

WebSphere Application Server for z/OS and OS/390 WorkShop Using J2EE Technology WAS/390 J2EE Connector for CICS

IBM
Washington Systems Center

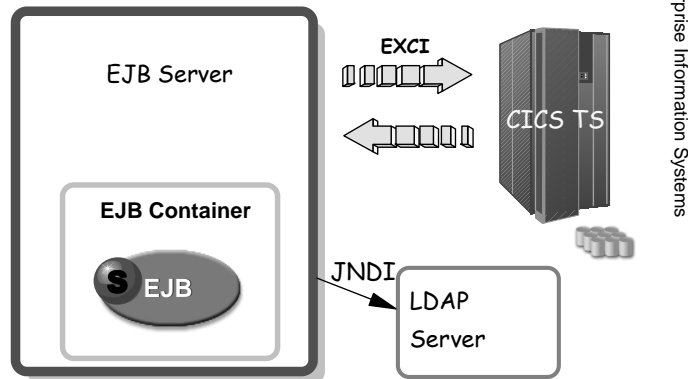


Outline



- **WAS/390 CICSEHCI Connector Overview**
- **WAS/390 CICSEHCI Requirements**
- **Installing the CICS Transaction Gateway**
- **Configuring the CICS Transaction Server for CICS EJBs**
- **Application Assembly Tool for J2EE EJB Application**
- **Introduction to J2EE Connector Session Bean Lab**
- **References**

Overview of WAS/390 CICSEXCI



Implemented based on the Sun J2EE Connector Architecture

- 1 The business-critical applications running on a CICS Server can be accessed from an EJB application running under a WAS/390 application server which uses the EXCI to communicate with a CICS TS.
2. Non-conversational CICS transactions are supported
3. Supports 2-phase commit of global transactions using RRS.

Transaction Management



- Flat transaction model (not nested)
- May be delimited by the enterprise bean
 - ▶ begin, commit, rollback
 - ▶ Stateful Session Beans can start a transaction in one method and commit it in another method
 - ▶ Transactions started in a Stateless Session Bean method do not extend past the end of that method

CICS Connector Transaction Processing

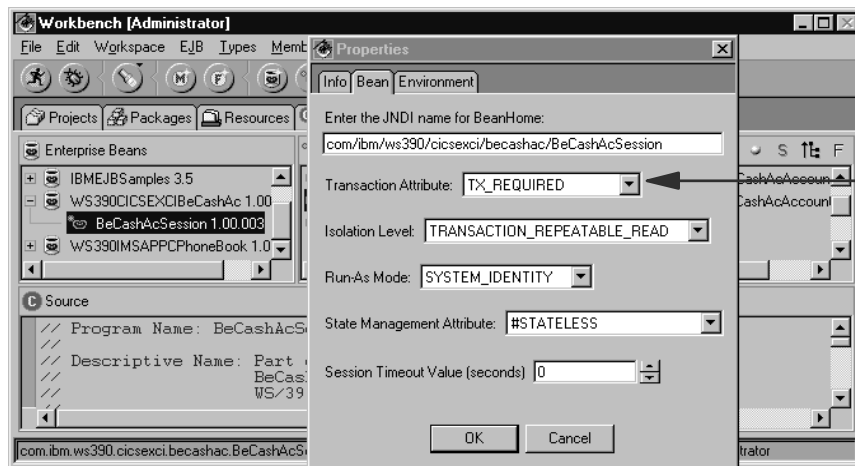


- **CICSEXCI supports two types of transaction handling :**
 - ▶ If the current thread is running under a global transaction, the current transaction context will be propagated across the CICSEXCI interface, and 2-phase commit or rollback processing will be coordinated using RRS.
 - ▶ If the current thread is not running under a transaction, the request will be sent to CICS indicating that processing performed for the request should be committed before returning.
- **Transaction attributes can be set at VAJ.**

CICS Connector Transaction Processing



Transaction Management Declaration

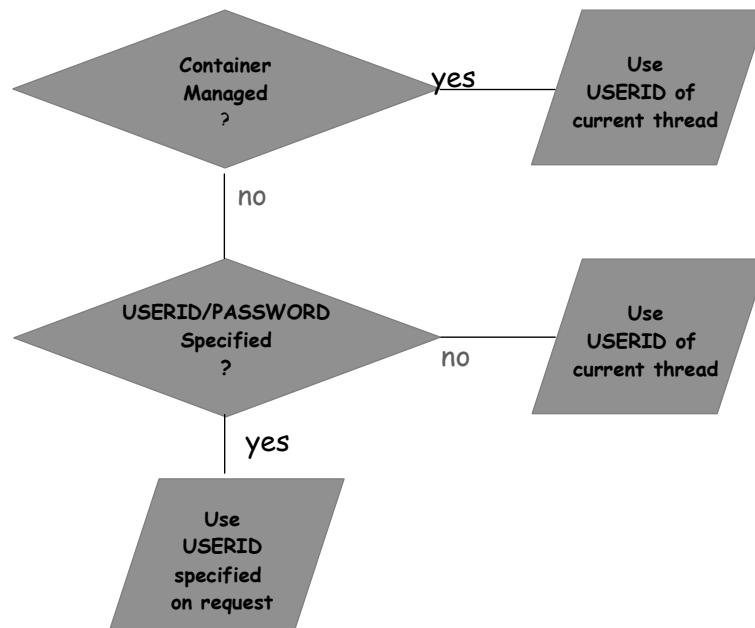


Connection Security in WAS V4.0.1



- **ResAuth - resource authentication**
 - ▶ Application authentication
 - ▶ Container authentication
- **Security identity is passed by the connection manager through the connector to the back end**

