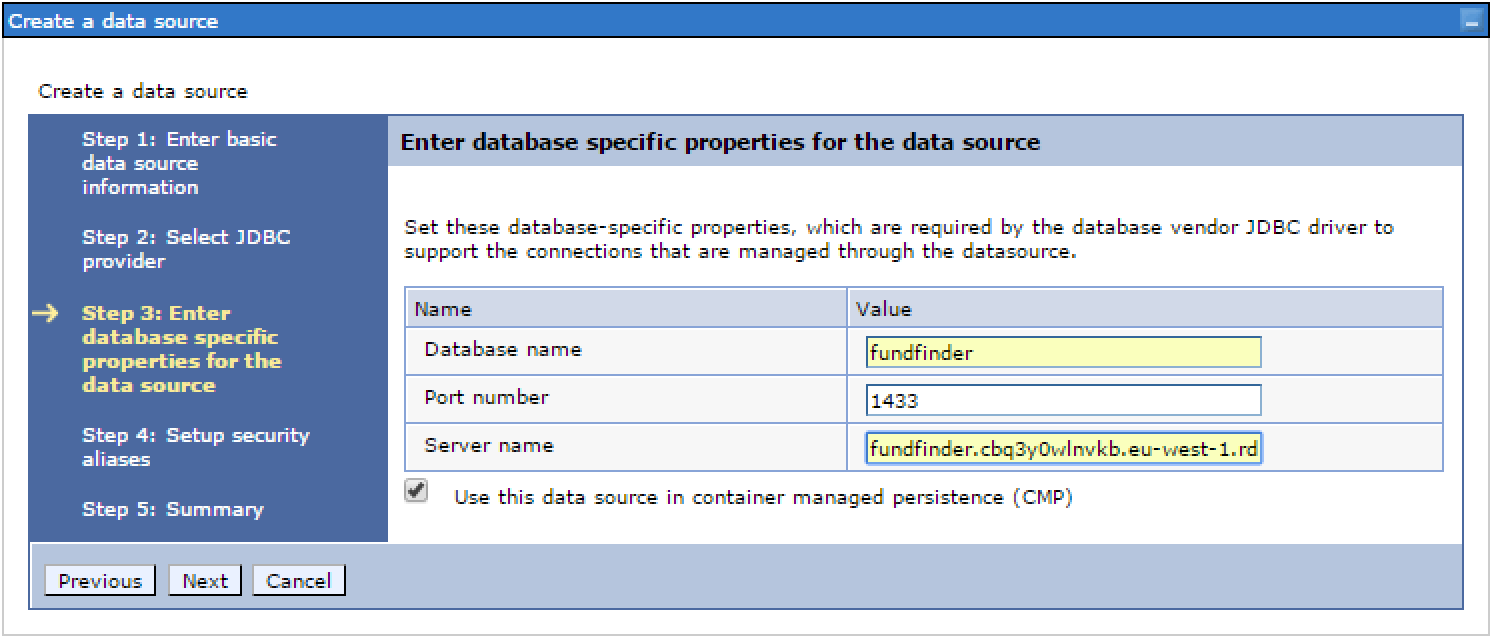
In step 1 we define data source name and JNDI name.



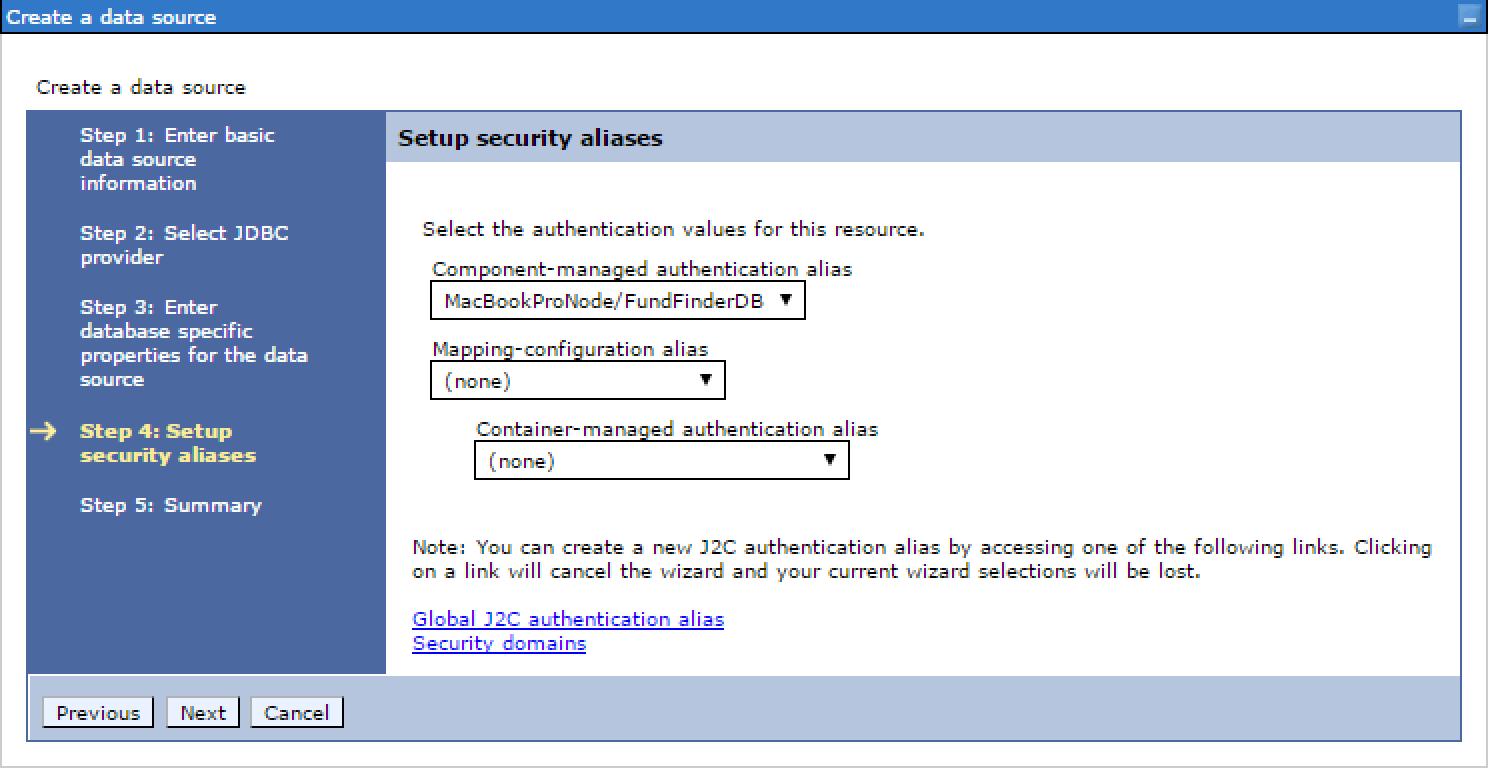
In step 2 we select JDBC provider - in our case the one created in chapter JDBC providers.



