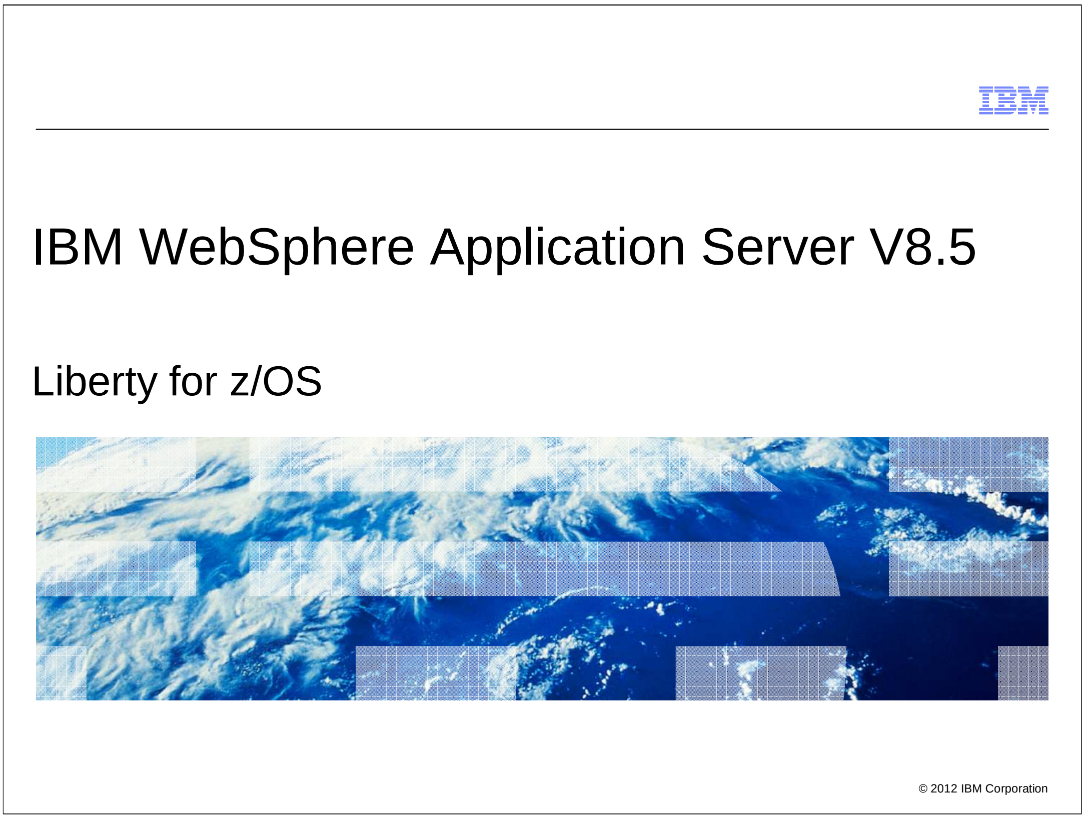
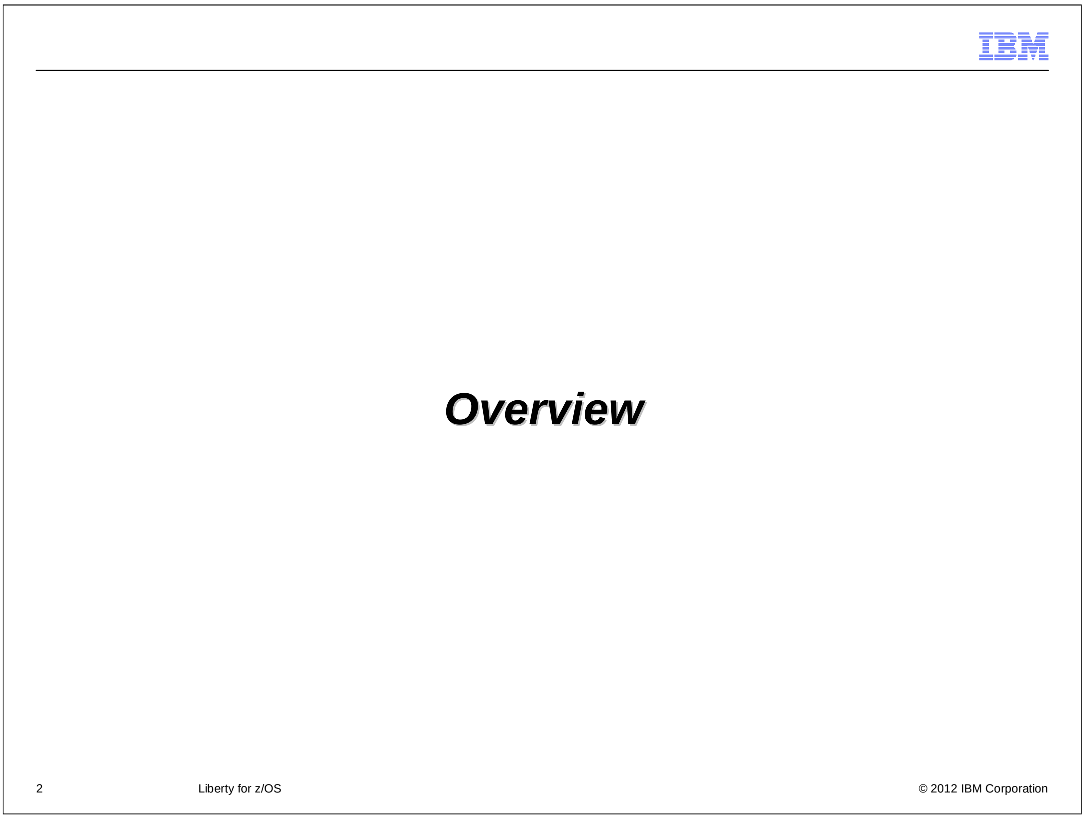


IBM WebSphere Application Server V8.5

Liberty for z/OS

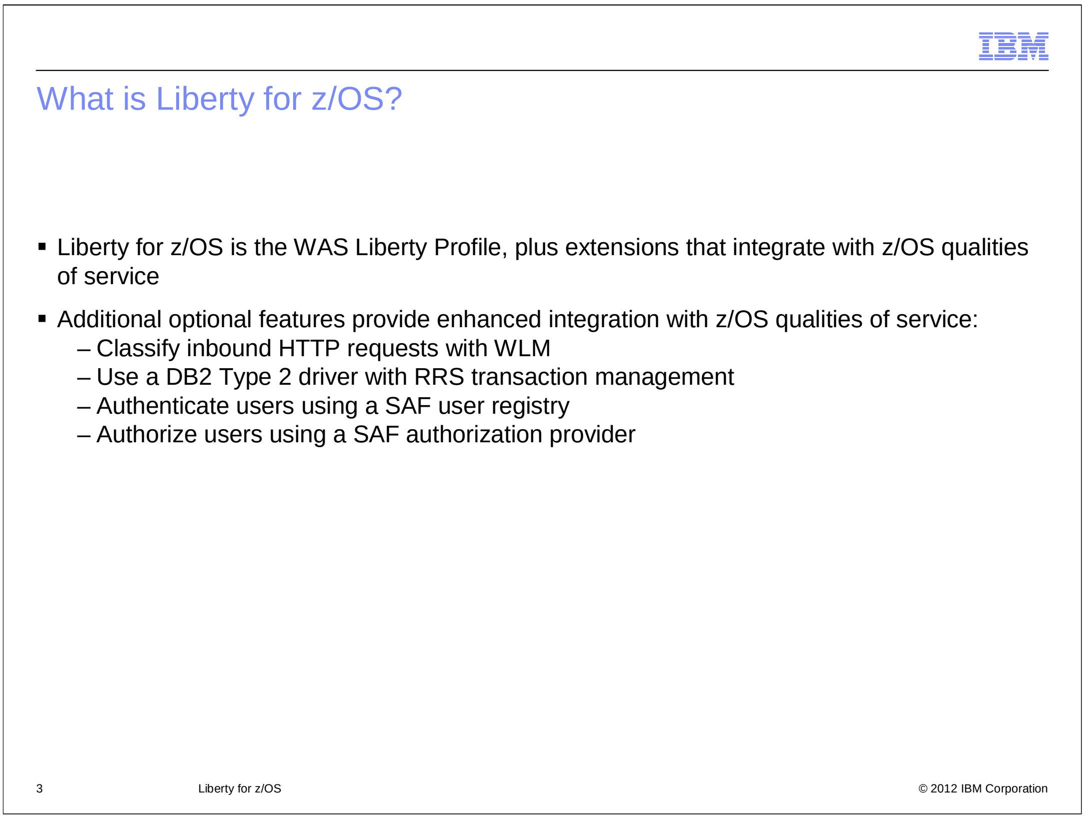
This presentation describes support for Liberty for z/OS included in IBM WebSphere Application Server V8.5



Overview

2 Liberty for z/OS

This section will contain an overview of **Liberty for z/OS**.



What is Liberty for z/OS?

- Liberty for z/OS is the WAS Liberty Profile, plus extensions that integrate with z/OS qualities of service
- Additional optional features provide enhanced integration with z/OS qualities of service:
  - Classify inbound HTTP requests with WLM
  - Use a DB2 Type 2 driver with RRS transaction management
  - Authenticate users using a SAF user registry
  - Authorize users using a SAF authorization provider

3 Liberty for z/OS

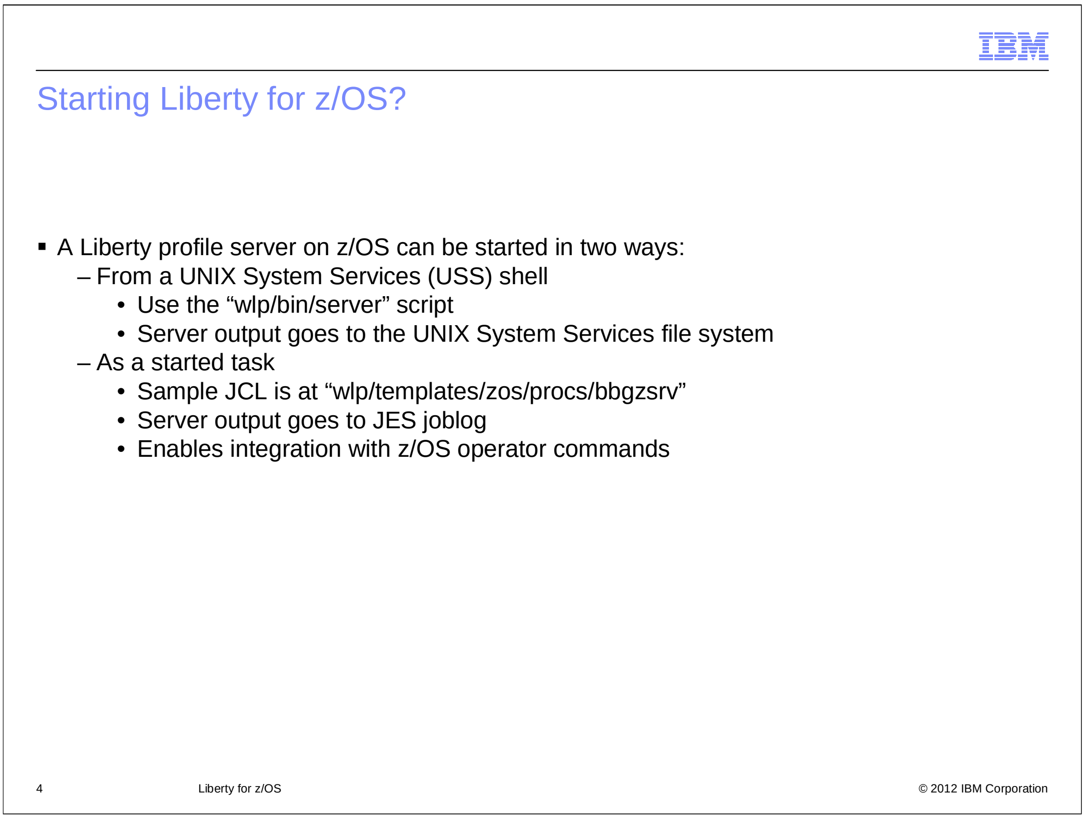
Liberty for z/OS provides everything that Liberty provides on other platforms in addition to optional features that provide integration with the z/OS operating system. These optional features can be individually enabled and disabled on a per server basis. If you want Server A to integrate with WLM but not RRS, and Server B to integrate with SAF but not WLM, you can accomplish that by editing each server's configuration appropriately.

The WAS 8.5 Liberty profile for z/OS supports integration with these qualities of service:

- Inbound HTTP requests can be classified and assigned a transaction class. The transaction class, along with an optional collection name, are used by WLM to achieve response time and throughput goals.

- Data access to local DB2 subsystems can be managed with RRS and can be accessed using the DB2 Type 2 JDBC driver.