Connection Security: Identity



WAS/390 CICSEXCI Requirements



- **CICS TS V1.3 or higher**
- **CICS Transaction Gateway 4.02**
- **VisualAge for Java (VAJ) V4.0**
 - ▶ The VAJ BetaJ2EEConnectors support4.0 must be installed
- **WebSphere Studio Application Developer (WSAD) does not support building CICS applications to be deployed on WebSphere for 390.**

Installing the CICS Transaction Gateway



- **The code comes in a compressed tar file `ctg_402m.tar.Z`**
 - ▶ Uncompress the file and extract the files
 - `ctg-install` script
 - Readme files
 - ▶ Run the `ctg-install` script
 - creates several directories containing the jar files and DLLs
 - ▶ For WebSphere 390 the important directories are:
 - `ctg/deployable`
 - contains the resource files and runtime jar files
 - `ctg/classes`
 - contains jar files need for development
 - these are downloaded and installed into your VAJ development environment

Installing the CICS Transaction Gateway



■ CTG Connector resource file - **cicseciRRs.rar**

► unjar the cicseciRRS.rar file and extract the files:

- ccf2.jar
- cicseci.jar
- cicseciRRS.jar
- cicsframe.jar
- ctgclient.jar
- ctgserver.jar
- libCTGJNI.so
- libCTGJNI_g.so

► jar files will be added to the application server CLASSPATH

► .so (DLL) files will be added to the application server LIBPATH

CICS TS Requirements



■ An EXCI connection definition

- CTG supports both generic and specific connections
- If you use specific pipes, you must tell WebSphere what the NETNAME is using the DFHJVPIPE environment variable.

Specific Connection Definition



Define connection and session definition for EXCI (in CICS)

```
*****
* DEFINE connections
*****
DEFINE CONNECTION(BXCS) GROUP(BOSSEXCI)
  DESCRIPTION(S390-PAA EXCI SPECIFIC CONNECTION DEFINITION)
  NETNAME(WASASR2) —————> In WebSphere DFHJVPIPE=WASASR2
  ACCESSMETHOD(IRC)
  PROTOCOL(EXCI)
  CONNTYPE(SPECIFIC)
  INSERVICE(YES)
*****
* DEFINE sessions
*****
DEFINE SESSIONS(BXCS) GROUP(BOSSEXCI)
  DESCRIPTION(S390-PAA EXCI SPECIFIC SESSIONS DEFINITION)
  CONNECTION(BXCS)
  PROTOCOL(EXCI)
  RECEIVEPFX(RB)
  RECEIVCOUNT(004)
  OPERSECURITY(1)
  AUTOCONNECT(NO)
  BUILDCHAIN(YES)
  RELREQ(NO)
  DISCREQ(NO)
  INSERVICE(YES)
```

Configuring the CICS TS for CICS J2C EJBs



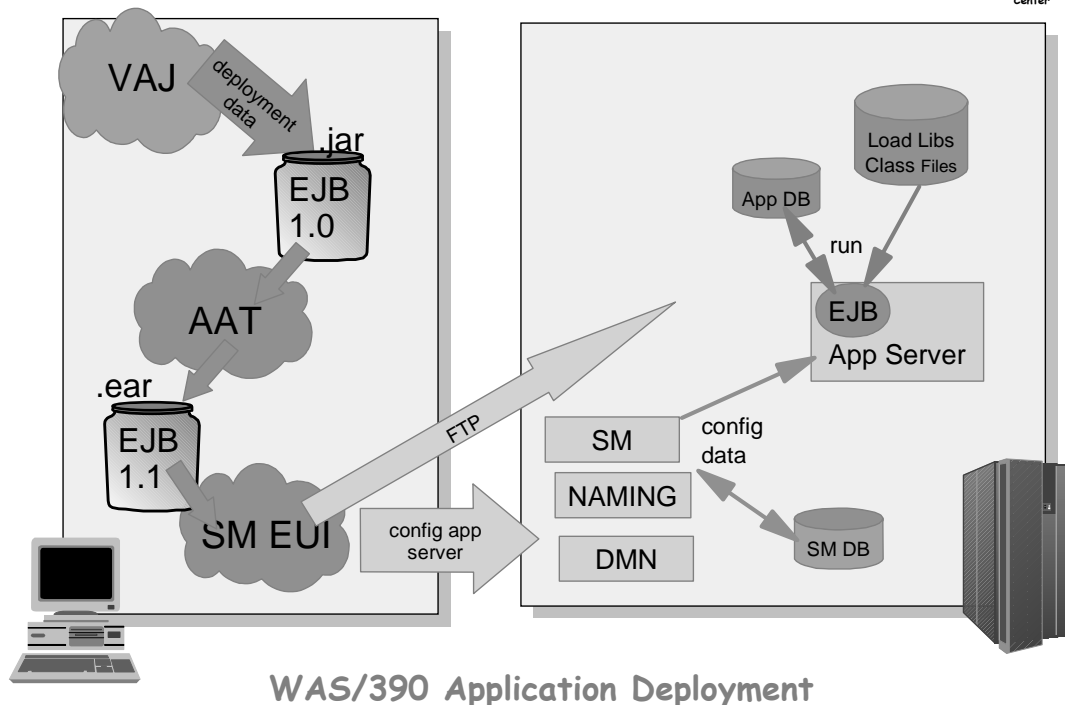
- Open Interregion Communication (IRC)
 - ▶ IRCSTRT=YES
- Register as a resource manager with RRMS
 - ▶ RRMS=YES
- RACF Definitions
 - ▶ RDEFINE SURROGATE *.DFHEXCI UACC(NONE) OWNER(profile-owner-id)
 - ▶ PERMIT *.DFHEXCI CLASS(SURROGATE) ID(wasuser) ACCESS(READ)
- The hlq.SDFHEXCI dataset must be available to the WebSphere Application Server regions
 - ▶ usually specified in LINKLIST