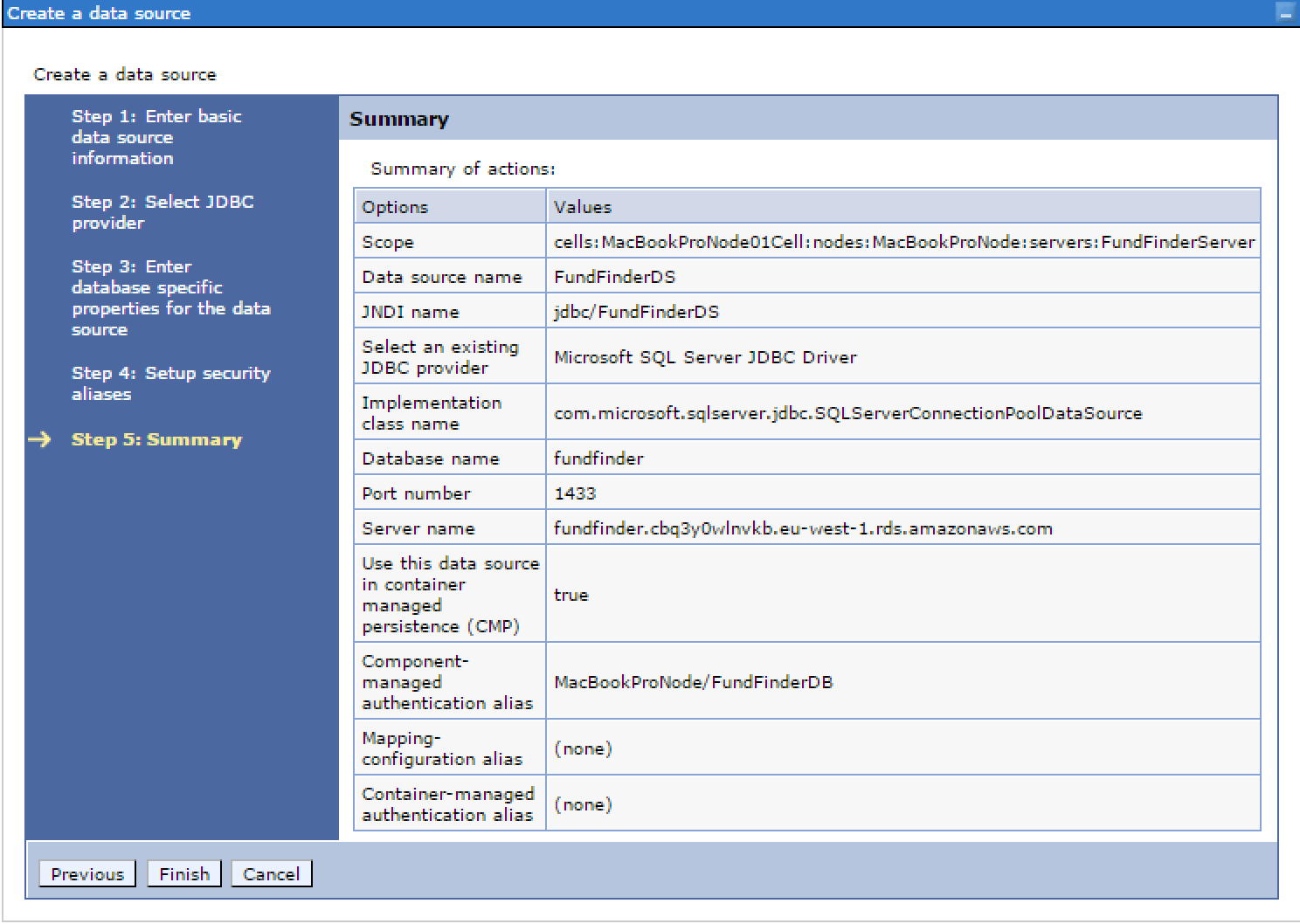
In step 3 we enter database info like database name, server name and port number.



In step 4 we select authentication alias created in chapter JAAS - J2C authentication data.