- User authentication and authorization can be integrated with your z/OS security product using SAF user registries and SAF authorization providers.

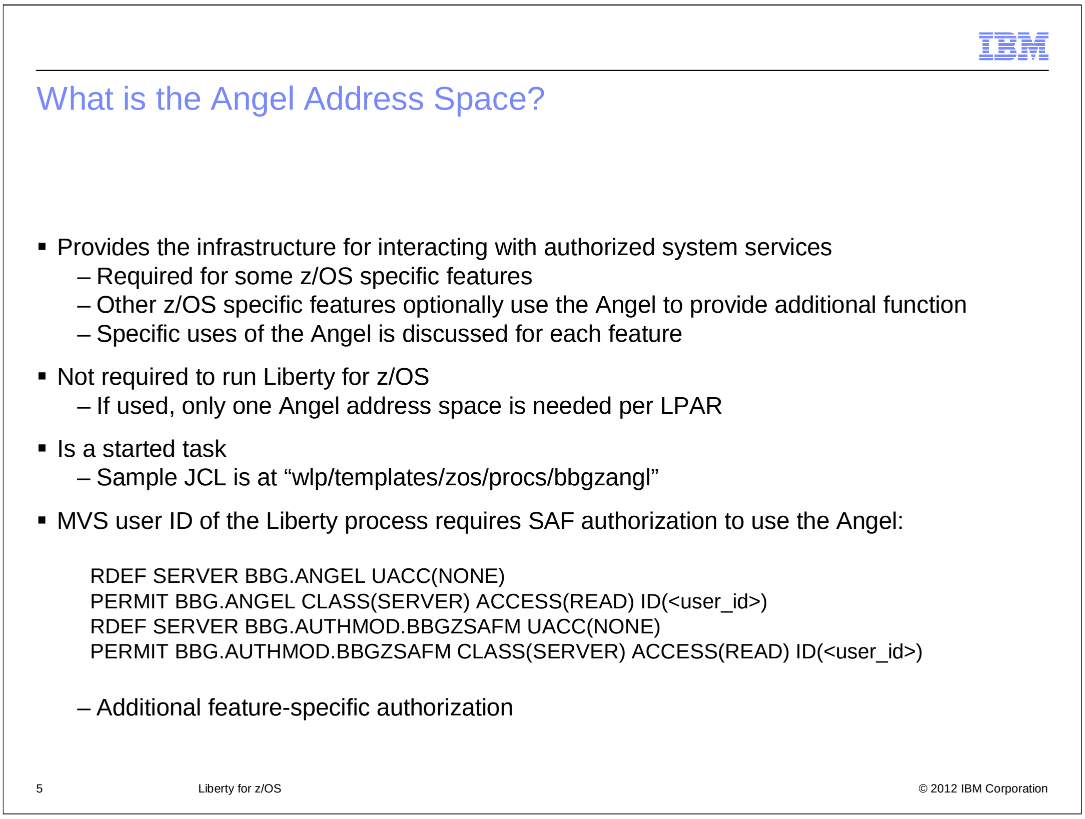


Starting Liberty for z/OS?

- A Liberty profile server on z/OS can be started in two ways:
  - From a UNIX System Services (USS) shell
    - Use the “wlp/bin/server” script
    - Server output goes to the UNIX System Services file system
  - As a started task
    - Sample JCL is at “wlp/templates/zos/procs/bbgzsrv”
    - Server output goes to JES joblog
    - Enables integration with z/OS operator commands

There are two options to start a Liberty for z/OS server. You can use the “server” script in the “bin” directory to start and stop servers the same way they are started and stopped on other platforms. If required, a Liberty server can also be started as a started task using the sample JCL in the “templates” directory as a model.

When a Liberty server is started as a started task, the server output will go to the JES joblog. In addition, z/OS operator commands such as “STOP” can be issued to the server's address space.



What is the Angel Address Space?

- Provides the infrastructure for interacting with authorized system services
  - Required for some z/OS specific features
  - Other z/OS specific features optionally use the Angel to provide additional function
  - Specific uses of the Angel is discussed for each feature
- Not required to run Liberty for z/OS
  - If used, only one Angel address space is needed per LPAR
- Is a started task
  - Sample JCL is at “wlp/templates/zos/procs/bbgzangl”
- MVS user ID of the Liberty process requires SAF authorization to use the Angel:

```
RDEF SERVER BBG.ANGEL UACC(NONE)
PERMIT BBG.ANGEL CLASS(SERVER) ACCESS(READ) ID(<user_id>)
RDEF SERVER BBG.AUTHMOD.BBGZSAFM UACC(NONE)
PERMIT BBG.AUTHMOD.BBGZSAFM CLASS(SERVER) ACCESS(READ) ID(<user_id>)
```

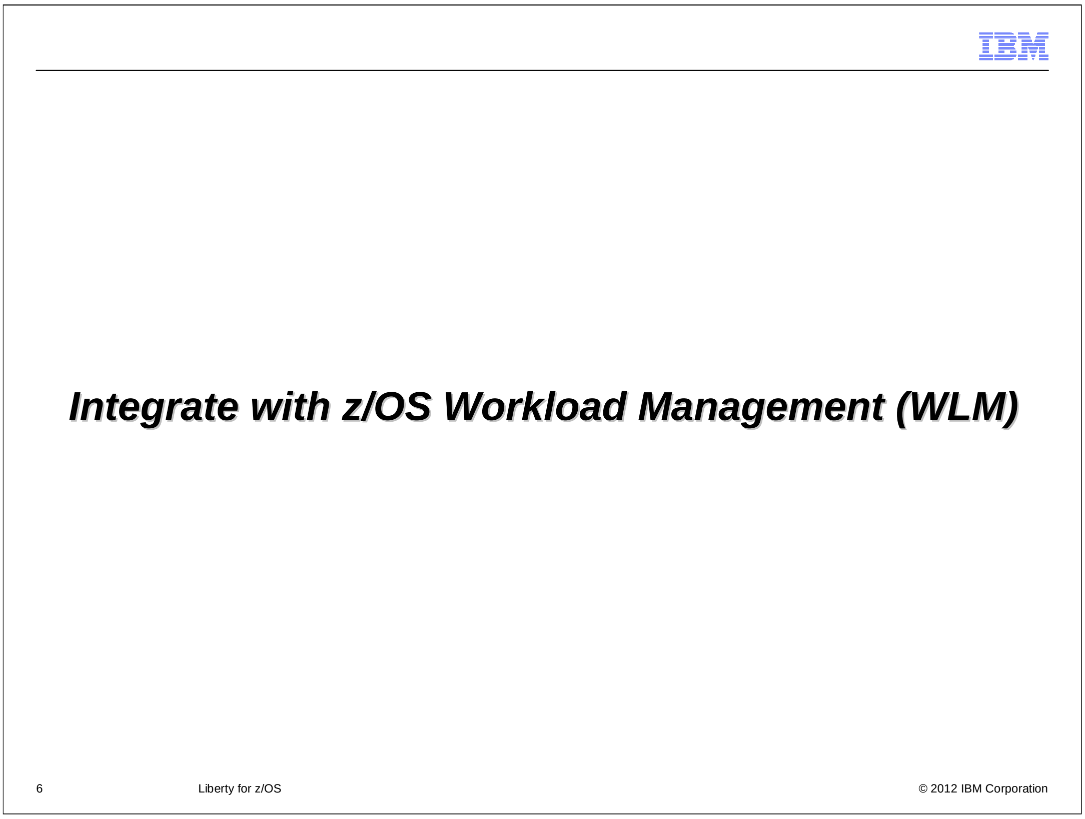
  - Additional feature-specific authorization

5 Liberty for z/OS © 2012 IBM Corporation

The Angel address space provides the infrastructure you need to interact with authorized system services. The Angel address space is not required to use Liberty for z/OS, but it can be required for certain z/OS-specific features. In addition, some z/OS-specific features can operate without the Angel but will provide additional function if the Angel is used.

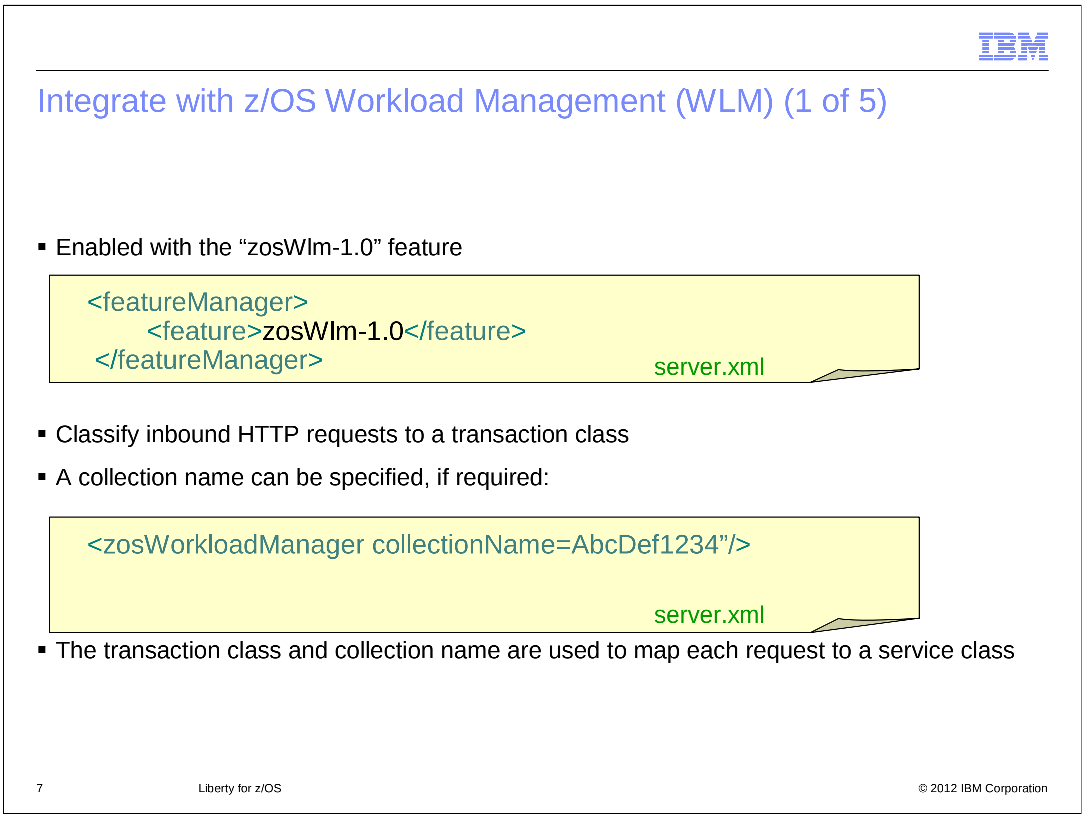
If the Angel is required, sample JCL for starting it can be found in the “templates” directory. Only one Angel address space is needed on a single LPAR, no matter how many Liberty profile servers are running.

Note that starting the Angel is not enough to take full advantage of authorized services. The user ID of Liberty profile servers that want to use authorized services will need SAF authorization for each authorized service that is desired. The specific authorization that is needed is discussed in a separate section for each feature.



Integrate with z/OS Workload Management (WLM)

Liberty for z/OS is used in these scenarios.