Application Assembly Tool (AAT)

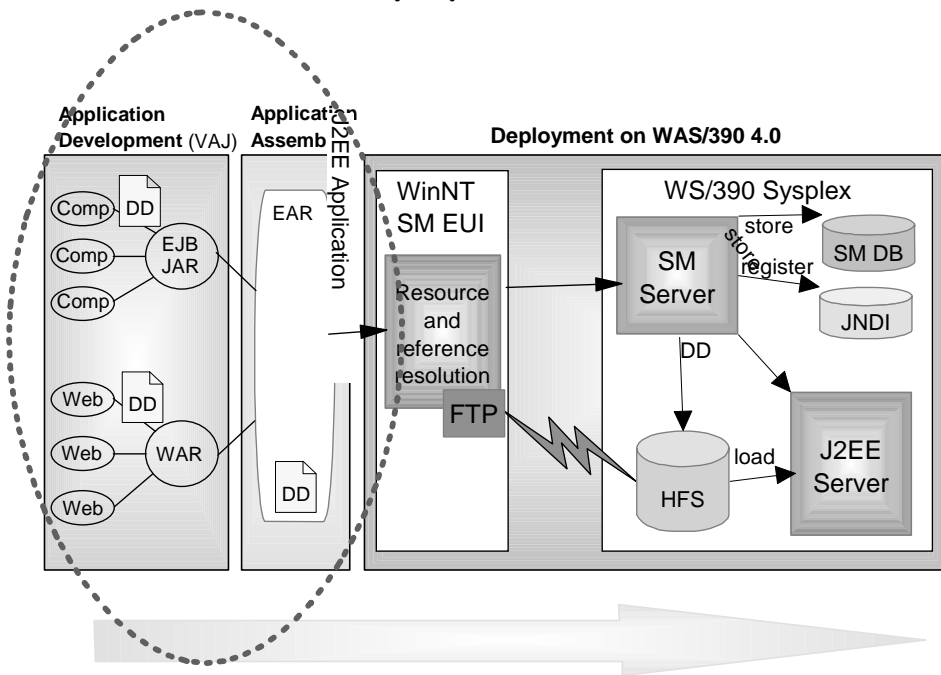


- **J2EE Application Assembly Tool**
 - ▶ Not used for CORBA applications
- **Graphical User Interface for assembling J2EE Modules**
 - ▶ (e.g. EjbJar v1.0, EjbJar v1.1, WebApp)

Creating a Deployable Application



J2EE Application Development, Assembly & Deployment



Application Assembly Tool (AAT)