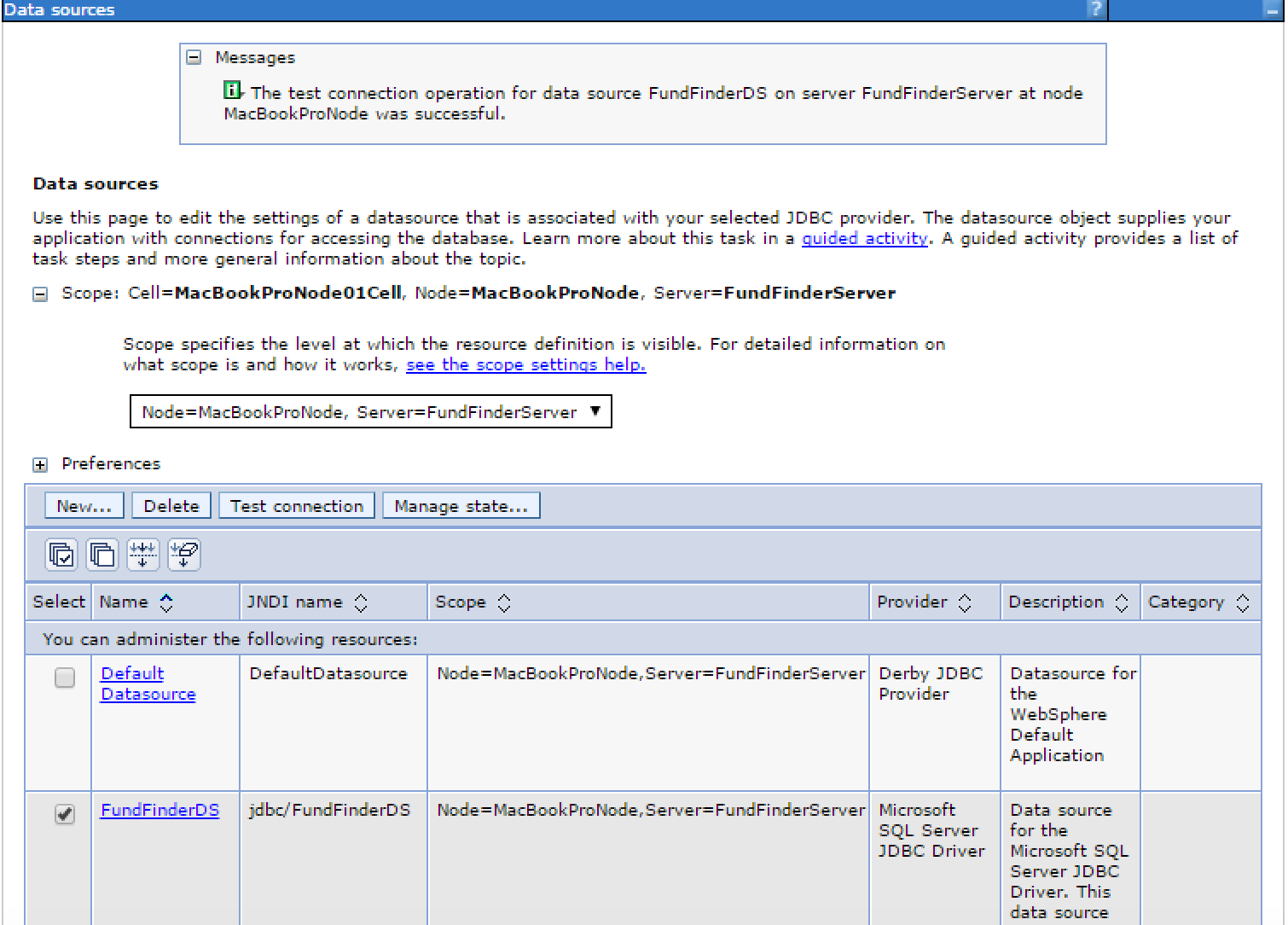


Last step gives summary of data source. If everything is OK, press **Finished** button.



Created data source should now appear in list of data sources.

To test it, select your data source and click on **Test connection** button. If data source if properly configured you should see appropriate message indicating that connection was successful.

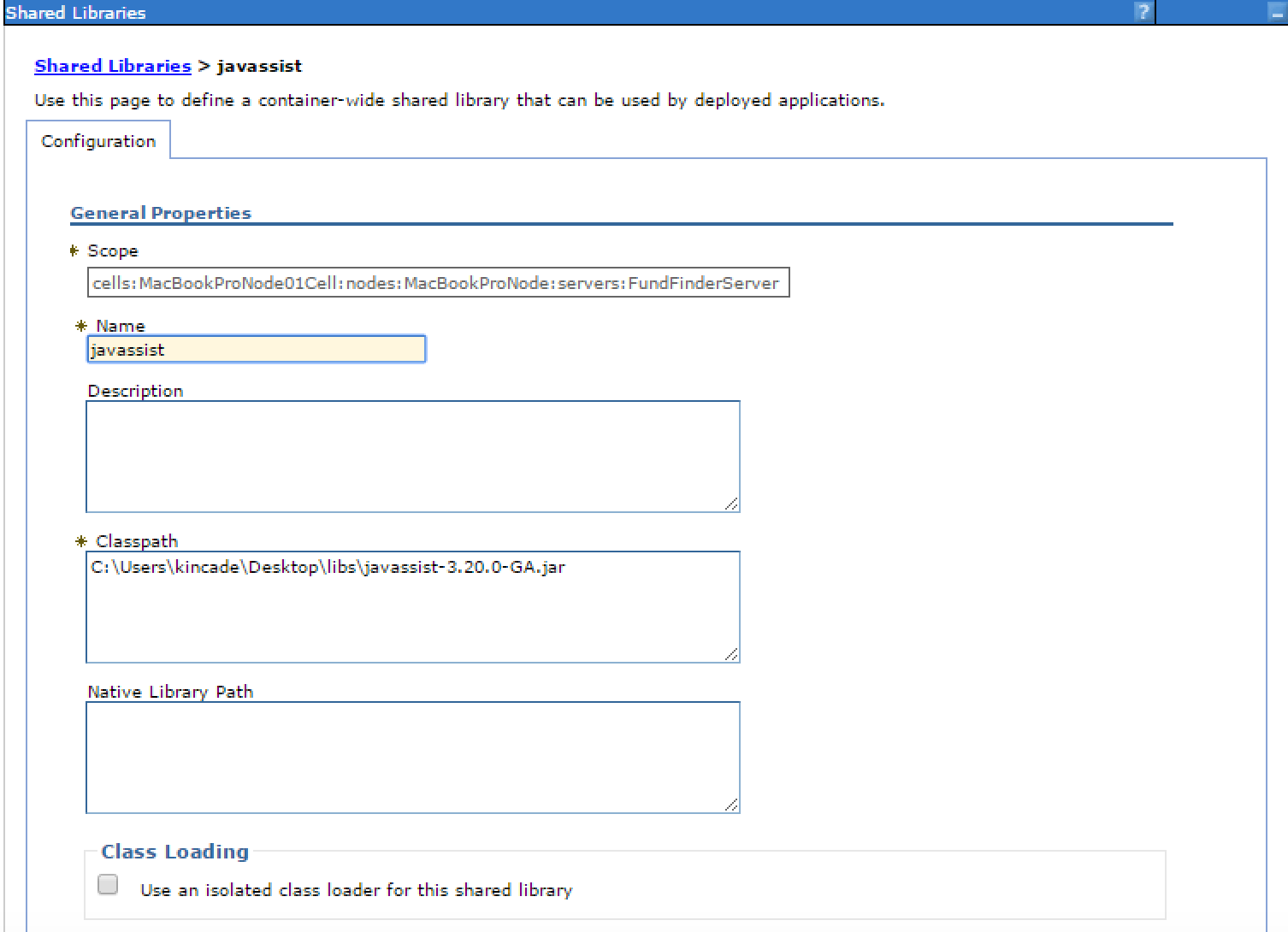


### Shared libraries

Note: It is important that shared folders for Fund Finder User and Fund Finder Admin applications **are separated** physically on the file system and have different logical name in Shared Libraries configuration of WAS (e.g. FFUserShared and FFAdminShared).