Integrate with z/OS Workload Management (WLM) (1 of 5)

- Enabled with the “zosWlm-1.0” feature

```
<featureManager>
        <feature>zosWlm-1.0</feature>
 </featureManager>
                                                      server.xml
```

- Classify inbound HTTP requests to a transaction class
- A collection name can be specified, if required:

```
<zosWorkloadManager collectionName=AbcDef1234”/>

                                                      server.xml
```

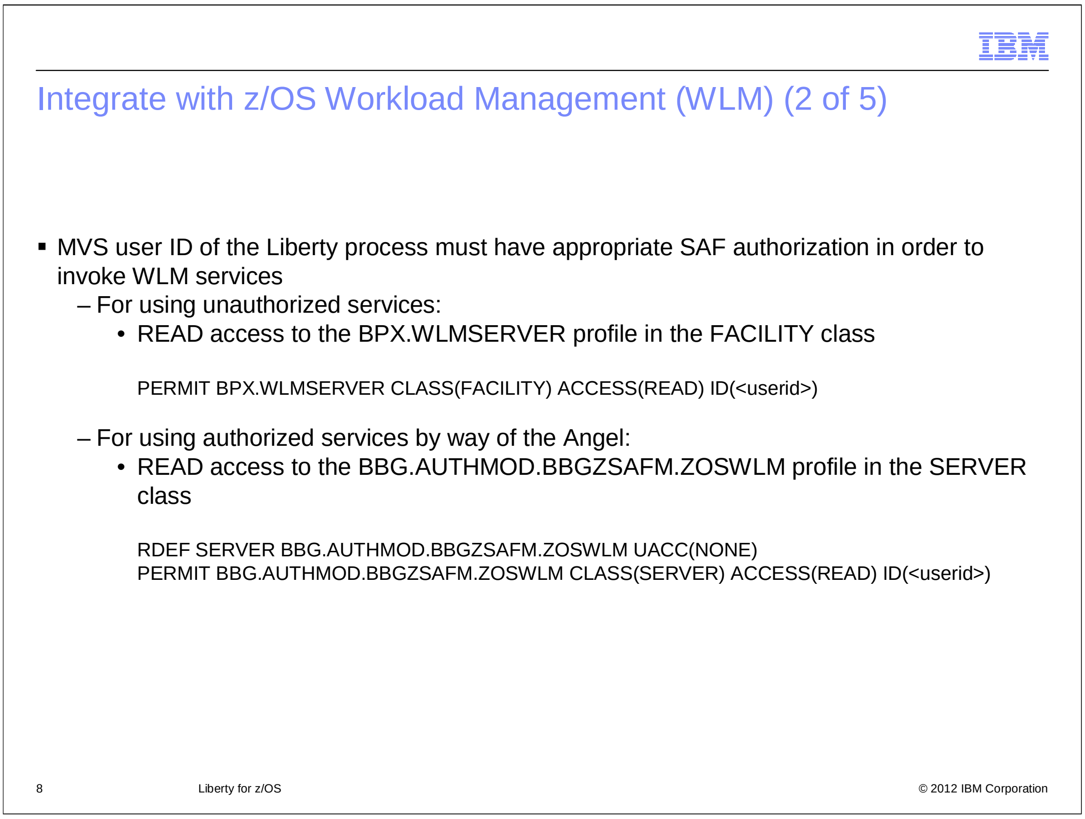
- The transaction class and collection name are used to map each request to a service class

7 Liberty for z/OS © 2012 IBM Corporation

A Liberty profile server can integrate with z/OS Workload Management. To accomplish this, you first need to enable the “zosWlm-1.0” feature in your server.xml. Additional configuration elements can be supplied to classify inbound HTTP requests to a transaction class.

If required, a collectionName can be supplied on the zosWorkloadManager configuration element for use in classifying the request.

The transaction class and collection name are used by WLM to map each request to a service class, that is then used by WLM to manage response time and throughput goals.



Integrate with z/OS Workload Management (WLM) (2 of 5)

- MVS user ID of the Liberty process must have appropriate SAF authorization in order to invoke WLM services
  - For using unauthorized services:
    - READ access to the BPX.WLMSERVER profile in the FACILITY class

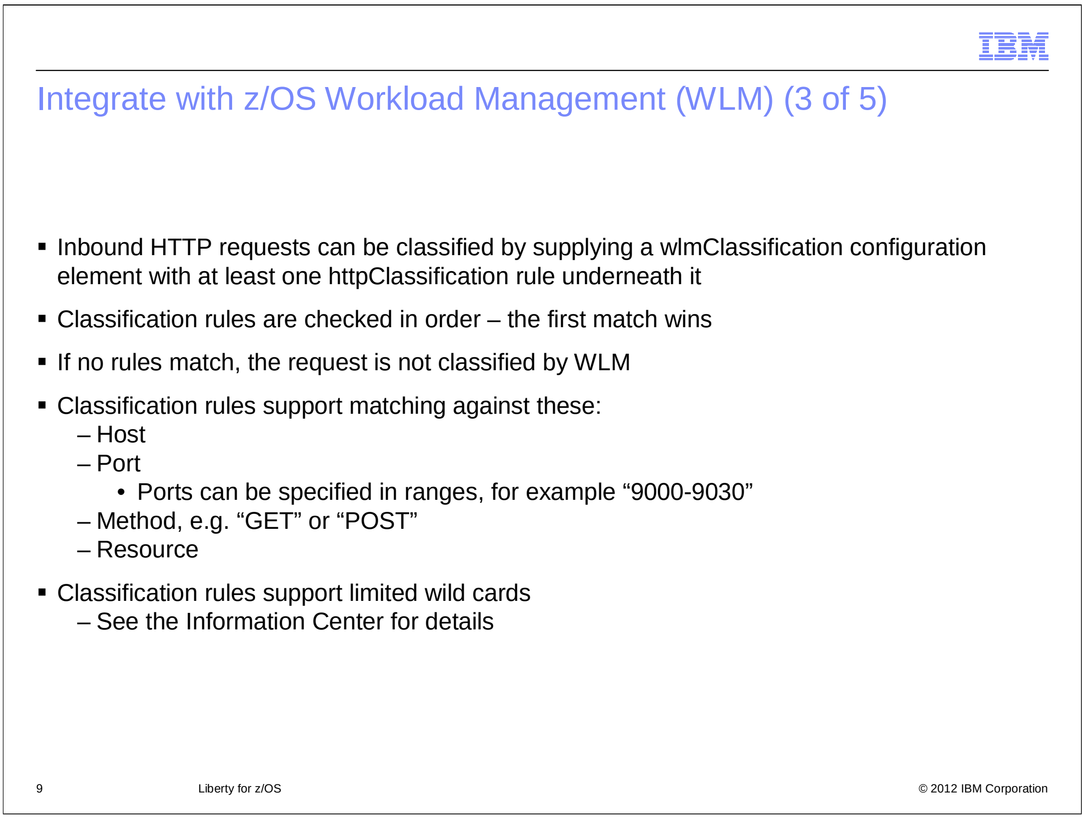
```
PERMIT BPX.WLMSERVER CLASS(FACILITY) ACCESS(READ) ID(<userid>)
```

  - For using authorized services by way of the Angel:
    - READ access to the BBG.AUTHMOD.BBGZSAFM.ZOSWLM profile in the SERVER class

```
RDEF SERVER BBG.AUTHMOD.BBGZSAFM.ZOSWLM UACC(NONE)
PERMIT BBG.AUTHMOD.BBGZSAFM.ZOSWLM CLASS(SERVER) ACCESS(READ) ID(<userid>)
```

Additional SAF authorization is required to integrate with WLM, regardless of whether you are using unauthorized or authorized services.

Sample commands to provide the needed SAF authorization for both the use of unauthorized and authorized services are shown here.



Integrate with z/OS Workload Management (WLM) (3 of 5)

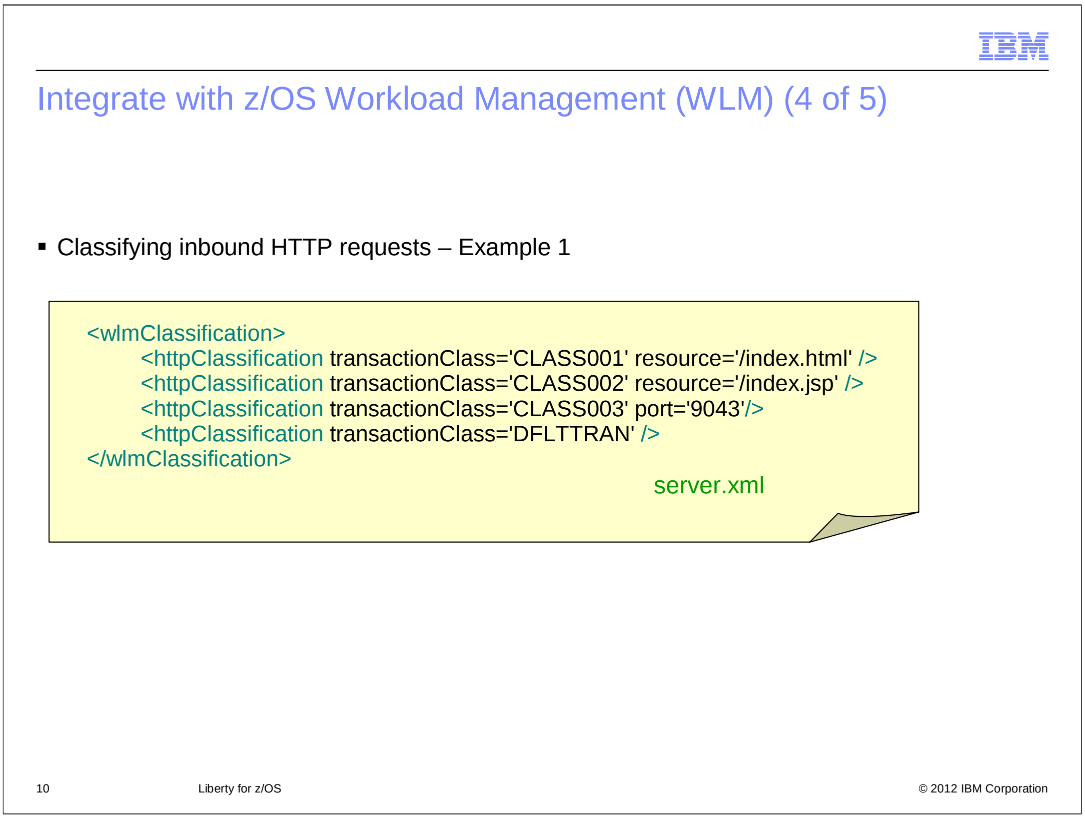
- Inbound HTTP requests can be classified by supplying a wlmClassification configuration element with at least one httpClassification rule underneath it
- Classification rules are checked in order – the first match wins
- If no rules match, the request is not classified by WLM
- Classification rules support matching against these:
  - Host
  - Port
    - Ports can be specified in ranges, for example “9000-9030”
  - Method, e.g. “GET” or “POST”
  - Resource
- Classification rules support limited wild cards
  - See the Information Center for details

In order for work to be classified by WLM, a wlmClassification configuration element with at least one httpClassification rule must be supplied. HTTP classification rules are processed sequentially for each inbound HTTP request. As soon as an HTTP request matches a classification rule, it is assigned that transaction class even if it has matched another rule further below.

An optional “catch-all” classification rule can always be supplied as the last rule. This rule can specify a transaction class but no additional criteria. Any HTTP request that does not match any of the preceding rules will automatically be assigned the transaction class of the catch-all rule.

If an inbound request does not match any classification rules, it is not assigned a transaction class and therefore will not be classified by WLM.

Classification rules can match against a hostname, a port, an HTTP method such as GET or POST, and the requested resource. Ports can be specified in a comma separated list, as a range of ports, or some combination of the two. Most criteria support a limited form of wild cards. For more details, consult the Information Center.



Integrate with z/OS Workload Management (WLM) (4 of 5)

- Classifying inbound HTTP requests – Example 1

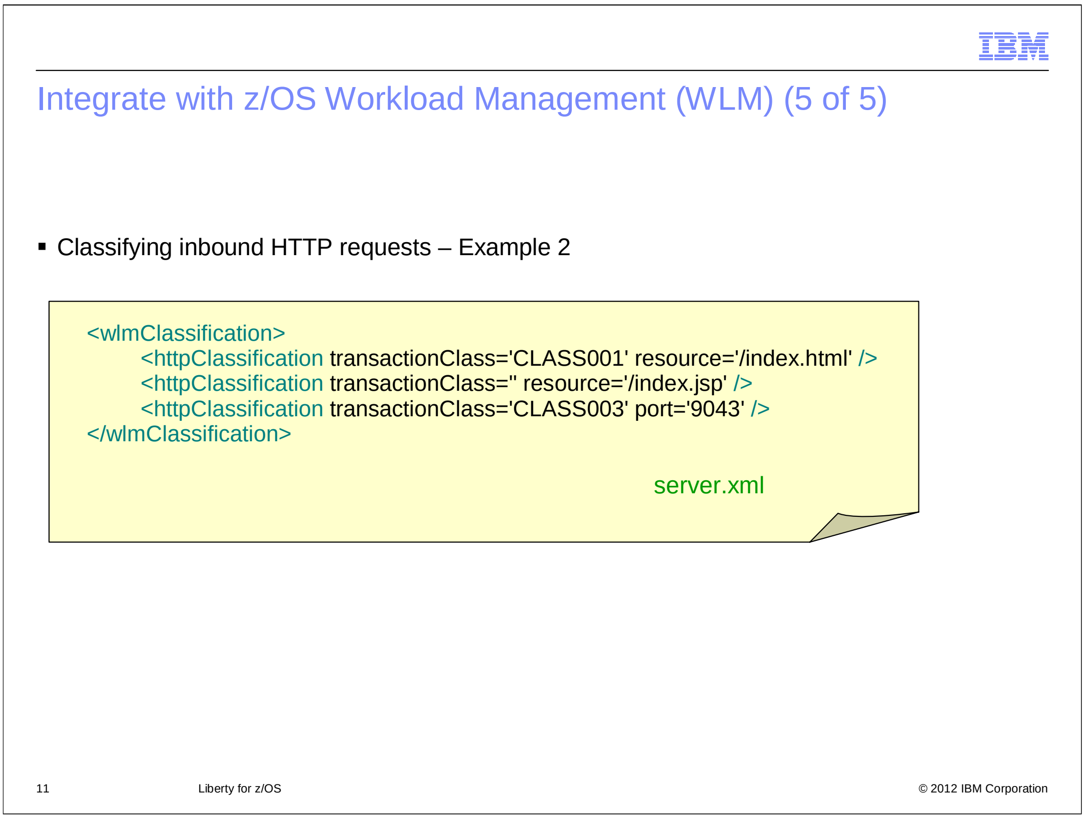
```
<wlmClassification>
        <httpClassification transactionClass='CLASS001' resource='/index.html' />
        <httpClassification transactionClass='CLASS002' resource='/index.jsp' />
        <httpClassification transactionClass='CLASS003' port='9043'/>
        <httpClassification transactionClass='DFLTTRAN' />
</wlmClassification>
```

server.xml

In this example, any request for a resource of “/index.html” is given a transaction class of CLASS001, no matter what the host, port, or method is. Likewise, any request for a resource of “/index.jsp” will be given a transaction class of CLASS002.

