

# CUSTOM EXTERNAL DIRECTORY SERVICES IN MY WEBMETHODS SERVER SAMPLE PROJECT

**Author: Meglena Atanasova, Andrey Lazarov**

Version 1.0 | 20, October, 2016

## TABLE OF CONTENTS

<b>1. Introduction</b>	<b>3</b>
1.1. Configuration overview	3
<b>2. Sample Project with Atlassian Crowd Directory Provider</b>	<b>4</b>
2.1. Requirements	4
2.2. How the attached sample works?	4
2.3. Configuring Crowd	4
2.4. Using the sample project	6
2.5. Creating a universal directory service from scratch	8

## 1. Introduction

In addition to the system directory service in My webMethods Server, you can configure external directory services and connect to directory servers, such as LDAP and ADSI, or to a database.

Since My webMethods Server 9.12, you can also create a directory service that connects to an external data source via custom services, for example using RESTful interfaces.

My webMethods Server now exposes the `IExternalDirDataService` interface. You can provide user data and authentication details through that interface.

In a concrete implementation, you expose the `IExternalDirDataService` interface as an OSGI service, and declare an external directory service, to which you provide the name of the OSGI service.

### 1.1. Configuration overview

1. Configure your directory provider.
2. Create a portlet application, and a class that implements `IExternalDirDataService` to connect to the directory provider.
3. Expose your implementation as an OSGI service.
4. Declare a directory service that connects your implementation to My webMethods Server.

Note: When using CDS, you must provide the name of your `IExternalDirDataService` implementation as the "dataProviderName" attribute of the external directory service. The .jar file for your implementation must be placed in:

**`/IntegrationServer/instances/default/lib/jars/custom/`**

## 2. Sample Project with Atlassian Crowd Directory Provider

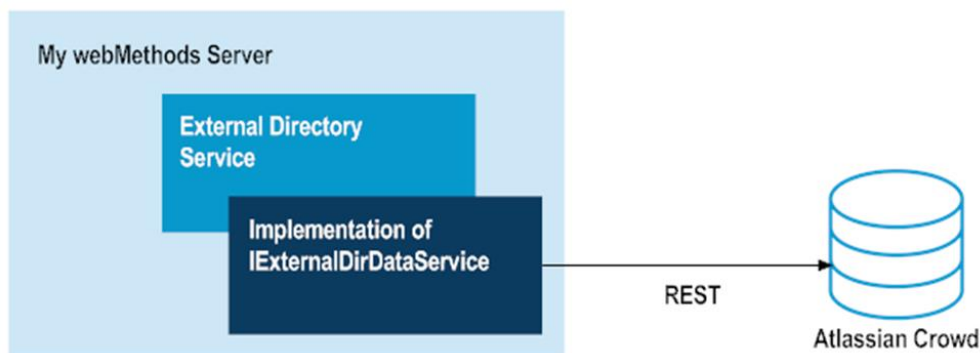
### 2.1. Requirements

The sample project requires:

- Atlassian Crowd - you can download a 30-day trial version from <https://confluence.atlassian.com/crowd>.
- Jersey and Jackson to be available for Integration Server – place the following jars in the custom jars directory:
  - `com.sun.jersey.core`
  - `com.sun.jersey.client`
  - `jackson-core-asl`
  - `jackson-mapper-asl`

### 2.2. How the attached sample works?

The sample project includes the source code for 2 directory service implementations that connect to Atlassian Crowd using REST via Jersey and Jackson.



### 2.3. Configuring Crowd

For detailed instructions, refer to the Atlassian Crowd documentation.

To install Crowd, see <https://confluence.atlassian.com/crowd/installing-crowd-and-crowdid-191862.html>.




To create a directory in Crowd, see <https://confluence.atlassian.com/crowd/configuring-an-internal-directory-18579551.html>.

To add groups, see <https://confluence.atlassian.com/crowd/adding-a-group-20807693.html>. Nested groups are supported.

To add users, see <https://confluence.atlassian.com/crowd/adding-a-user-18579611.html>.




The provided sample includes the source files to create two custom services - **Crowd Sample Directory** and **Crowd Second Directory**. For the sample project, we created corresponding directories in Crowd, and added groups and users to those directories, as described below.