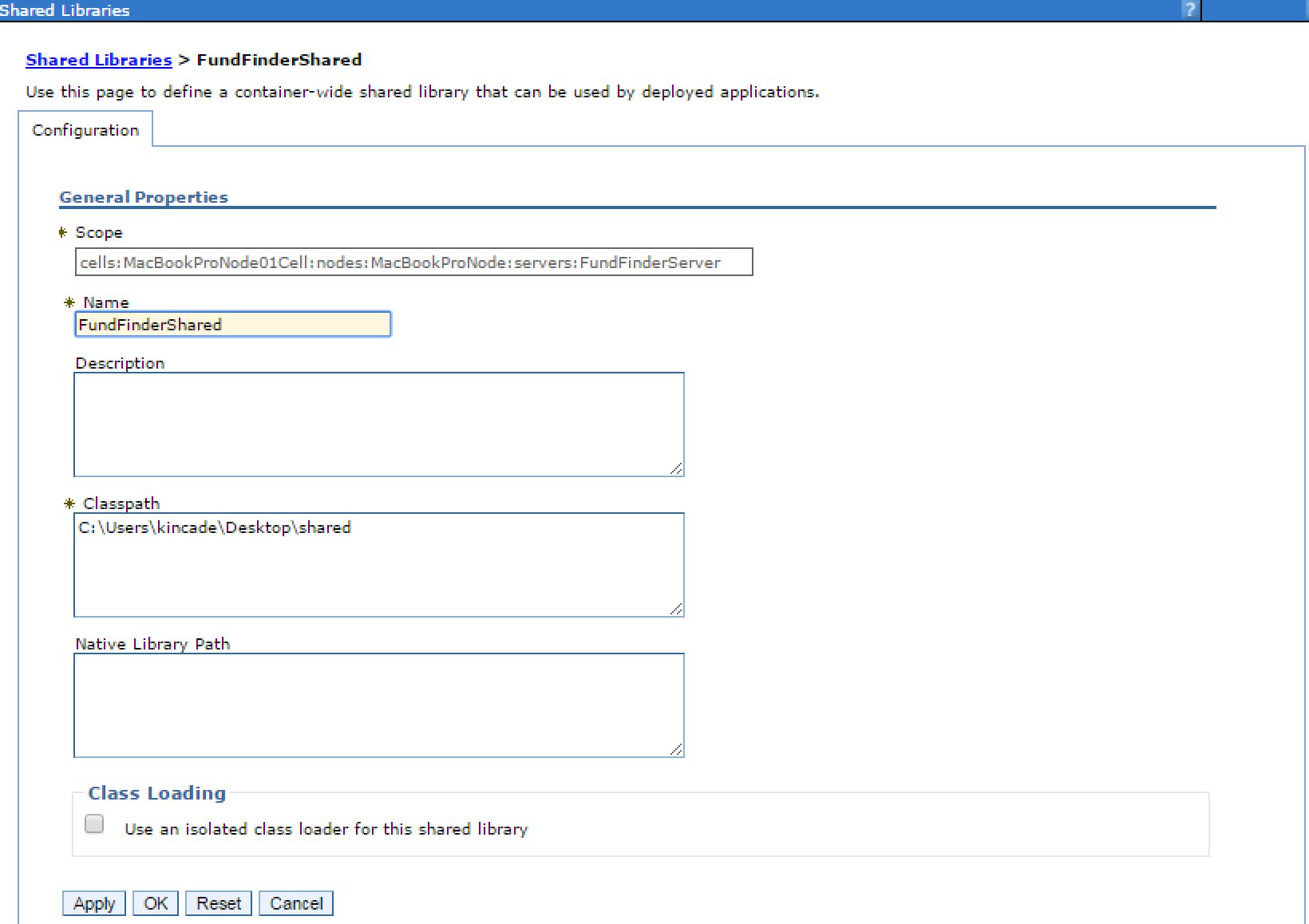
In **Administrative console** go to **Environment / Shared libraries** and add new shared library.

Notice the scope that restricts this shared library only to server where Fund Finder is deployed.