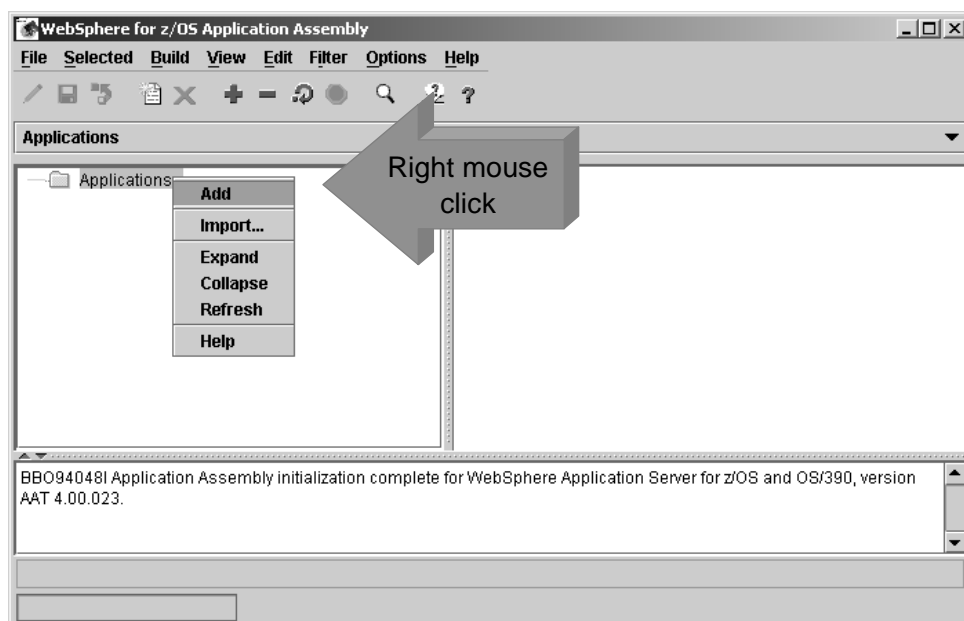
- Download from the website:
 - ▶ http://www.ibm.com/software/webservers/appserv/download_v4z.html
- A file, `aat##_setup.exe`, contains the AAT software.
- Installs on your workstation under
 - ▶ IBM WebSphere for z/OS

AAT Flow

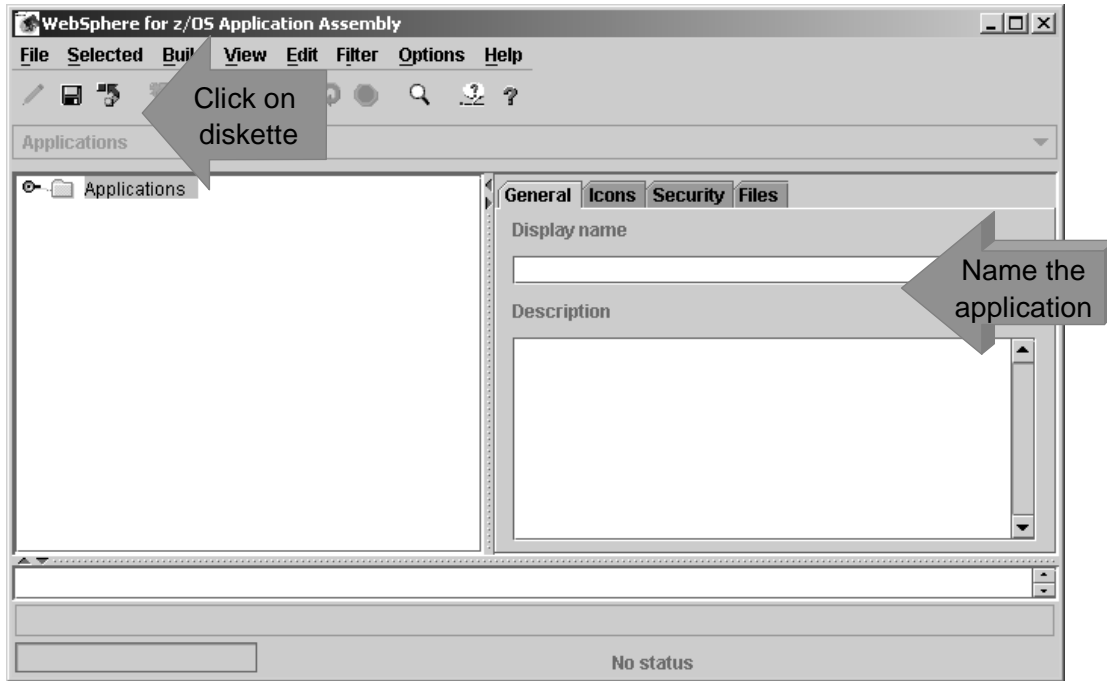


- **Create Application**
- **Import EJB .jar files**
 - ▶ Exported from VAJ or other tooling
- **Import web app .war files (if any)**
 - ▶ Exported from WebSphere Studio or other tooling
- **Resolve application deployment dependencies by modifying the properties of the EJB**
 - ▶ Resources
 - ▶ Security
 - ▶ etc.
- **Create the deployable .ear file**
 - ▶ Contains application and ejb xml descriptor files