Any request for port 9043 that is for a resource other than index.html or index.jsp will be assigned a transaction class of CLASS003.

Finally, any request that does not match any of the previous rules will be assigned a transaction class of DFLTTRAN.



Integrate with z/OS Workload Management (WLM) (5 of 5)

- Classifying inbound HTTP requests – Example 2

```
<wlmClassification>
        <httpClassification transactionClass='CLASS001' resource='/index.html' />
        <httpClassification transactionClass='' resource='/index.jsp' />
        <httpClassification transactionClass='CLASS003' port='9043' />
</wlmClassification>

                                                  server.xml
```

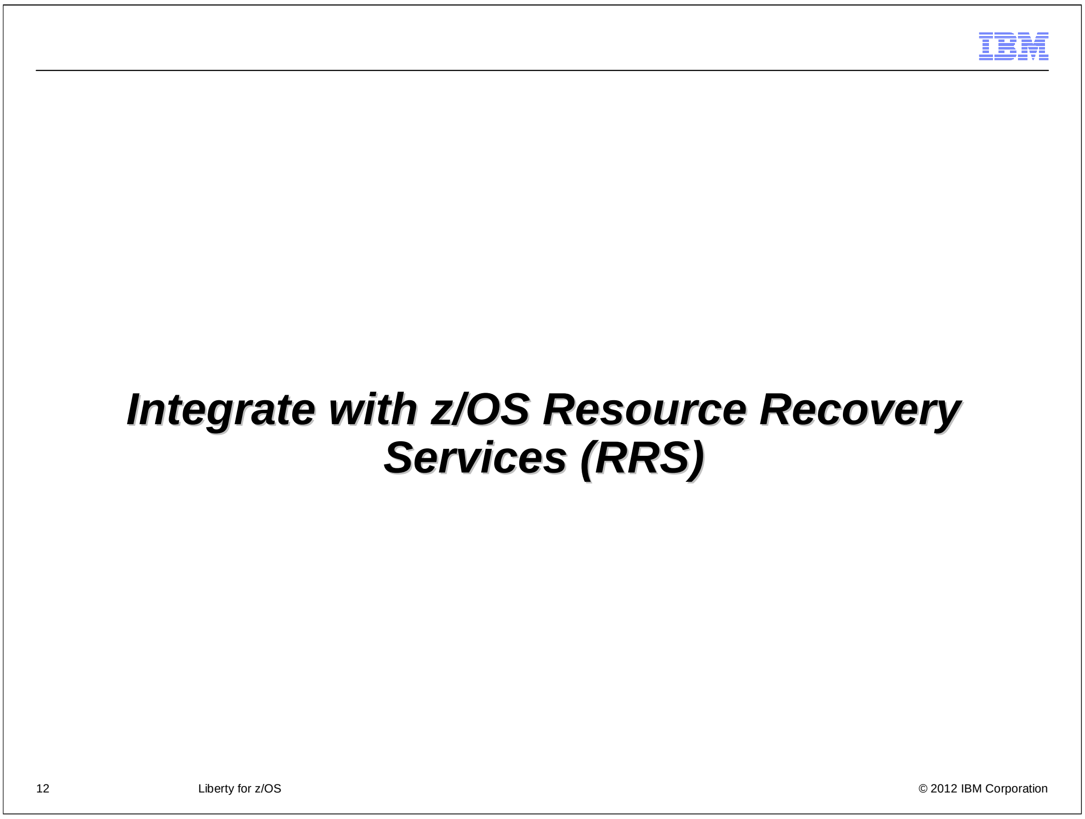
11 Liberty for z/OS © 2012 IBM Corporation

In this second example of classifying an inbound HTTP request, any request for a resource of “/index.html” will still be given a transaction class of CLASS001, no matter what the host, port, or method is.

This time, however, any request for a resource of “/index.jsp” is assigned an empty transaction class which means that these requests are not classified by WLM.

Any request for port 9043 that is for a resource other than index.html or index.jsp will be assigned a transaction class of CLASS003.

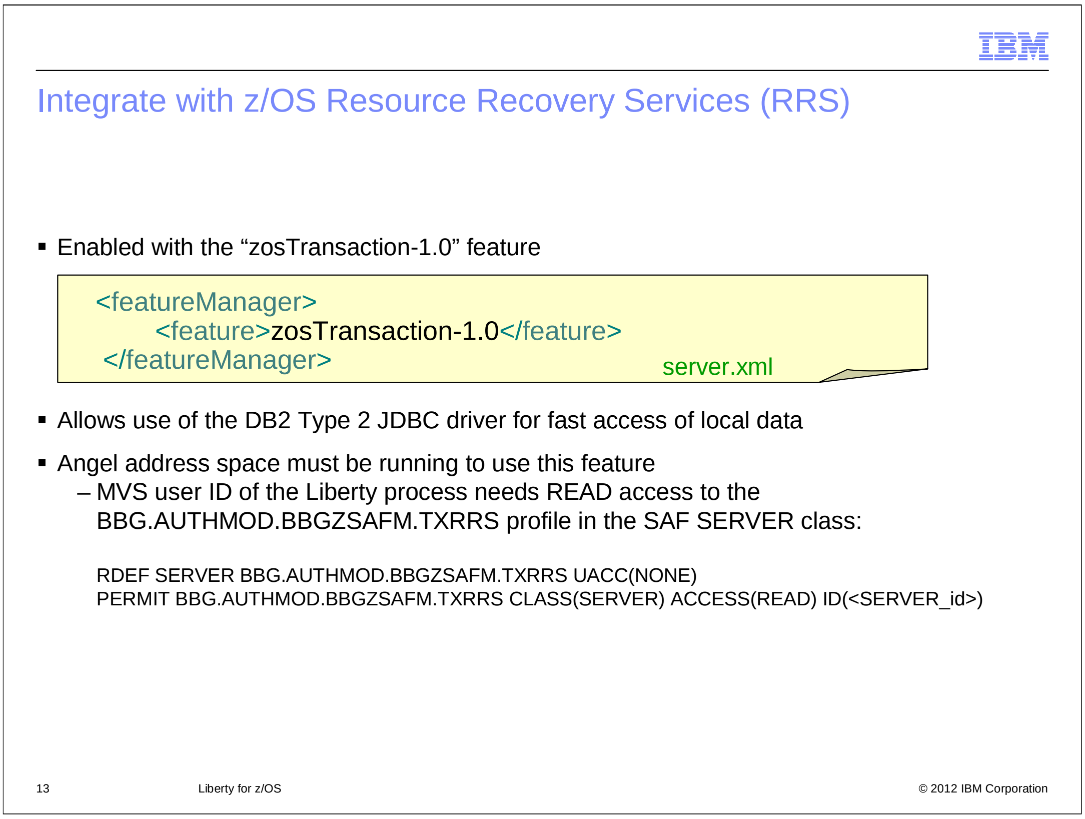
Finally, any request that does not match any of the previous rules will not be assigned a transaction class and will not be classified by WLM.



Integrate with z/OS Resource Recovery Services (RRS)

12 Liberty for z/OS © 2012 IBM Corporation

Liberty for z/OS can be integrated with z/OS Resource Recovery Services (RRS).



Integrate with z/OS Resource Recovery Services (RRS)

- Enabled with the “zosTransaction-1.0” feature

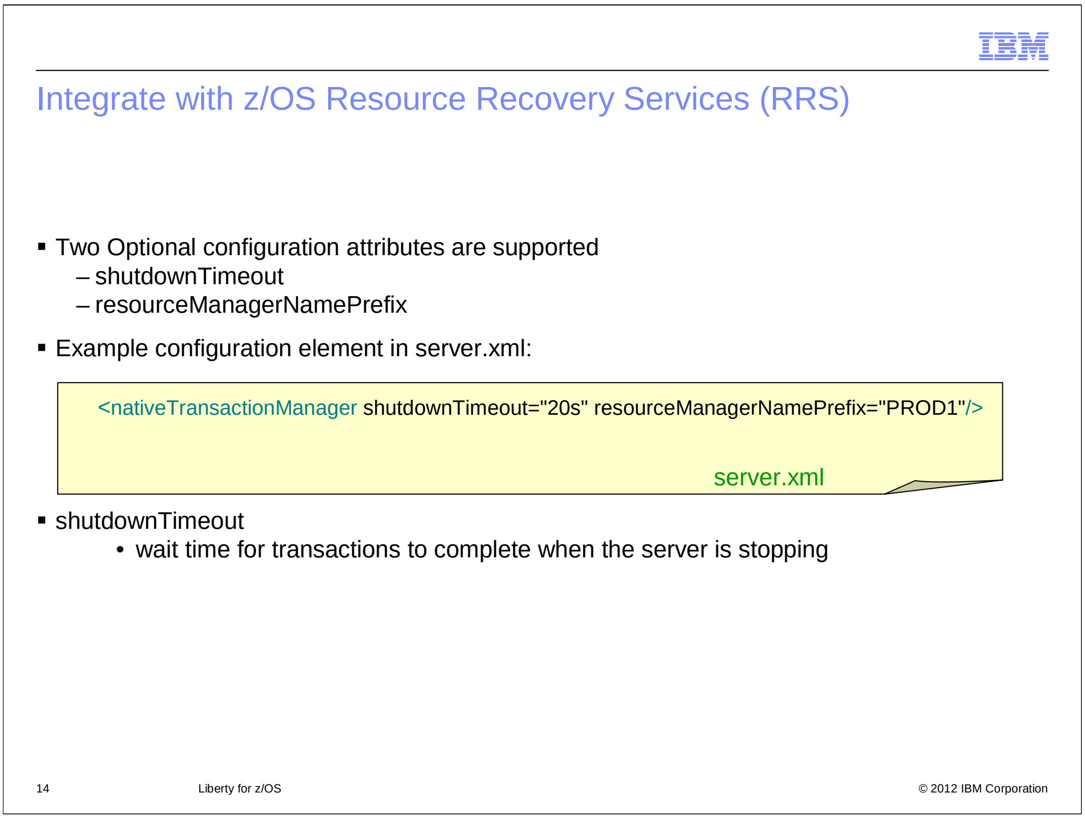
```
<featureManager>
        <feature>zosTransaction-1.0</feature>
 </featureManager>
```

server.xml

- Allows use of the DB2 Type 2 JDBC driver for fast access of local data
- Angel address space must be running to use this feature
  - MVS user ID of the Liberty process needs READ access to the BBG.AUTHMOD.BBGZSAFM.TXRRS profile in the SAF SERVER class:

```
RDEF SERVER BBG.AUTHMOD.BBGZSAFM.TXRRS UACC(NONE)
PERMIT BBG.AUTHMOD.BBGZSAFM.TXRRS CLASS(SERVER) ACCESS(READ) ID(<SERVER_id>)
```

Integrate with z/OS Resource Recovery Services (RRS) in order to enable use of the DB2 Type 2 JDBC driver for accessing co-located data. There are a few prerequisites for enabling this integration. First, you must add the “zosTransaction-1.0” feature to your server.xml. Second, use of the Angel address space is required for integration with RRS. Third, the MVS user ID of the Liberty process must have the appropriate SAF authority to invoke the set of authorized services used for RRS integration. An example of the SAF commands for supplying this authority is seen here.