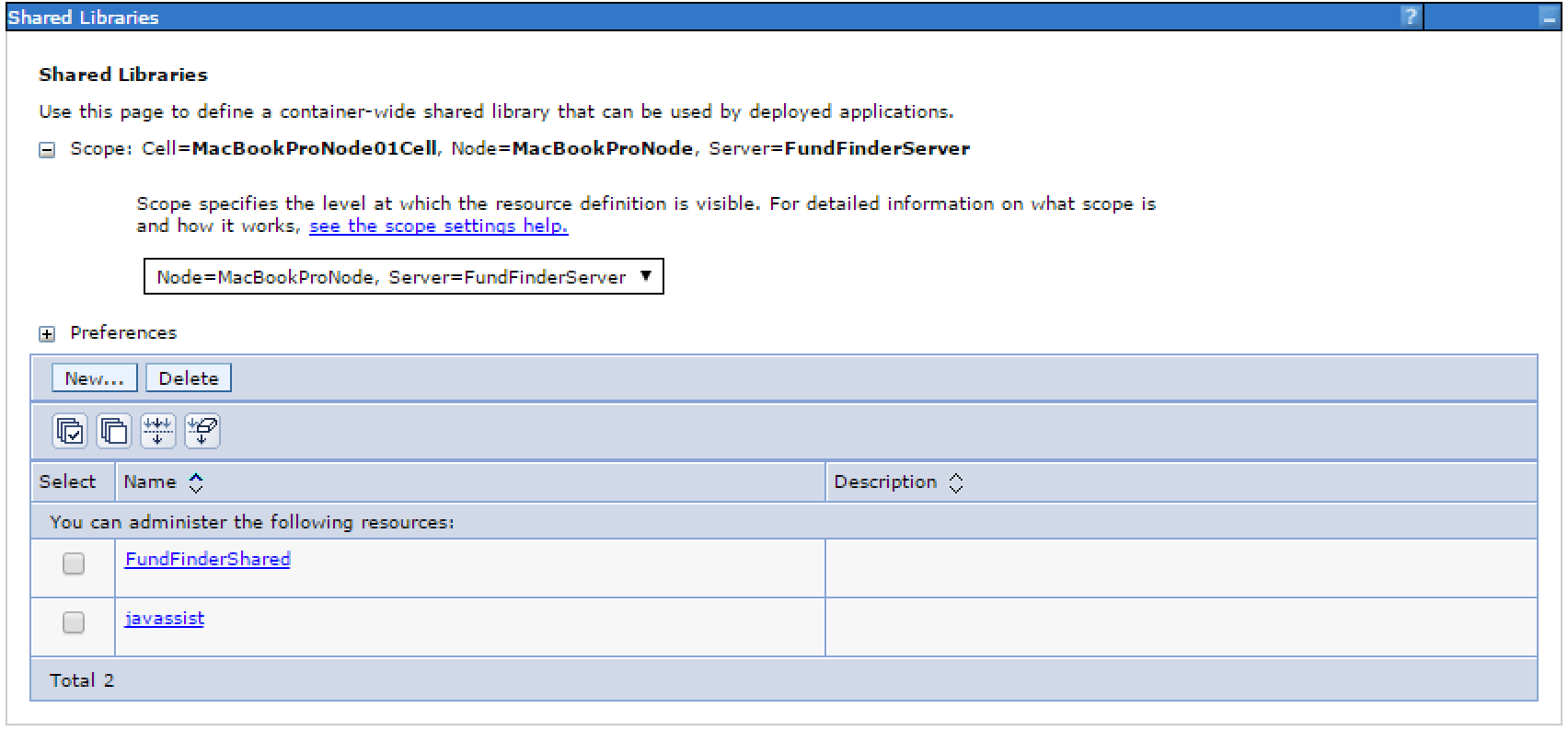


Referenced JAR (javassist-3.20.0-GA.jar) is provided within Delivery package.

Additionally, we need to create another shared library that will point to the **[shared]** folder delivered within delivery package.



After the shared libraries are created you should see them in the list of shared libraries.

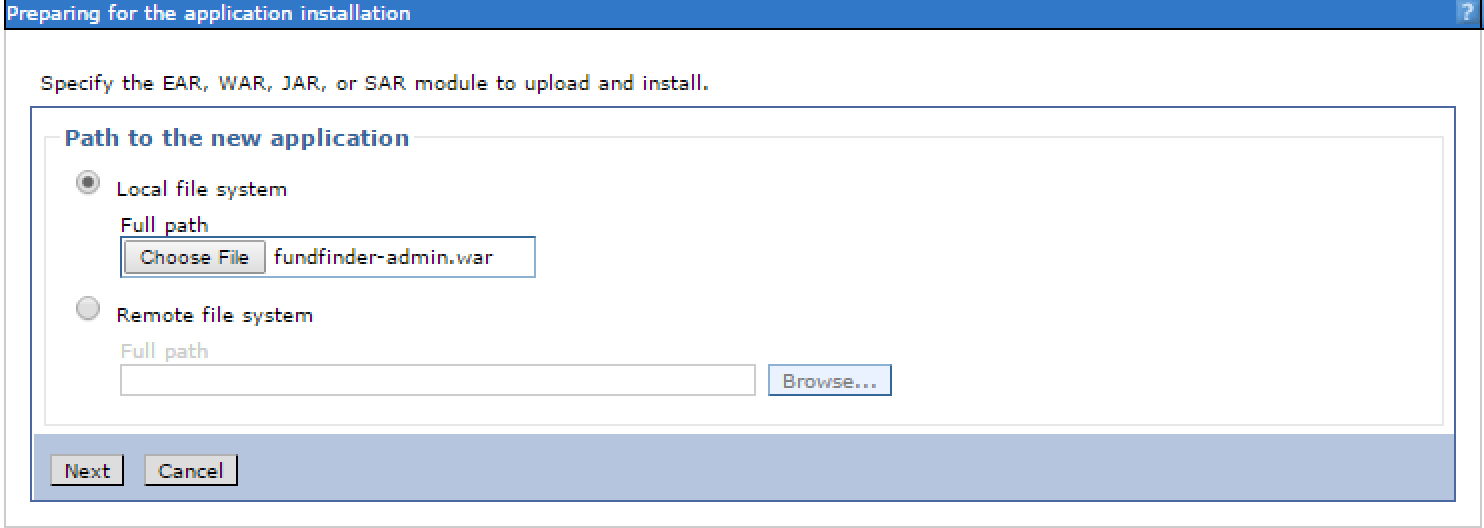


### Deploying application

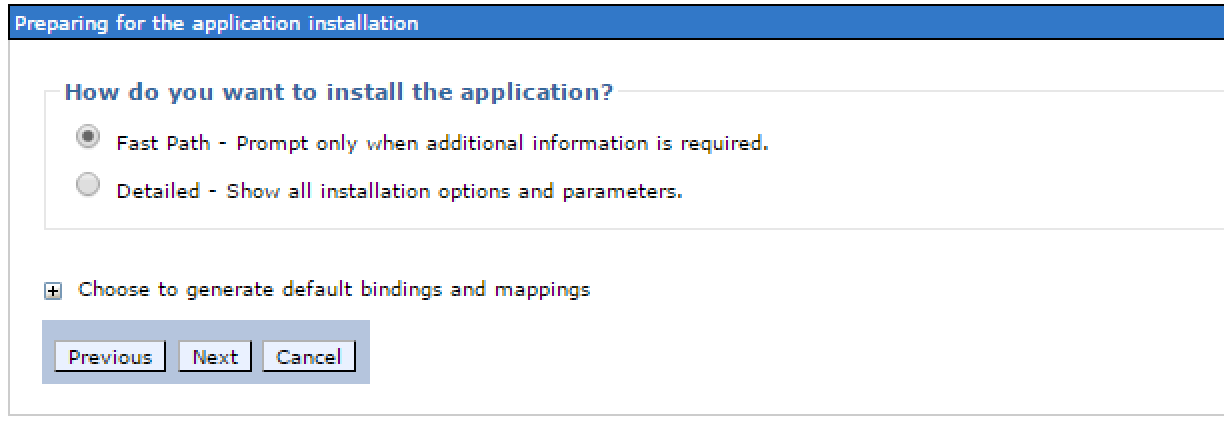
Note: This is the crucial step in installation of Fund Finder User application, so please execute it carefully. Also, note that screenshots below are from deployment of Fund Finder Admin application (fundfinder-admin), but the process is the same (just replace every occurance of fundfinder-admin with fundfinder.