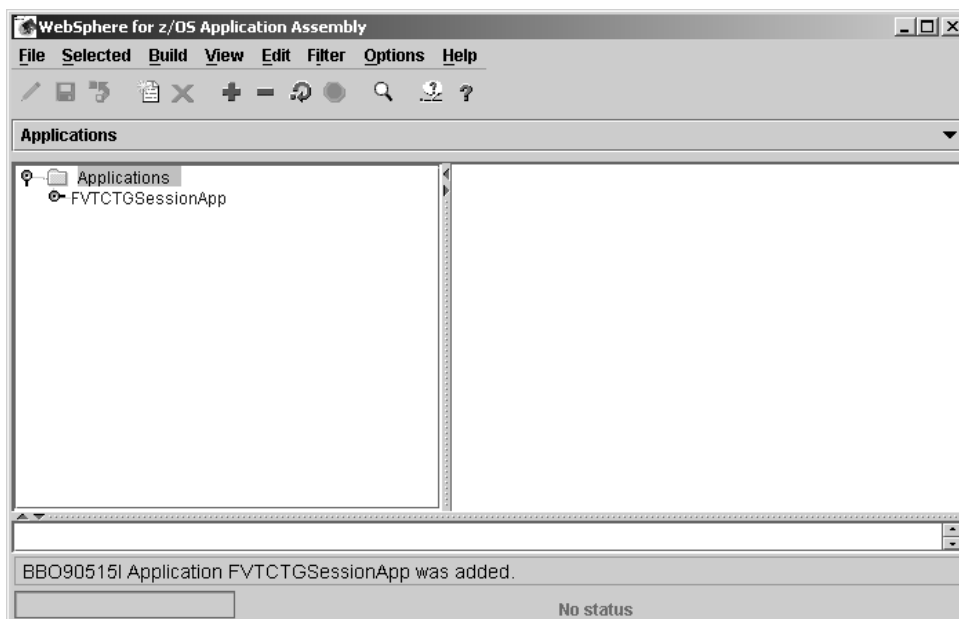
Adding an Application



Adding an Application



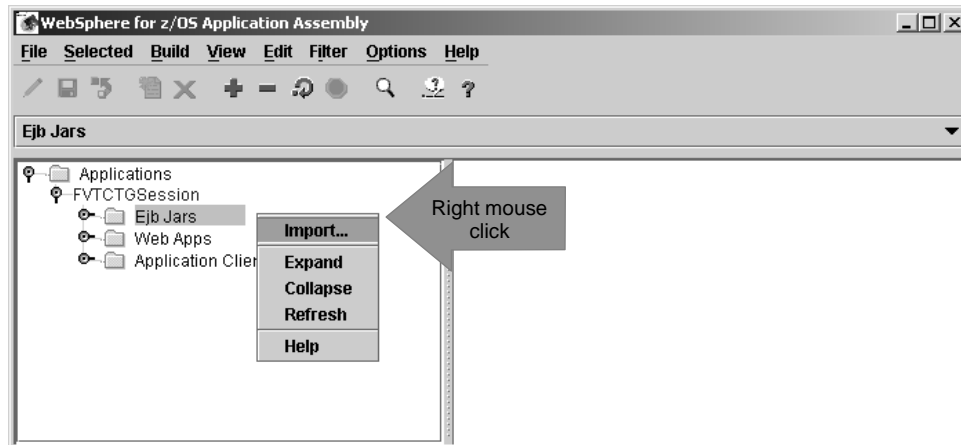
Application Assembly



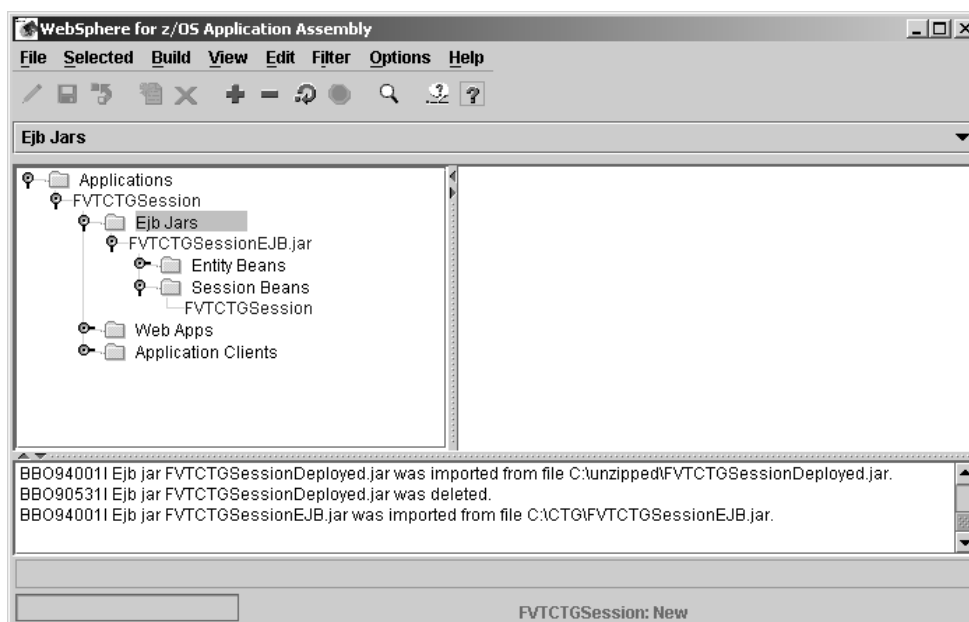
Import EJB .jar files