Integrate with z/OS Resource Recovery Services (RRS)

- Two Optional configuration attributes are supported
  - shutdownTimeout
  - resourceManagerNamePrefix
- Example configuration element in server.xml:

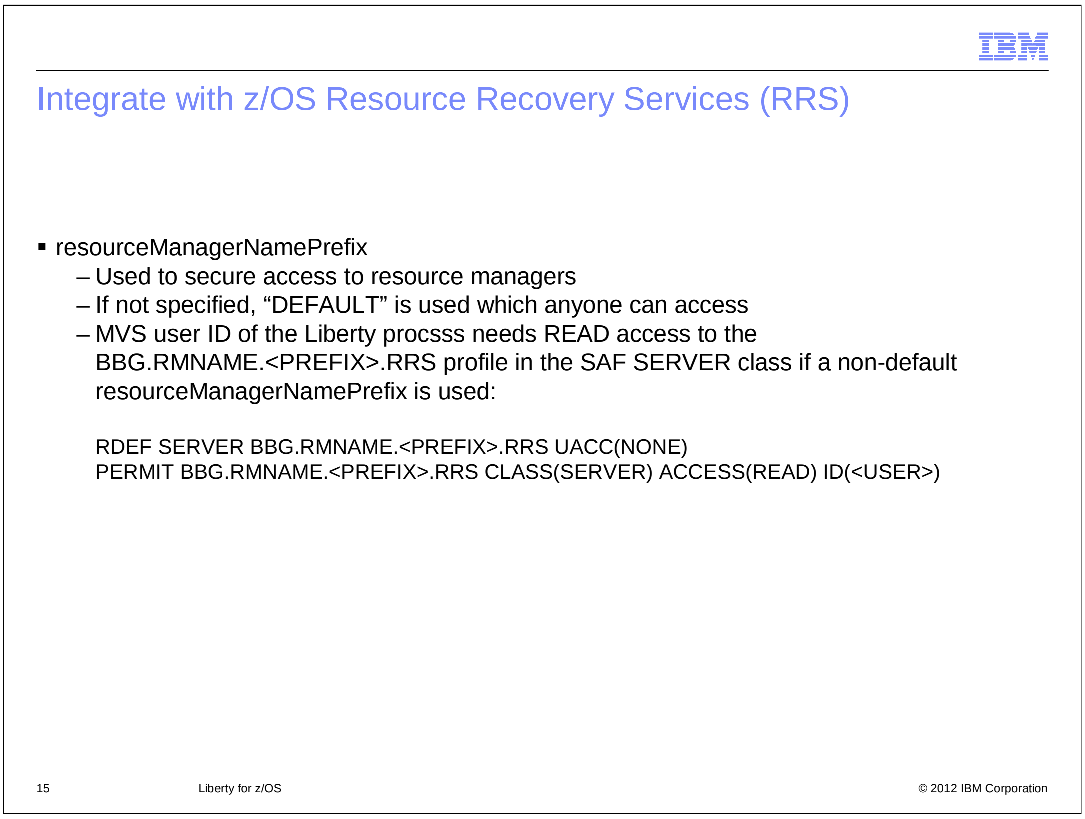
```
<nativeTransactionManager shutdownTimeout="20s" resourceManagerNamePrefix="PROD1"/>
```

server.xml

- shutdownTimeout
  - wait time for transactions to complete when the server is stopping

14 Liberty for z/OS © 2012 IBM Corporation

There are two optional configuration attributes that can be specified to affect the behavior of the z/OS transaction management component. The first is the “shutdownTimeout.” This value is specified in seconds and controls how long the server will wait for outstanding transactions to complete when the server is either stopped or the “zosTransaction-1.0” feature is disabled.



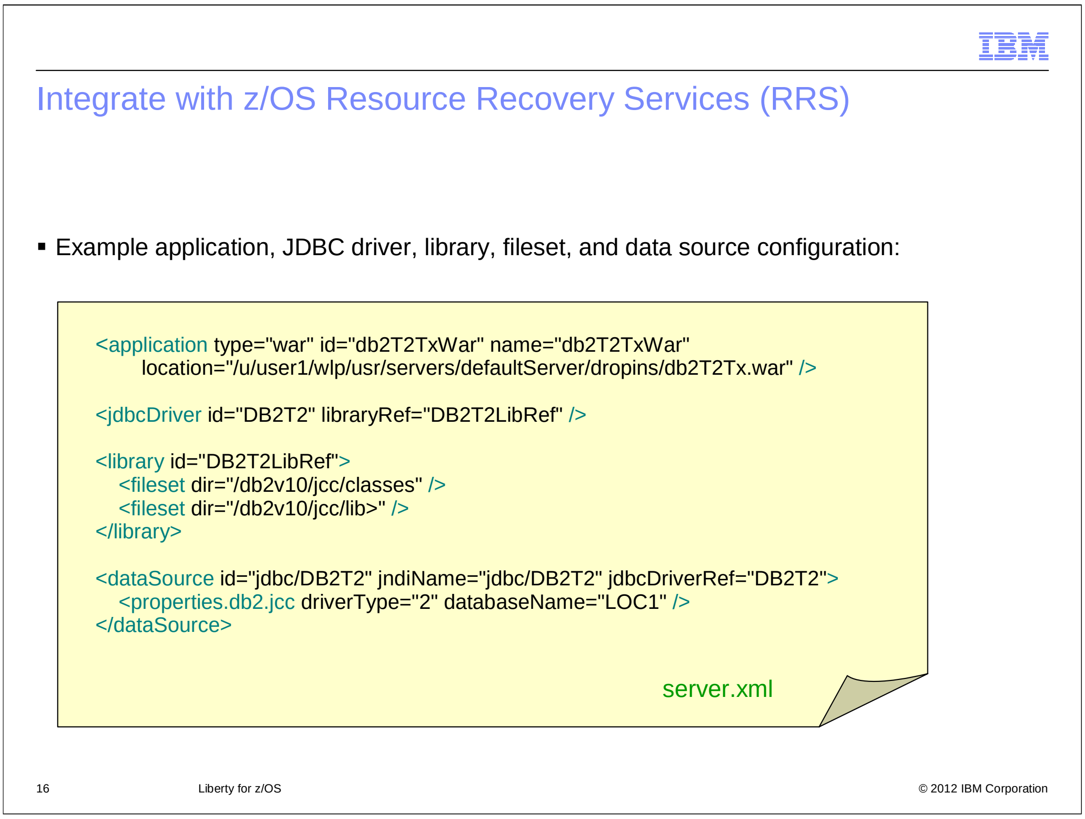
Integrate with z/OS Resource Recovery Services (RRS)

- resourceManagerNamePrefix
  - Used to secure access to resource managers
  - If not specified, “DEFAULT” is used which anyone can access
  - MVS user ID of the Liberty procsss needs READ access to the BBG.RMNAME.<PREFIX>.RRS profile in the SAF SERVER class if a non-default resourceManagerNamePrefix is used:

```
RDEF SERVER BBG.RMNAME.<PREFIX>.RRS UACC(NONE)
PERMIT BBG.RMNAME.<PREFIX>.RRS CLASS(SERVER) ACCESS(READ) ID(<USER>)
```

The second optional configuration attribute is the “resourceManagerNamePrefix.” By default, a prefix of “DEFAULT” is used, which means that anyone is able to access the resource manager. This can suffice for development environments, but for production environments it may be necessary to secure access to the resource manager. By supplying a custom value for the resourceManagerNamePrefix, users will require READ access to the BBG.RMNAME.<PREFIX>.RRS entity in the RACF SERVER class before being able to access the resource manager.

In addition, providing a custom resourceManagerNamePrefix makes it easier for system administrators to determine which resource manager belongs to which server.



Integrate with z/OS Resource Recovery Services (RRS)

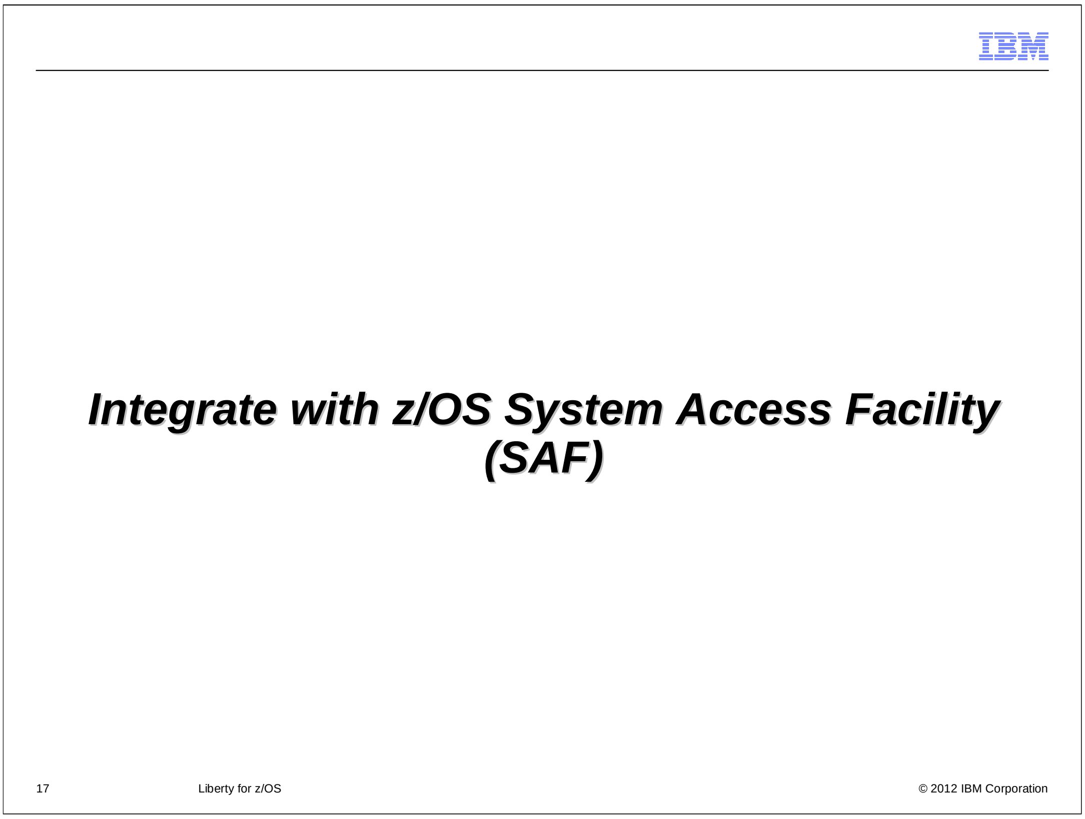
- Example application, JDBC driver, library, fileset, and data source configuration:

```
<application type="war" id="db2T2TxWar" name="db2T2TxWar"
      location="/u/user1/wlp/usr/servers/defaultServer/dropins/db2T2Tx.war" />

<jdbcDriver id="DB2T2" libraryRef="DB2T2LibRef" />

<library id="DB2T2LibRef">
   <fileset dir="/db2v10/jcc/classes" />
   <fileset dir="/db2v10/jcc/lib>" />
</library>

<dataSource id="jdbc/DB2T2" jndiName="jdbc/DB2T2" jdbcDriverRef="DB2T2">
   <properties.db2.jcc driverType="2" databaseName="LOC1" />
</dataSource>
```

server.xml

16 Liberty for z/OS © 2012 IBM Corporation

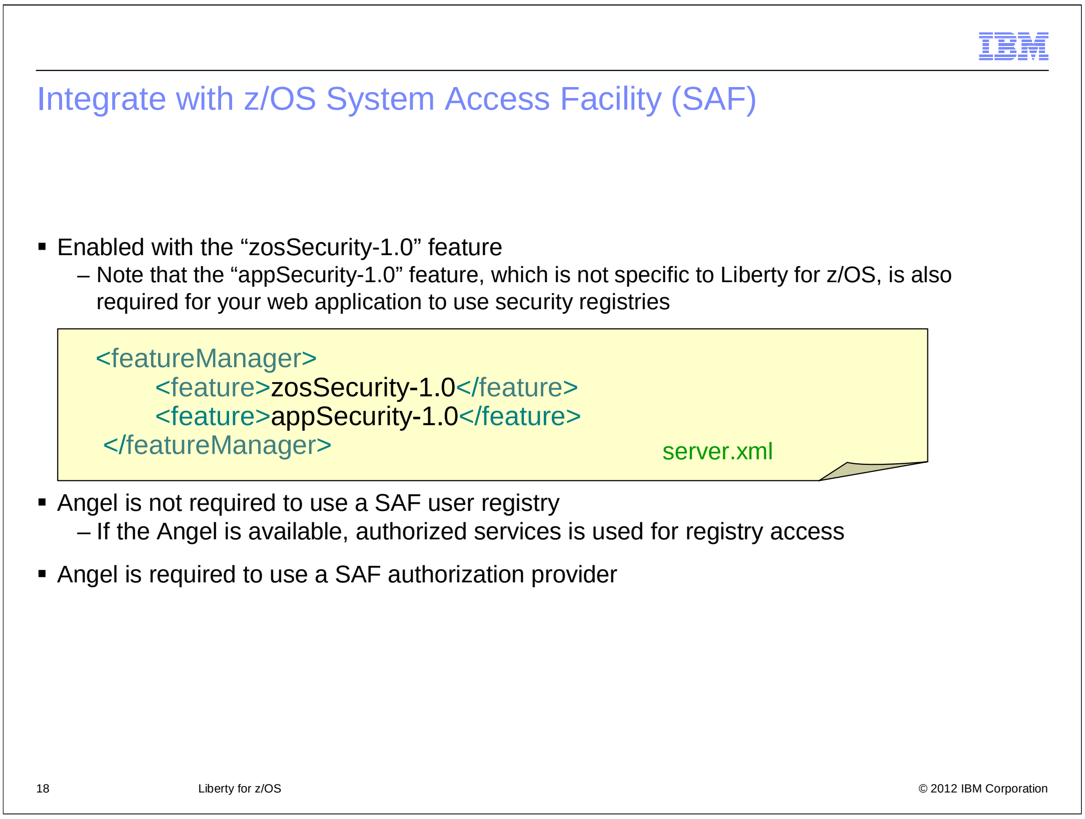
Here is an example of a complete configuration that allows a web application to use the DB2 Type 2 JDBC Driver. The dataSource element at the bottom defines a basic JDBC Data Source which uses a jdbcDriverRef property to point to the correct driver. At the top of the page, a jdbcDriver element is defined to represent an instance of the driver, and it uses a libraryRef property to point to the location of its jar files and native libraries. Finally, a library element is defined that actually points to those directories.