In **Administrative console** go to **Application / New application** and click on **New enterprise application**.

Select **fundfinder.war** from delivery package. Click *Next*.



Note: This can take a while, so be patient.



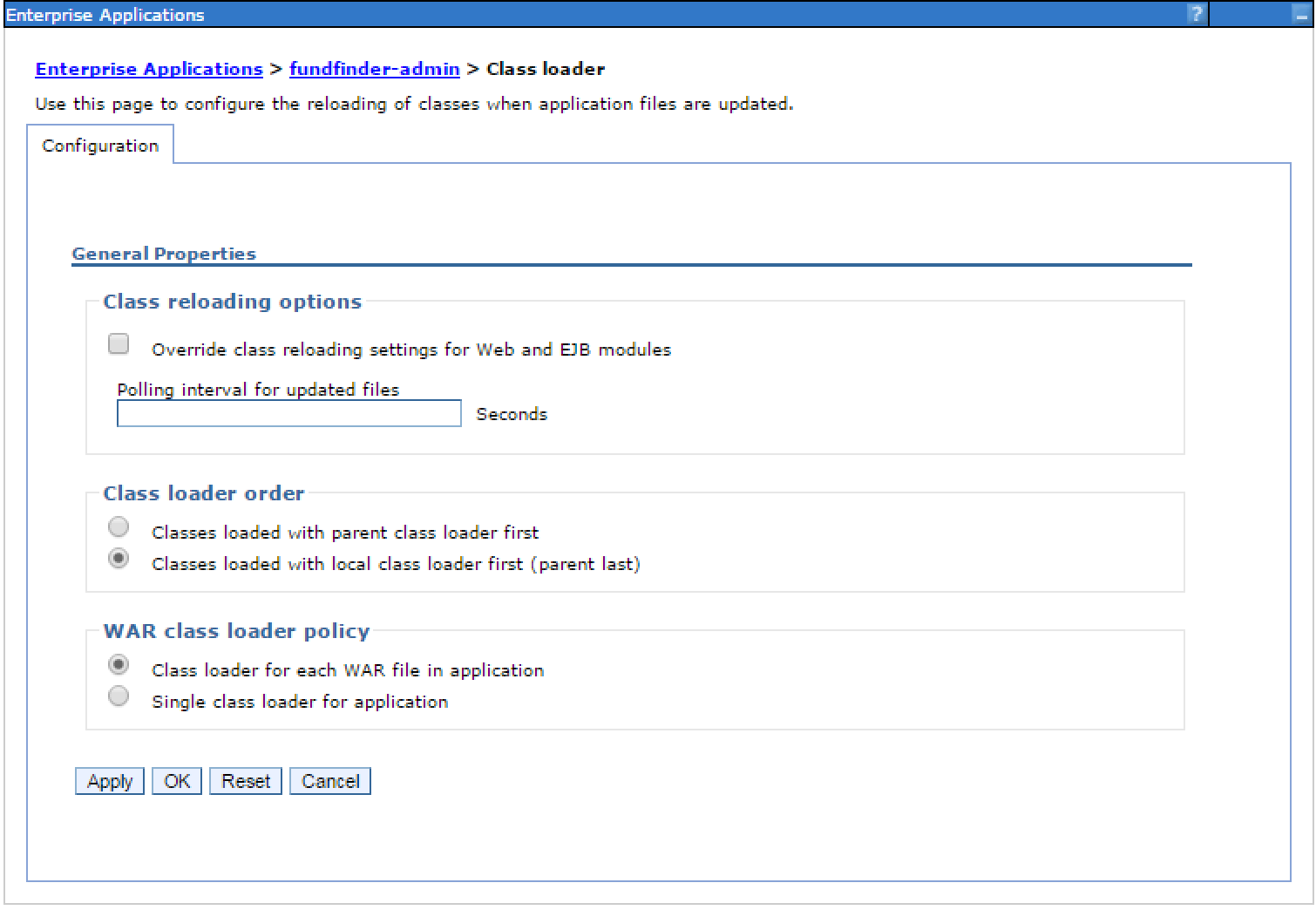
In the first step set the application name to **fundfinder**.

After that just keep clicking the **Next** button until application is created.

Once application is created go to **Application / Application types / WebSphere enterprise applications** and click on ***fundfinder*** to enter edit mode.