We added the following groups to the first directory:

<u>GROUP NAME</u>	<u>GROUP ID</u> 	<u>E-MAIL</u>	<u>DIRECTORY SERVICE</u>	<u>EDIT</u>
<a href="#">group1</a>	group1		Crowd Sample Directory	
<a href="#">group1_1</a>	group1_1		Crowd Sample Directory	
<a href="#">group2</a>	group2		Crowd Sample Directory	




**Note that group1\_1 is nested in group1.**


We added the following groups to the second directory:

<u>GROUP NAME</u>	<u>GROUP ID</u> 	<u>E-MAIL</u>	<u>DIRECTORY SERVICE</u>	<u>EDIT</u>
<a href="#">group2_second</a>	group2_second		Crowd Second Directory	
<a href="#">group3_second</a>	group3_second		Crowd Second Directory	




« Previous | 1 | Next »

Then we added the following users in the first directory:

<u>USER ID</u> 	<u>FIRST NAME</u>	<u>LAST NAME</u>	<u>E-MAIL</u>	<u>DIRECTORY SERVICE</u>	<u>EDIT</u>
<a href="#">user1</a>	<a href="#">user1</a>	<a href="#">user1</a>	vmtest@softwareag.com	Crowd Sample Directory	
<a href="#">user2</a>	<a href="#">user2</a>	<a href="#">user2</a>	vmtest@softwareag.com1	Crowd Sample	

<u>USER ID</u> ▲	<u>FIRST NAME</u>	<u>LAST NAME</u>	<u>E-MAIL</u>	<u>DIRECTORY SERVICE</u>	<u>EDIT</u>
				Directory	
<a href="#">user3</a>	<a href="#">user3</a>	<a href="#">user3</a>	vmtest@softwareag.com	Crowd Sample Directory	

And the following users to the second directory:

<a href="#">user1_second</a>	<a href="#">user1_second</a>	<a href="#">user1_second</a>	vmtest@softwareag.com	Crowd Second Directory	
<a href="#">user2_second</a>	<a href="#">user2_second</a>	<a href="#">user2_second</a>	vmtest@softwareag.com	Crowd Second Directory	
<a href="#">user3_second</a>	<a href="#">user3_second</a>	<a href="#">user3_second</a>	vmtest@softwareag.com	Crowd Second Directory	

« Previous | 1 | Next »

## 2.4. Using the sample project

After configuring Crowd, you can deploy and test the provided sample project, or create a custom universal directory service from scratch (see 2.5.).

To compile the sample project, add `wm-directory-components.jar` from `common/lib` to your build path.

The sample implementation registers two different directory services that connect to the same Crowd server as different applications. These can be used to access different Crowd directories. An OSGI component is registered for each service.

## Sample Crowd Services in My webMethods:

Directory Services Administration			
List Directory Services	Modify Directory Search Order	Create New Directory Service	
Type	Name	Description	Modified On
☒	system		3/28/2016 5:23 PM
☒	Crowd_Sample_Directory		3/29/2016 10:02 AM
☒	Crowd_Second_Directory		3/29/2016 10:02 AM
☒	DummyExtDirDataService		3/29/2016 12:57 PM

Items 1 - 4 of 4

The BND file under OSGI-OPT includes instructions about the location of service components. These instructions customize OSGI bundle manifest, generated when deploying the project .war file to My webMethods Server.

## Editing a sample Crowd directory service in webMethods:

Directory Services > Crowd Sample Directory

Properties of Crowd Sample Directory

**General**

\* Name: Crowd Sample Directory

Service Enabled: Yes. This service is enabled Whether this service is active

\* Name of the data provider: com.softwareag.example.dirservice.crowd.CrowdD Name of the component that is providing a data provider that implements the IExternalDirDataService interface.

Configuration data: http://localhost:8095/crowd##mws##mws Optional configuration data that will be passed to the data provider.

---

**Cache**

Cache Enabled: Yes, Enable caching

Cache Capacity: 1000

Cache Timeout: Forever Length of time that entries may be cached

---

**Display**

Full Page View: Default

Portlet View: Default

---

**Maintenance**

Owner: sysadmin

Created On: Tuesday, March 29, 2016 10:02:20 AM EEST

Modified On: Tuesday, March 29, 2016 10:02:20 AM EEST

Aliases: dir.crowd.sample Add Edit Remove

To use the services with Integration Server, add a .jar file with your project (in WEB-INF/lib after export, as .war) to the following Integration Server directory:

**IntegrationServer/instances/<instance\_name>/lib/jars/custom.**

## Searching for users in Atlassian Crowd from My webMethods:

USER ID	FIRST NAME	last name	email	DIRECTORY SERVICE
Administrator	My webMethods			system
DeletedItems	Deleted Items			system
Designer	Designer			system
Guest	Guest			system
SysAdmin	Sys Admin			system
user0EXT_TEST	firstName: user0EXT_TEST	lastName: user0EXT_TEST	user0EXT_TEST@mail.com	DummyExtDirDataService
user1	user1	user1	vmtest@softwareag.com	Crowd Sample Directory
user1_second	user1_second	user1_second	vmtest@softwareag.com	Crowd Second Directory
user1EXT_TEST	firstName: user1EXT_TEST	lastName: user1EXT_TEST	user1EXT_TEST@mail.com	DummyExtDirDataService
user2	user2	user2	vmtest@softwareag.com1	Crowd Sample Directory
user2_second	user2_second	user2_second	vmtest@softwareag.com	Crowd Second Directory
user2EXT_TEST	firstName: user2EXT_TEST	lastName: user2EXT_TEST	user2EXT_TEST@mail.com	DummyExtDirDataService
user3	user3	user3	vmtest@softwareag.com	Crowd Sample Directory
user3_second	user3_second	user3_second	vmtest@softwareag.com	Crowd Second Directory
user3EXT_TEST	firstName: user3EXT_TEST	lastName: user3EXT_TEST	user3EXT_TEST@mail.com	DummyExtDirDataService
user4EXT_TEST	firstName: user4EXT_TEST	lastName: user4EXT_TEST	user4EXT_TEST@mail.com	DummyExtDirDataService
user5EXT_TEST	firstName: user5EXT_TEST	lastName: user5EXT_TEST	user5EXT_TEST@mail.com	DummyExtDirDataService
user6EXT_TEST	firstName: user6EXT_TEST	lastName: user6EXT_TEST	user6EXT_TEST@mail.com	DummyExtDirDataService
user7EXT_TEST	firstName: user7EXT_TEST	lastName: user7EXT_TEST	user7EXT_TEST@mail.com	DummyExtDirDataService
user8EXT_TEST	firstName: user8EXT_TEST	lastName: user8EXT_TEST	user8EXT_TEST@mail.com	DummyExtDirDataService

The external directory service framework provides the "configData" parameter. This string parameter is passed to the setConfigData implementation for custom service configuration. In the sample application we use "##" for a separator to pass multiple parameters as a single string, as follows:

```
configData="http://localhost:8095/crowd##mws##mws"
```

You can utilize the "configData" parameter for your custom provider implementation, for example, for supplying an external configuration file.

## 2.5. Creating a universal directory service from scratch

After configuring Crowd, or another directory provider, you can create a custom universal directory services, following the steps below:

1. Create a portlet application, and a class that implements `IExternalDirDataService`:

```
package com.softwareag.univdirsvc;

import softwareag.dir.extdirsvc.datamodel.IExternalDirDataService;

public class UniversalDirectoryService implements IExternalDirDataService {

}
```

2. Register your service as an OSGI service by adding an xml file in the OSGI-INF directory of the project:

```
<?xml version="1.0" encoding="UTF-8"?>
<scr:component name="Crowd Sample Directory" xmlns:scr="http://www.osgi.org/xmlns/scr/v1.1.0">
  <service>
    <provide interface='com.softwareag.dir.extdirsvc.datamodel.IExternalDirDataService' />
  </service>
  <implementation class='com.softwareag.example.dirservice.crowd.CrowdDirectoryService' />
</scr:component>
```

3. Create a BND file (bnd.bnd) in the OSGI-OPT directory of your project to modify the manifest:

```
Service-Component: OSGI-INF/*
```

4. Declare a directory service that uses your concrete implementation - `concrete_impl\WebContent\WEB-INF\config\xmlImport.xml`

```
<CONFIG>
  <!-- create a new page for our portlet application -->
  <context alias="dir.svc.type.container">
    <wm_xt_extdirsvc name="Sample Atlassian Crowd Dir Service" skipValidation="true"/>
  </context>

  <context alias="dir.svc.container">
    <wm_xt_extdirsvc
      alias="dir.crowd.sample"
      name="Crowd Sample Directory"
      dataProviderName="com.softwareag.example.dirservice.crowd.CrowdDirectoryService"
      configData="http://localhost:8095/crowd##mws##mws"
      cacheEnabled="true" />
    <wm_xt_extdirsvc
      alias="dir.crowd.second"
      name="Crowd Second Directory"
      dataProviderName="com.softwareag.example.dirservice.crowd.CrowdDirectoryService"
      configData="http://localhost:8095/crowd##secret##secret"
      cacheEnabled="true" />
  </context>
</CONFIG>
```



---

#### ABOUT SOFTWARE AG

Software AG offers the world's first Digital Business Platform. Recognized as a leader by the industry's top analyst firms, Software AG helps you combine existing systems on premises and in the cloud into a single platform to optimize your business and delight your customers. With Software AG, you can rapidly build and deploy Digital Business Applications to exploit real-time market opportunities. Get maximum value from big data, make better decisions with streaming analytics, achieve more with the Internet of Things, and respond faster to shifting regulations and threats with intelligent governance, risk and compliance. The world's top brands trust Software AG to help them rapidly innovate, differentiate and win in the digital world. Learn more at [www.SoftwareAG.com](http://www.SoftwareAG.com).

© 2015 Software AG. All rights reserved. Software AG and all Software AG products are either trademarks or registered trademarks of Software AG. Other product and company names mentioned herein may be the trademarks of their respective owners.