Integrate with z/OS System Access Facility (SAF)

17 Liberty for z/OS © 2012 IBM Corporation

Liberty for z/OS can integrate with the System Access Facility (SAF).



Integrate with z/OS System Access Facility (SAF)

- Enabled with the “zosSecurity-1.0” feature
  - Note that the “appSecurity-1.0” feature, which is not specific to Liberty for z/OS, is also required for your web application to use security registries

```
<featureManager>
        <feature>zosSecurity-1.0</feature>
        <feature>appSecurity-1.0</feature>
 </featureManager>
```

server.xml

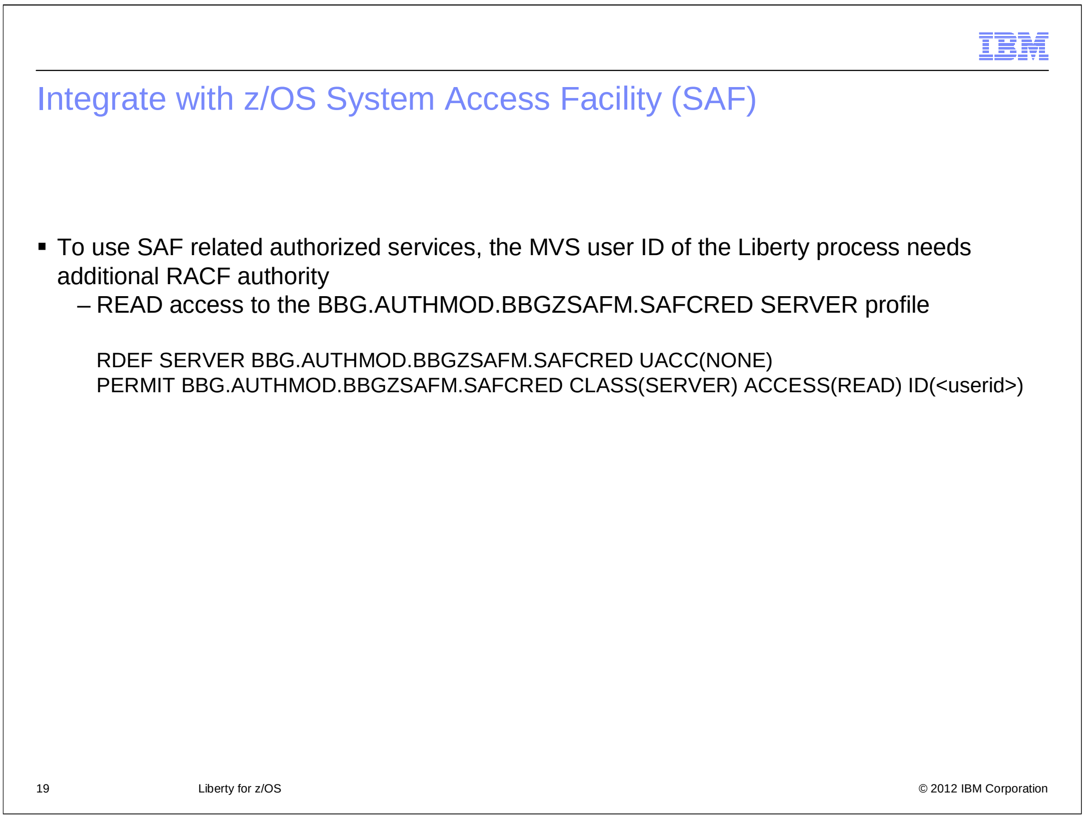
- Angel is not required to use a SAF user registry
  - If the Angel is available, authorized services is used for registry access
- Angel is required to use a SAF authorization provider

18 Liberty for z/OS © 2012 IBM Corporation

Integrate with the System Access Facility (SAF) to provide a SAF user registry and a SAF authorization provider. First, you must add the “zosSecurity-1.0” feature to your server.xml. Note that you will also need to enable the “appSecurity-1.0” feature in order to get any sort of user registry support from your application.

The Angel address space is not required to use a SAF user registry, however if the Angel address space is available and the server has appropriate SAF authorization, the server will take advantage of authorized services.

To use a SAF authorization provider, however, the Angel address space must be started and the server must be authorized to use the appropriate authorized services.



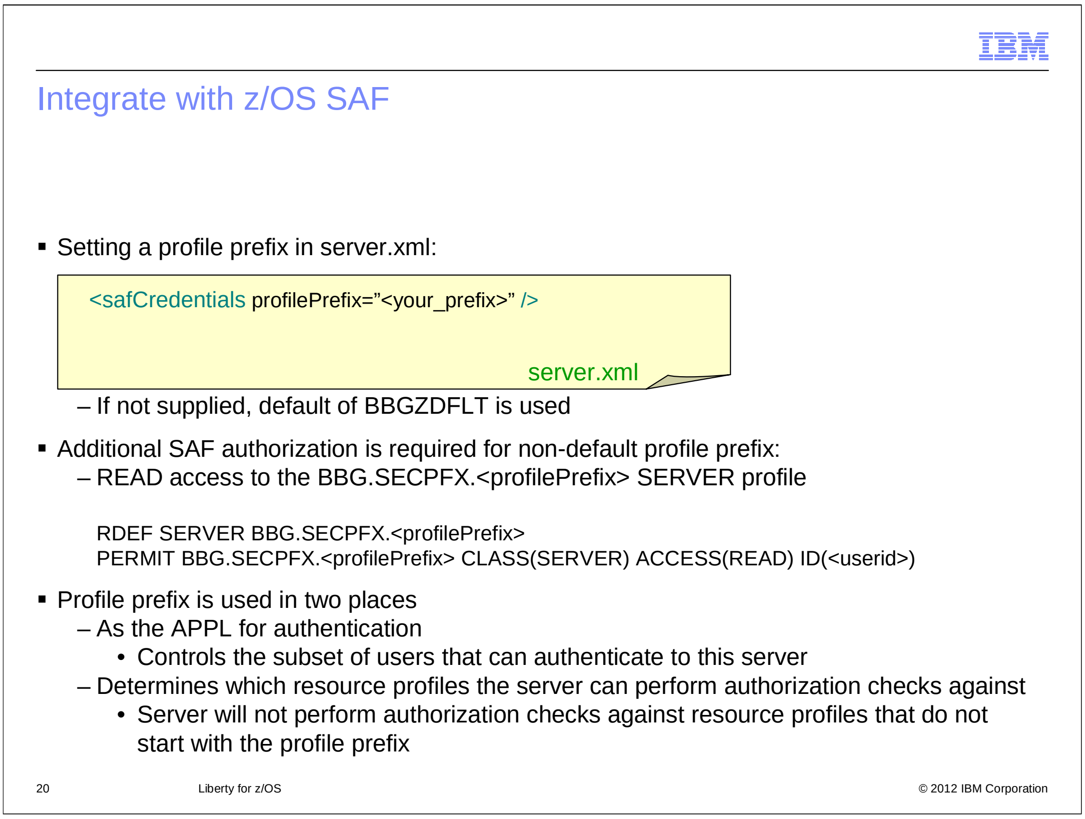
Integrate with z/OS System Access Facility (SAF)

- To use SAF related authorized services, the MVS user ID of the Liberty process needs additional RACF authority
  - READ access to the BBG.AUTHMOD.BBGZSAFM.SAFCRED SERVER profile

```
RDEF SERVER BBG.AUTHMOD.BBGZSAFM.SAFCRED UACC(NONE)
PERMIT BBG.AUTHMOD.BBGZSAFM.SAFCRED CLASS(SERVER) ACCESS(READ) ID(<userid>)
```

19 Liberty for z/OS © 2012 IBM Corporation

If the Angel address space is being used to invoke SAF authorized services, the Liberty for z/OS server user ID needs additional SAF authority. It must have READ access to the BBG.AUTHMOD.BBGZSAFM.SAFCRED resource profile in the SERVER class. Sample SAF commands for providing the necessary access are shown here.



Integrate with z/OS SAF

- Setting a profile prefix in server.xml:

```
<safCredentials profilePrefix=”<your_prefix>” />
```

server.xml

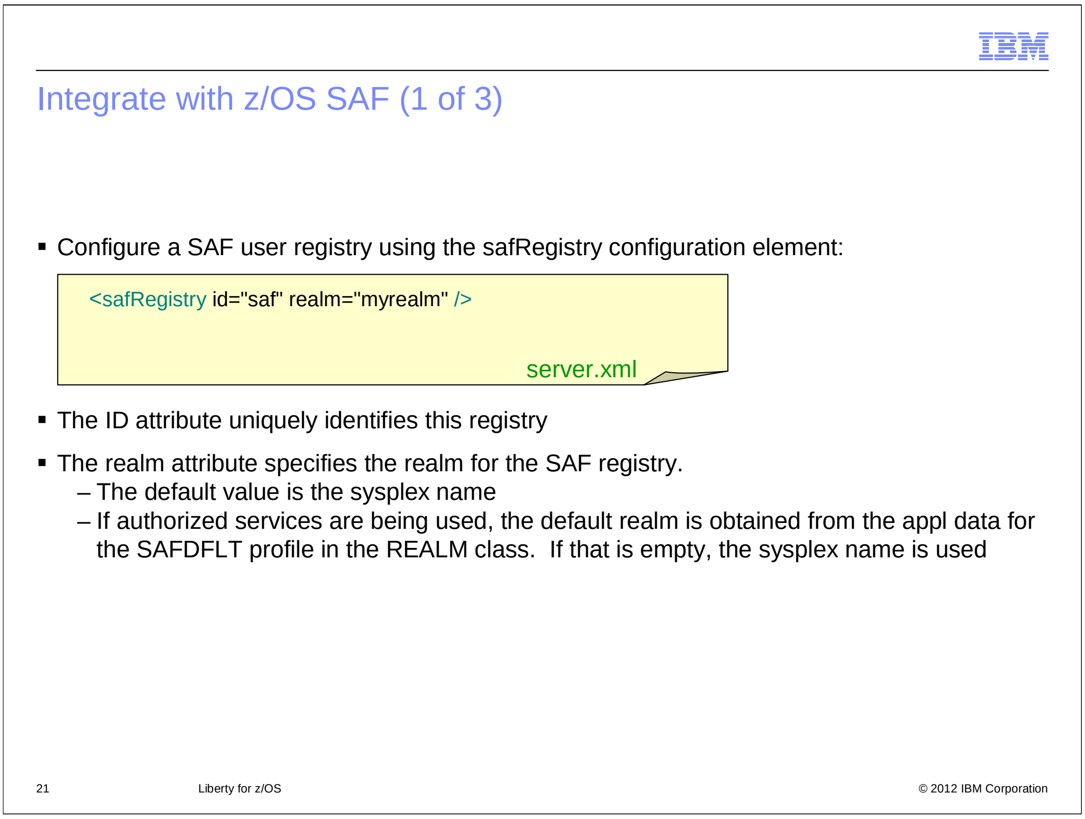
  - If not supplied, default of BBGZDFLT is used
- Additional SAF authorization is required for non-default profile prefix:
  - READ access to the BBG.SECPFX.<profilePrefix> SERVER profile

```
RDEF SERVER BBG.SECPFX.<profilePrefix>
PERMIT BBG.SECPFX.<profilePrefix> CLASS(SERVER) ACCESS(READ) ID(<userid>)
```

- Profile prefix is used in two places
  - As the APPL for authentication
    - Controls the subset of users that can authenticate to this server
  - Determines which resource profiles the server can perform authorization checks against
    - Server will not perform authorization checks against resource profiles that do not start with the profile prefix

The Liberty process must also have READ access to the BBG.SECPFX.<profilePrefix> resource profile in the SERVER class. The profile prefix can be defined in the safCredential configuration element, otherwise the default of BBGZDFLT is used. Sample commands for providing the necessary SAF authorization are shown here.