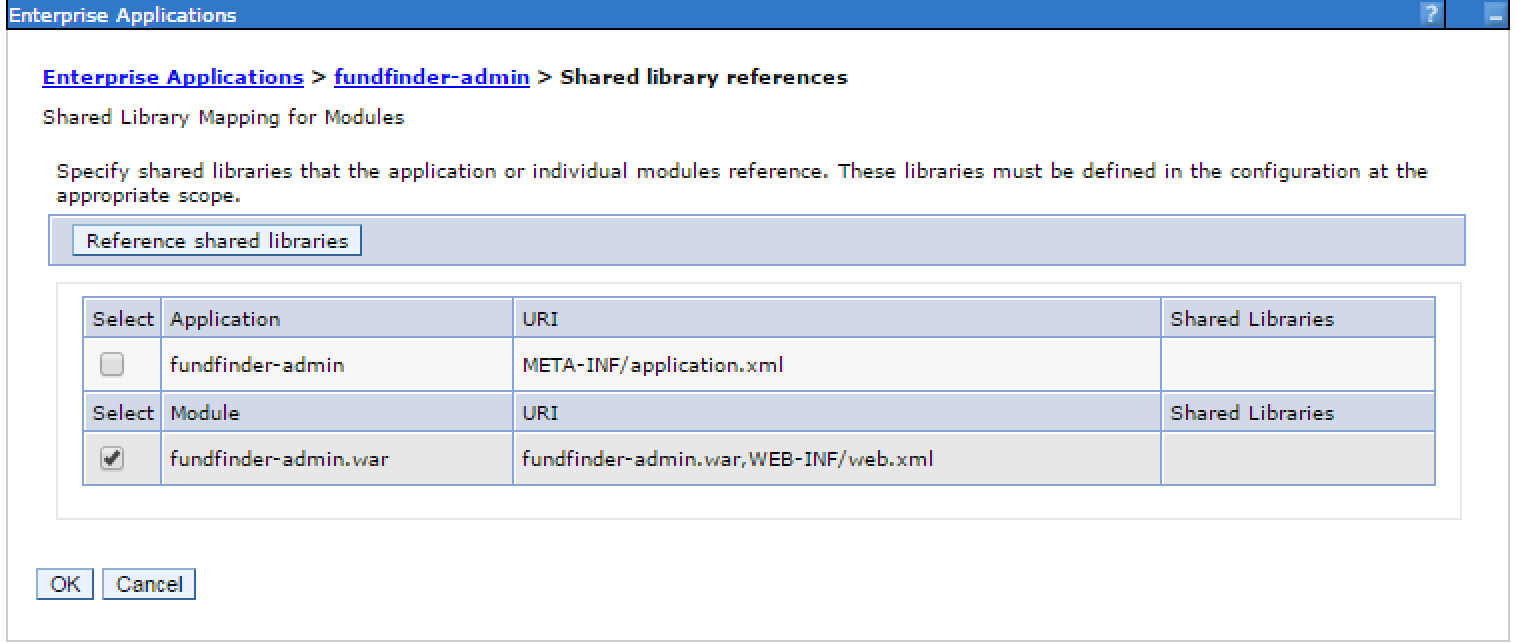
Here, under **Detail Properties**, click on **Class loading and update detection**.

In **Class loader order** section select **Classes loaded with local class loader first (parent last)** and click **OK** button.

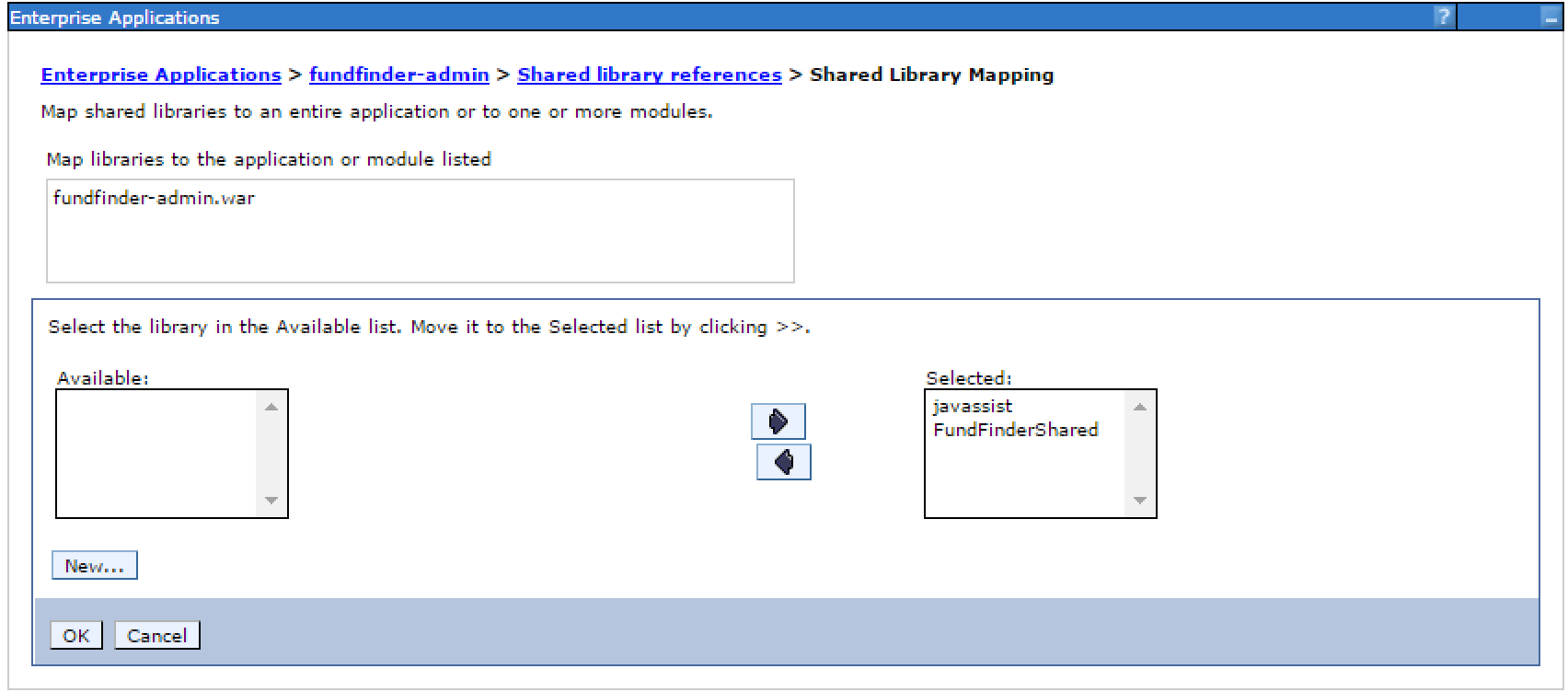


Next, under **References** click on **Shared library references**.