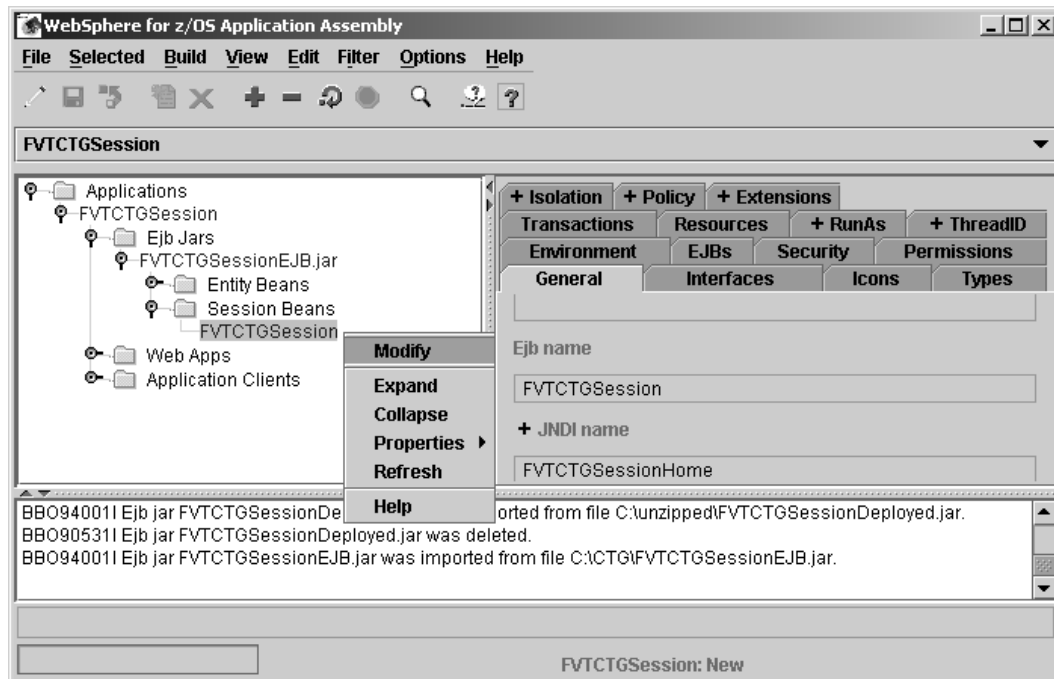
From the "Applications" folder within JTreePane, right click on the "Ejb Jars", select "Import..." to import an existing Enterprise Application into the AAT Workspace ...