The profile prefix is used as the APPL for authentication, and it also limits the profiles that the server can perform authorization checks against. The server will not perform authorization checks against any resource profiles that do not begin with the profile prefix. More information is provided on a later slide.



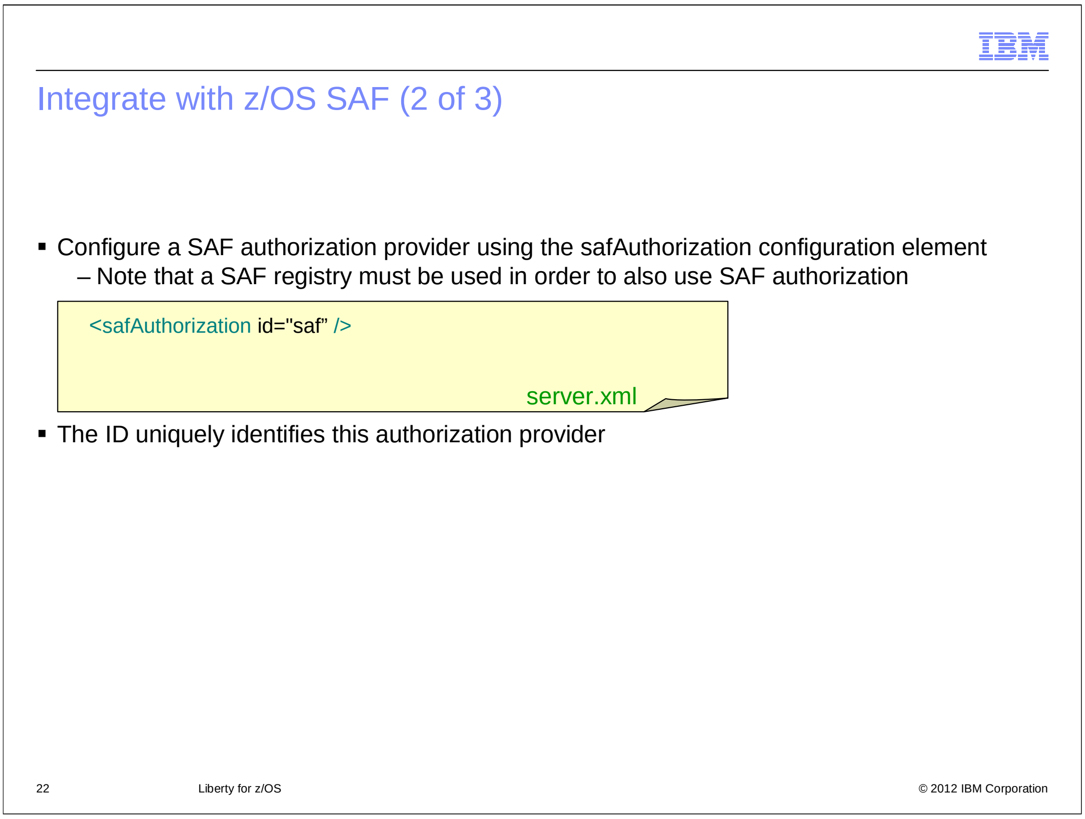
Integrate with z/OS SAF (1 of 3)

- Configure a SAF user registry using the safRegistry configuration element:

```
<safRegistry id="saf" realm="myrealm" />

                                        server.xml
```

- The ID attribute uniquely identifies this registry
- The realm attribute specifies the realm for the SAF registry.
  - The default value is the sysplex name
  - If authorized services are being used, the default realm is obtained from the appl data for the SAFDFLT profile in the REALM class. If that is empty, the sysplex name is used

A SAF user registry is configured by supplying a safRegistry configuration element in server.xml. There are two supported attributes, “id” and “realm.” The ID attribute is just a unique identifier for your registry. The realm is the realm associated with the SAF registry. The default value for the realm is the sysplex name, however if authorized services are being used, the appl data of the SAFDFLT profile in the realm class is checked first. If that is empty, the sysplex name will be used.



Integrate with z/OS SAF (2 of 3)

- Configure a SAF authorization provider using the safAuthorization configuration element
  - Note that a SAF registry must be used in order to also use SAF authorization

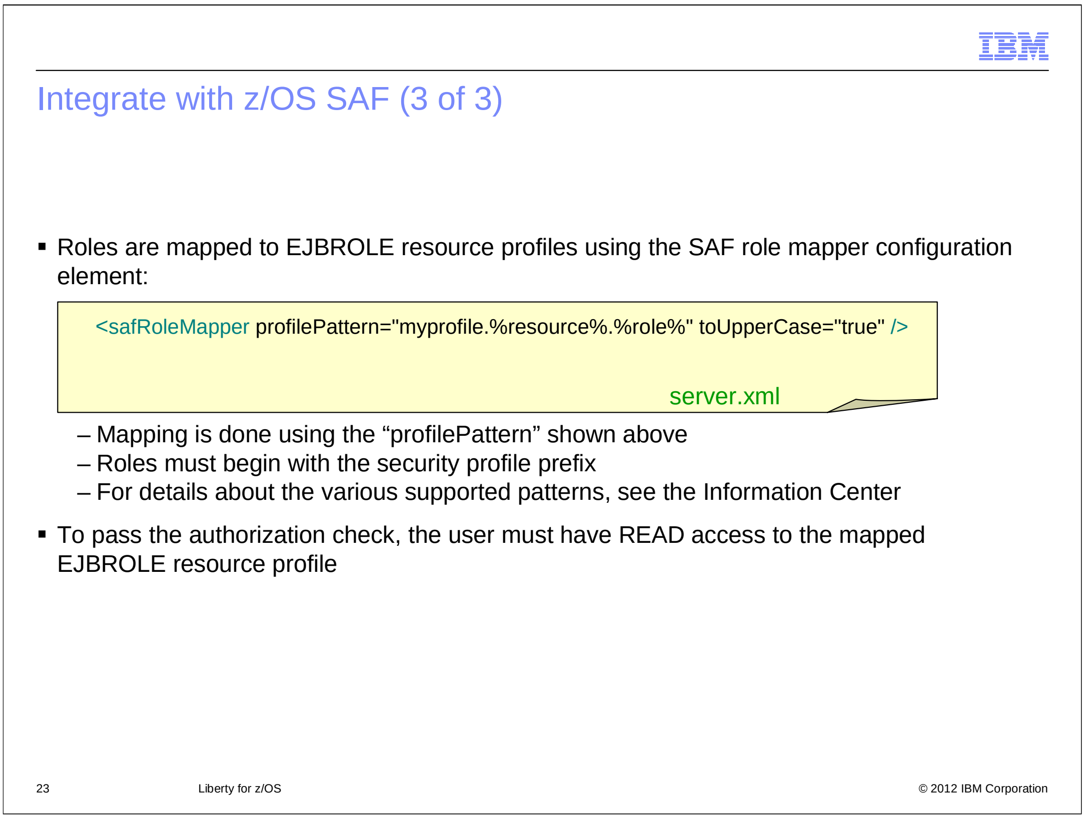
```
<safAuthorization id="saf" />
```

server.xml

- The ID uniquely identifies this authorization provider

22 Liberty for z/OS © 2012 IBM Corporation

A SAF authorization provider is configured by supplying a safAuthorization configuration element in server.xml. Note that in order to use SAF authorization, you must already be using a SAF registry as discussed on the previous slide.



Integrate with z/OS SAF (3 of 3)

- Roles are mapped to EJBROLE resource profiles using the SAF role mapper configuration element:

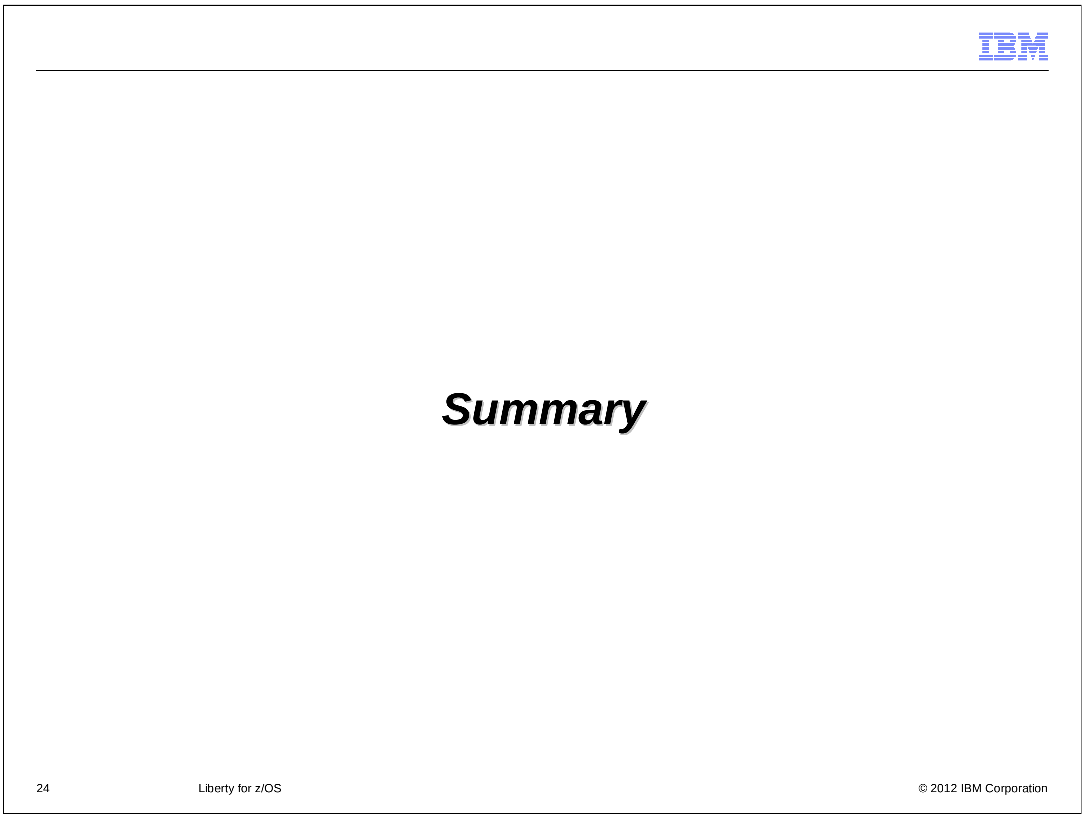
```
<safRoleMapper profilePattern="myprofile.%resource%.%role%" toUpperCase="true" />
```

server.xml

  - Mapping is done using the “profilePattern” shown above
  - Roles must begin with the security profile prefix
  - For details about the various supported patterns, see the Information Center
- To pass the authorization check, the user must have READ access to the mapped EJBROLE resource profile

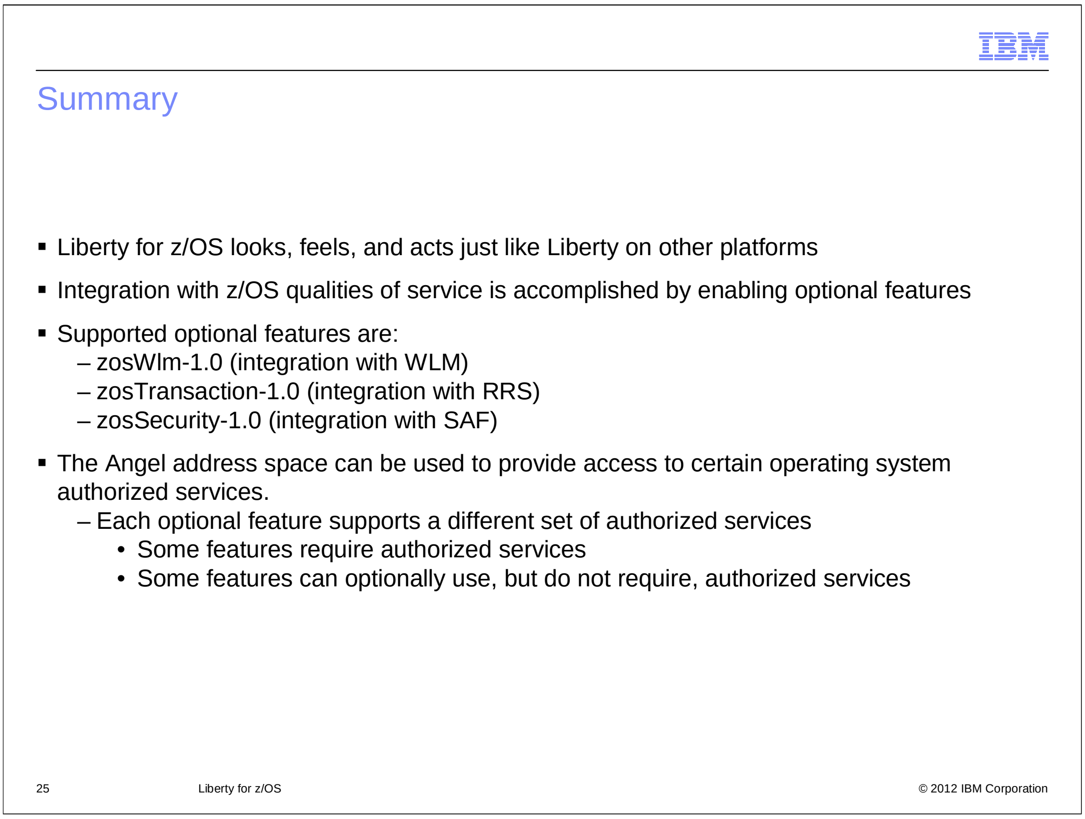
When an authorization check is performed, the resource and role are mapped to an EJBROLE resource profile. The way the mapping is done is determined by the profilePattern attribute of the safRoleMapper configuration element. The profilePattern maps various pieces of information such as the profile prefix, requesting resource, and role into an EJBROLE resource profile. The authorization check will pass if the user being authorized has READ access to the mapped EJBROLE. Details for configuring the role mapping can be found in the information center.