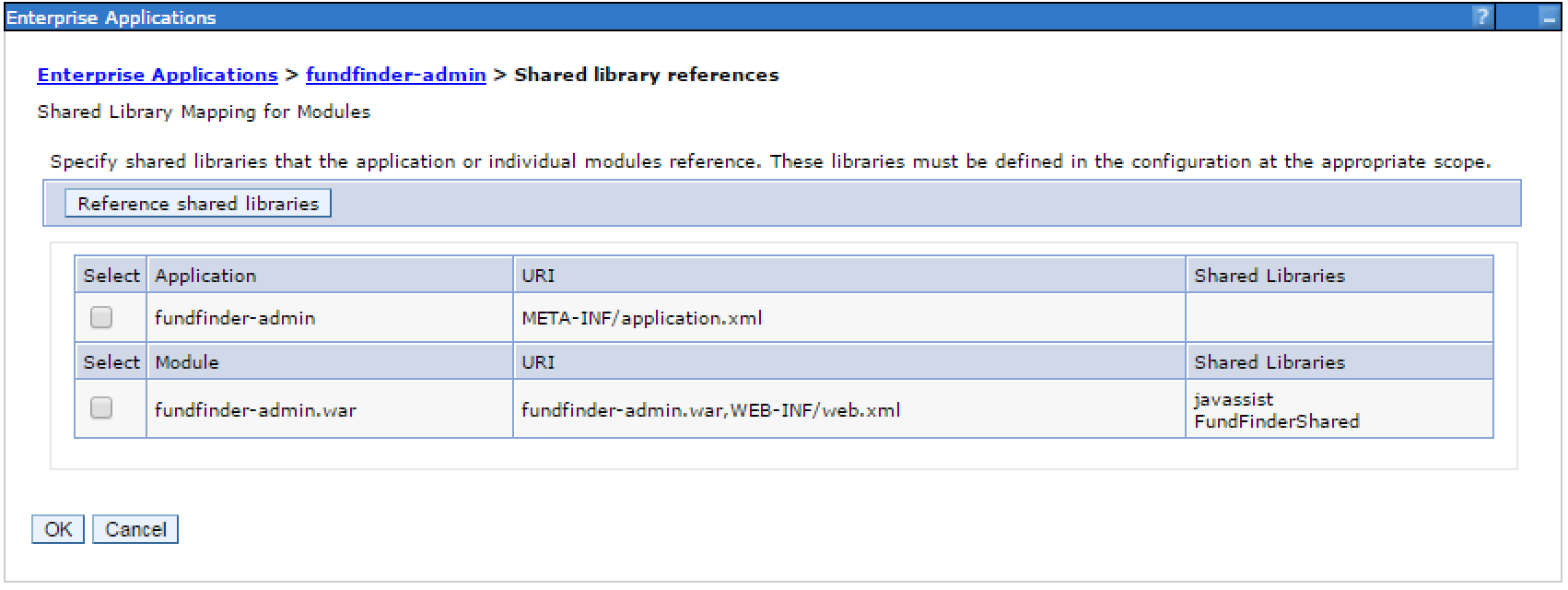
Then, select **fundfinder.war** module and click on **Reference shared libraries**.



Here, select **javassist** and **FundFinderShared**, configured in chapter Shared libraries, map it to selected module (fundfinder.war) and click **OK** button.



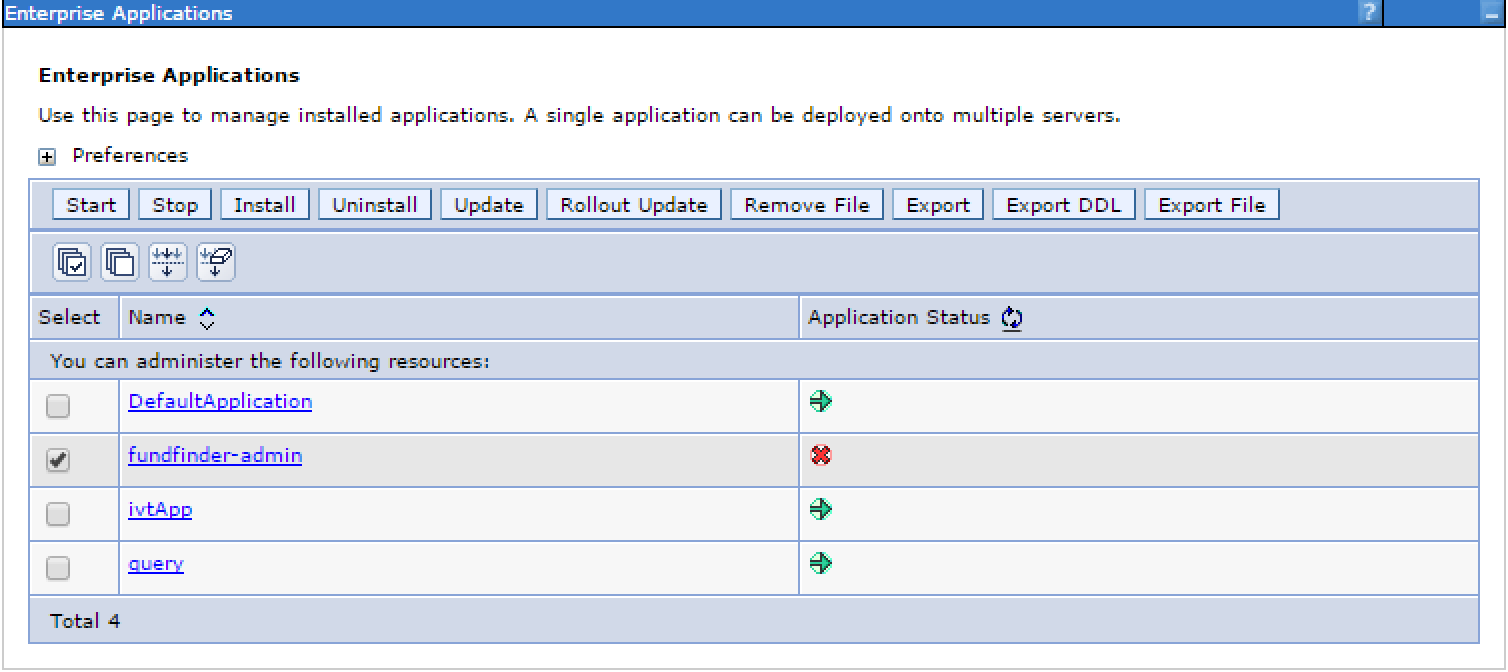
At the end you should see this:



Click **OK** button and save all changes to master configuration.

### Starting application

In **Administrative console** go to **Application / Application types / WebSphere enterprise applications**, select fundfinder and click on **Start** button.



Wait until application is started - to keep track of the progress you can check logs.

Open web browser and enter appropriate URL - e.g. **http://localhost:9080/fundfinder/**