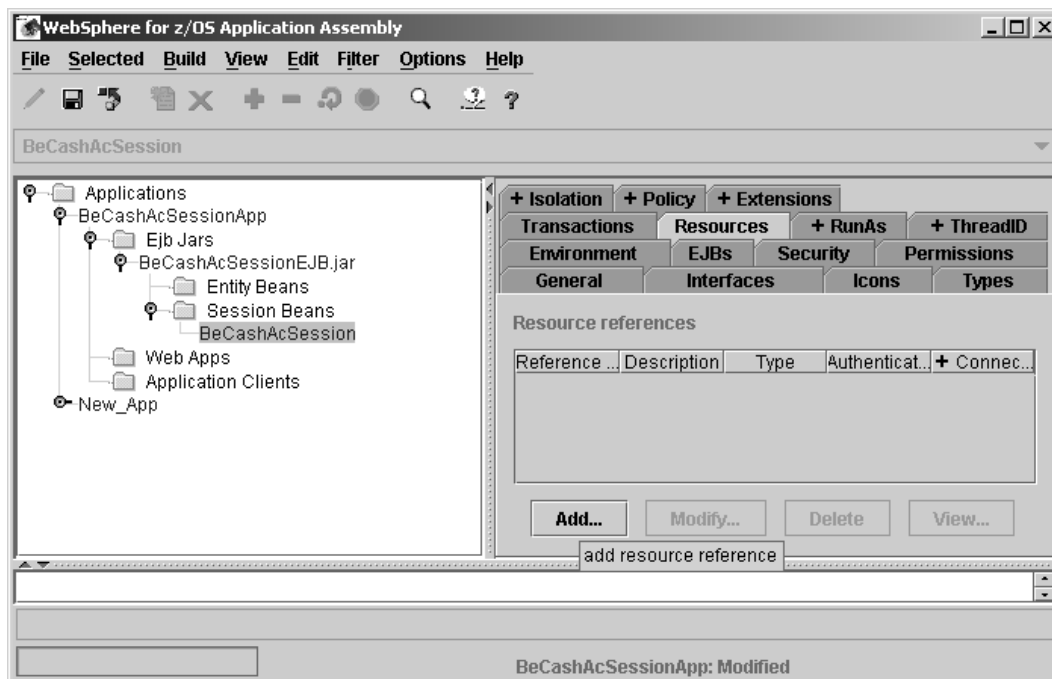
Expand



Define Properties



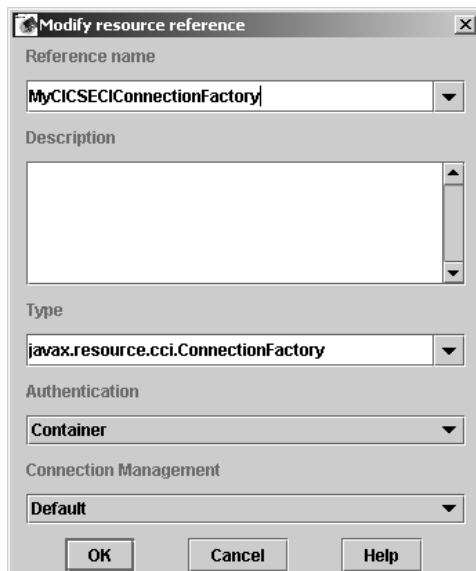
Add A Resource



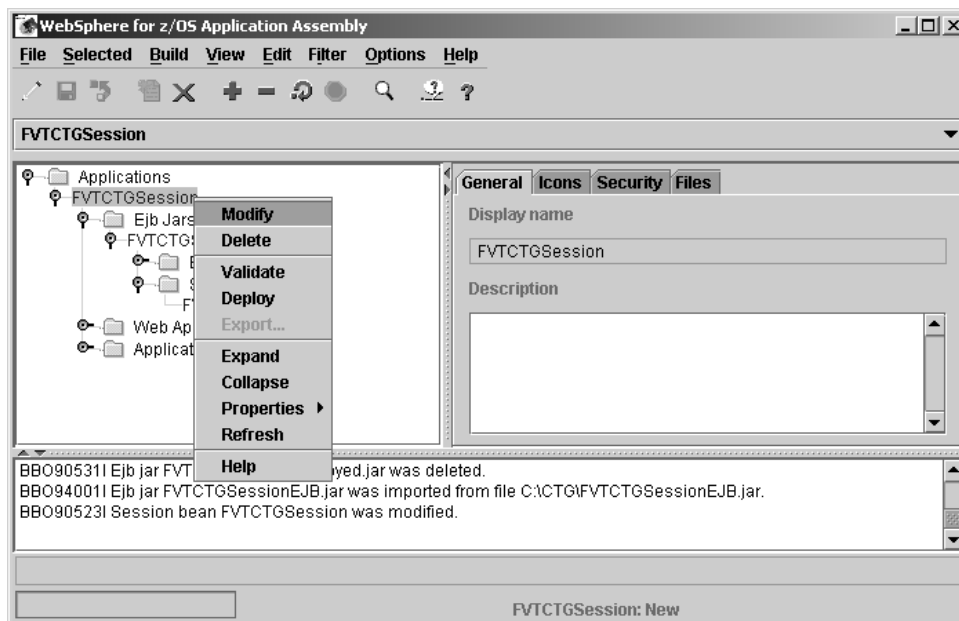
Connection Type



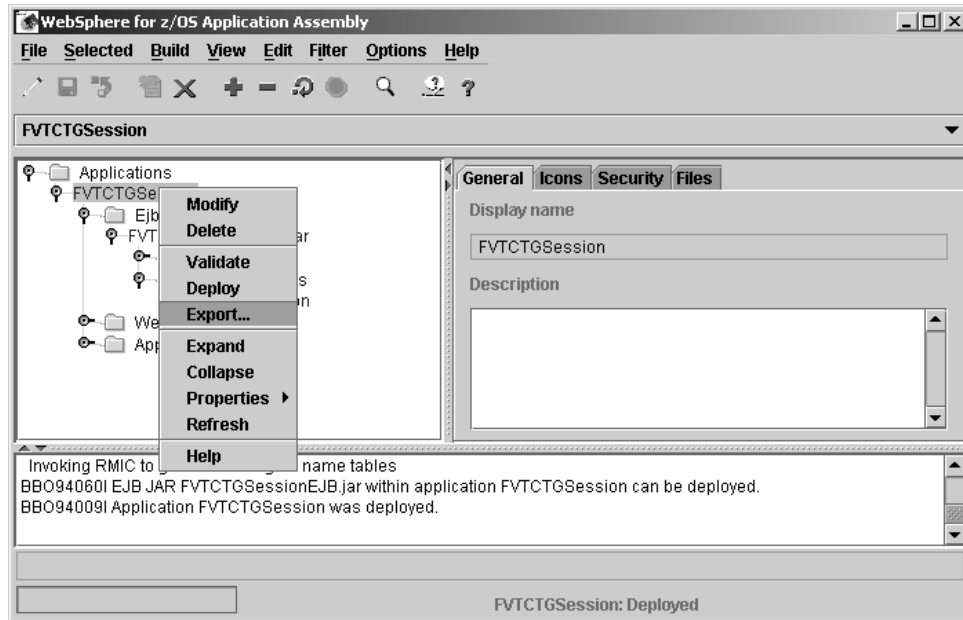
To resolve application deployment dependencies, such as:
Resources and Authentication