Note that the Liberty server will validate that the mapped resource profile for an authorization check begins with the profile prefix defined in the safCredentials configuration element. This limits the resources that the Liberty server can check authorization to and prevents abuse of Liberty authorization checks by users.



Summary

This section contains a summery of this presentation.



Summary

- Liberty for z/OS looks, feels, and acts just like Liberty on other platforms
- Integration with z/OS qualities of service is accomplished by enabling optional features
- Supported optional features are:
  - zosWlm-1.0 (integration with WLM)
  - zosTransaction-1.0 (integration with RRS)
  - zosSecurity-1.0 (integration with SAF)
- The Angel address space can be used to provide access to certain operating system authorized services.
  - Each optional feature supports a different set of authorized services
    - Some features require authorized services
    - Some features can optionally use, but do not require, authorized services

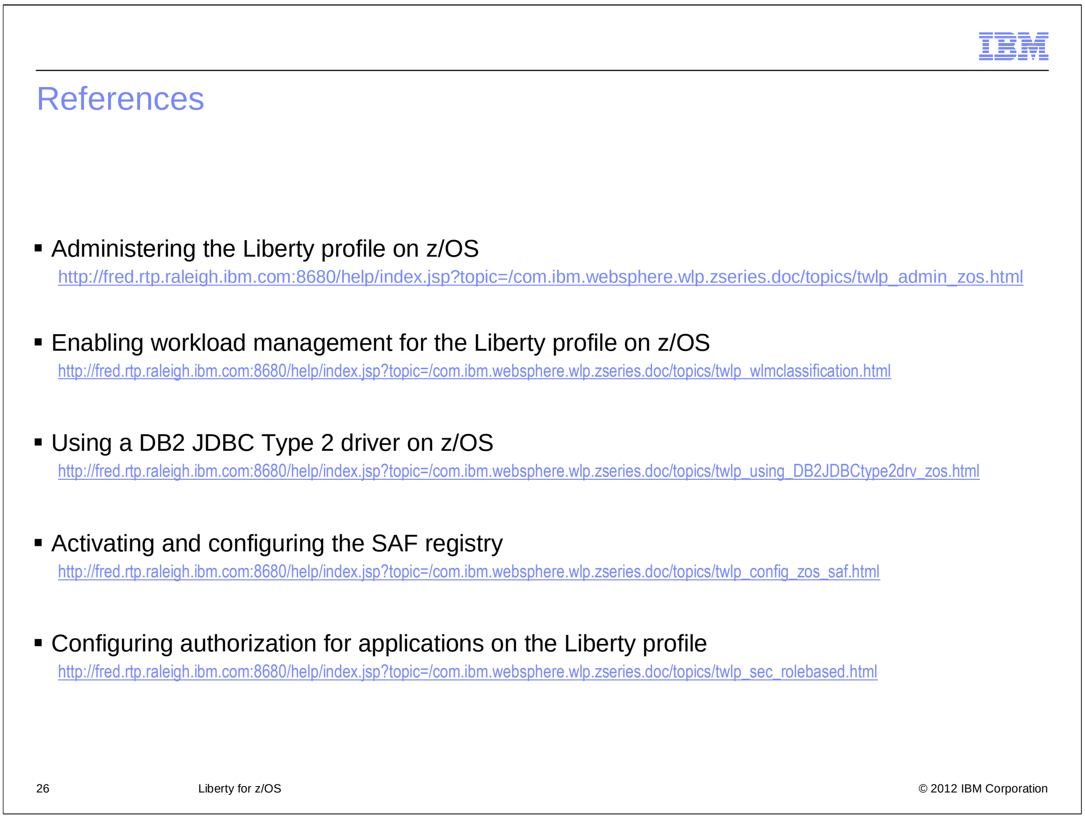
Out of the box, Liberty for z/OS is just like Liberty on any other platform. However, optional features are provided that allow access to various z/OS qualities of service.

The zosWlm-1.0 feature can be enabled to allow inbound HTTP requests to be classified and run under a WLM enclave.

The zosTransaction-1.0 feature can be enabled to allow transaction management with RRS and the use of the DB2 Type 2 JDBC driver.

The zosSecurity-1.0 feature can be enabled to allow the use of SAF user registries and SAF authorization providers.

An Angel address space can be optionally used to provide access to certain operating system authorized services. Some of the z/OS optional features require access to authorized services, whereas others can optionally use them but do not require them.

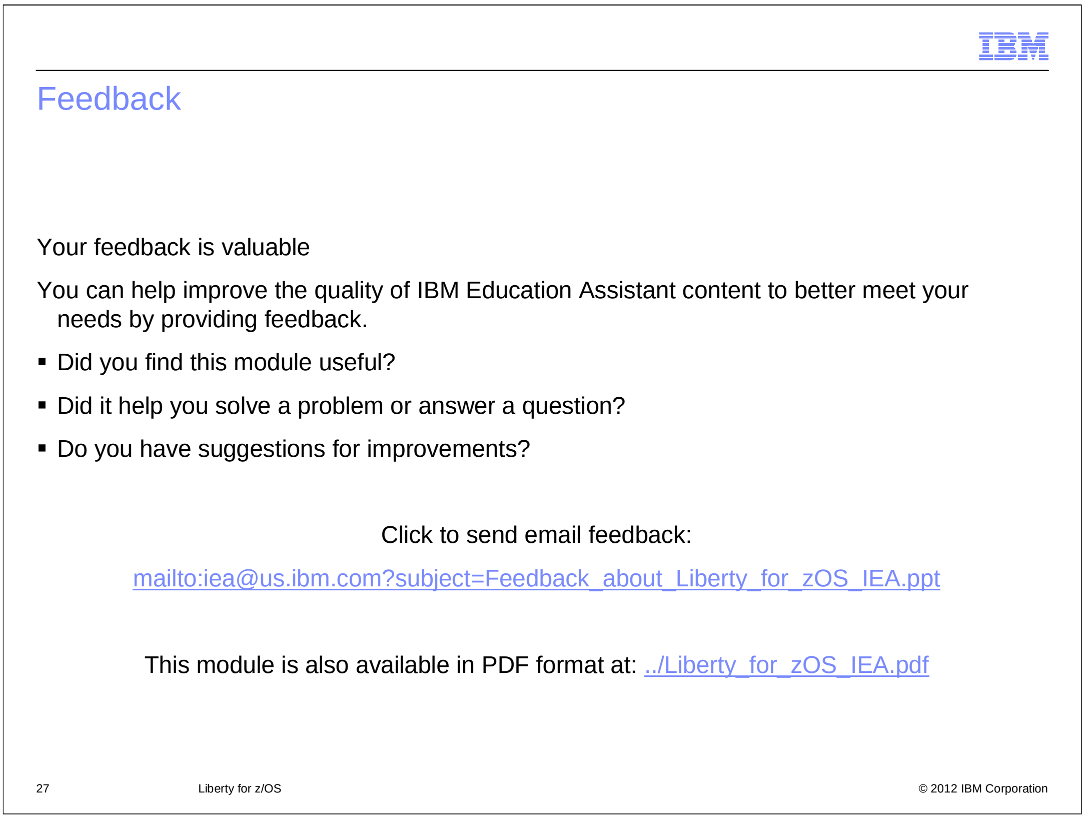


References

- Administering the Liberty profile on z/OS
  http://fred.rtp.raleigh.ibm.com:8680/help/index.jsp?topic=/com.ibm.websphere.wlp.zseries.doc/topics/twlp_admin_zos.html
- Enabling workload management for the Liberty profile on z/OS
  http://fred.rtp.raleigh.ibm.com:8680/help/index.jsp?topic=/com.ibm.websphere.wlp.zseries.doc/topics/twlp_wlmclassification.html
- Using a DB2 JDBC Type 2 driver on z/OS
  http://fred.rtp.raleigh.ibm.com:8680/help/index.jsp?topic=/com.ibm.websphere.wlp.zseries.doc/topics/twlp_using_DB2JDBCtype2drv_zos.html
- Activating and configuring the SAF registry
  http://fred.rtp.raleigh.ibm.com:8680/help/index.jsp?topic=/com.ibm.websphere.wlp.zseries.doc/topics/twlp_config_zos_saf.html
- Configuring authorization for applications on the Liberty profile
  http://fred.rtp.raleigh.ibm.com:8680/help/index.jsp?topic=/com.ibm.websphere.wlp.zseries.doc/topics/twlp_sec_rolebased.html

26 Liberty for z/OS © 2012 IBM Corporation

See these references for additional information about **Liberty for z/OS.**



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

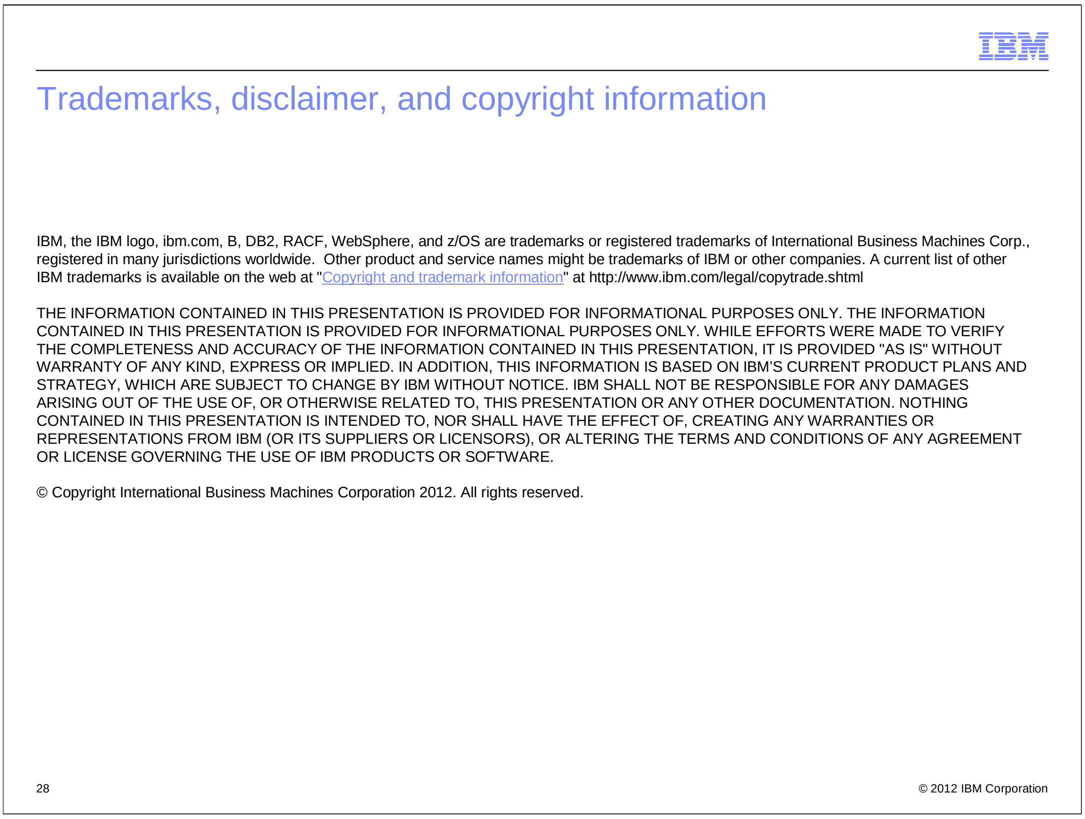
Click to send email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_Liberty_for_zOS_IEA.ppt

This module is also available in PDF format at: ../Liberty_for_zOS_IEA.pdf

27 Liberty for z/OS © 2012 IBM Corporation

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, B, DB2, RACF, WebSphere, and z/OS are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.