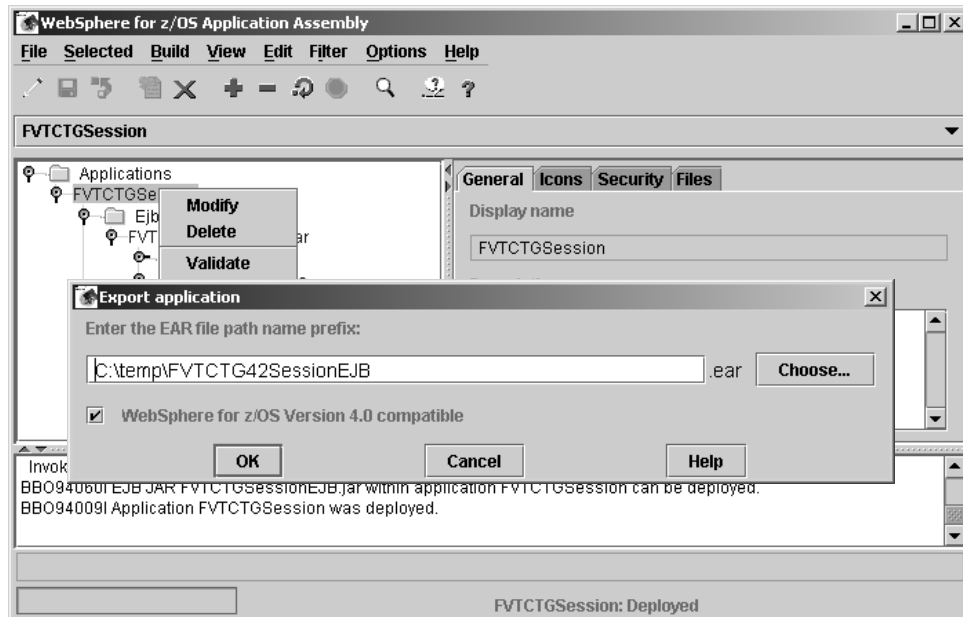
Validate



Export

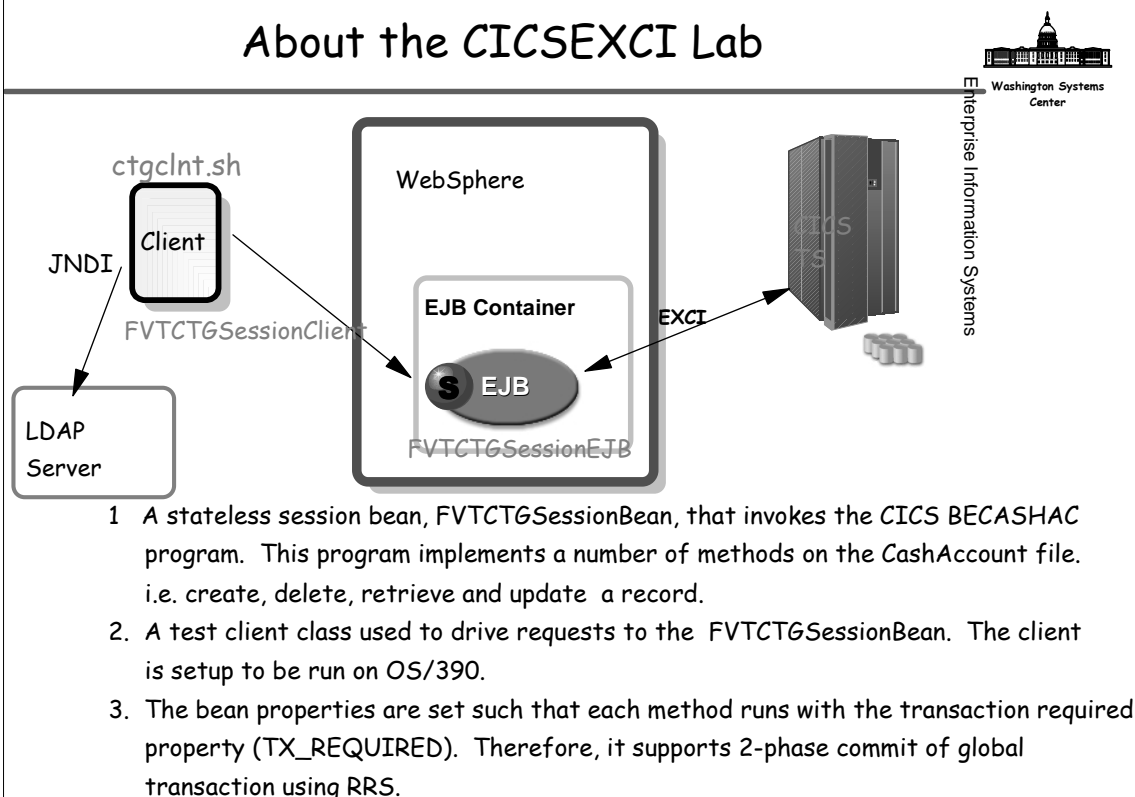


Export The ear File



LAB

About the CICSEXCI Lab



CICSEXCI Lab Flow



- Assemble the EJB for deployment using Application Assembly Tool(AAT).
- Install the EJB and required resources using Systems Management EUI.
- Create the FVTCTGSession client jar file.
- Run the provided USS shell script for the FVTCTGSession client on OS/390.

Questions?

