



IBM Software Group

WebSphere Business Process Management Suite V6.2

WebSphere Adapter for JD Edwards EnterpriseOne V6.2



@business on demand.

© 2009 IBM Corporation
Updated April 8, 2009

This presentation covers the IBM WebSphere® Adapter for JD Edwards EnterpriseOne V6.2.

Agenda

- Overview
- Installation
- Enterprise metadata discovery
- Business objects
- Summary and references



This presentation covers in detail the IBM WebSphere Adapter for JD Edwards (JDE) EnterpriseOne V6.2. It starts with the overview of JDE adapter with preparation steps in installation. Then it describes the roadmap in configuring a module in enterprise metadata discovery. It also captures different business object models with inbound and outbound operations. Last, JDE adapter provides the summary and references related to this specific adapter.

Section

Overview



This section will provide an overview of the WebSphere Adapter for JD Edwards EnterpriseOne V6.2

Overview

- The JD Edwards EnterpriseOne Resource Adapter enables outbound connectivity for integration to JD Edwards EnterpriseOne Applications
- Information is exchanged with the EnterpriseOne application through:
 - ▶ Business Function calls
 - Business function calls are core to EnterpriseOne interoperability. Business functions encapsulate transaction logic to perform specific tasks.
 - ▶ XML List calls
 - XML List is XML-based interoperability that runs as an EnterpriseOne process. It provides List functionality that enables retrieval of a list of records from EnterpriseOne.

The JD Edwards EnterpriseOne Resource Adapter enables outbound connection for integration to JD Edwards EnterpriseOne Applications. Information is processed with the EnterpriseOne application using one of two types of business objects: business functions and XML Lists. A business function is a business object container that can contain one or many business objects which can be processed as a single transaction. These business functions encapsulate transaction logic to perform specific tasks. For XML List call, it is XML-based interoperability that runs as an EnterpriseOne process. It provides List functionality that enables retrieval of a list of records from EnterpriseOne applications.

Overview



- The JD Edwards EnterpriseOne Resource Adapter enables inbound connectivity based on JDE real-time event mechanism
 - ▶ Runtime and external service wizard
 - Single type of JDE real-time events
 - Container type of JDE real-time events
 - ▶ JDE dynamic Java™ connector API
- Capabilities
 - ▶ Generate artifacts for real-time events in EMD
 - ▶ Process real-time events in runtime



In this release, WebSphere Adapter for JD Edwards EnterpriseOne supports asynchronous inbound processing. This means that the adapter polls the JD Edwards EnterpriseOne server at specified intervals for events. When the adapter receives an event, it converts the event data into a business object and sends the business object to the component.

WebSphere Adapter for JD Edwards EnterpriseOne V6.2 supports real-time event. A real-time event is a business transaction that provides information from the JD Edwards EnterpriseOne server that can be used to interoperate with a vendor system. Real-time events can be generated wherever business functions run, such as HTML, WIN32, and enterprise servers. Real-time events are useful for producing notifications in real-time. The adapter supports both single and container real-time events. Single type of real-time events are real-time events which do not include child real-time events. Container type of real-time events include some single type of real-time events.

The adapter gets a real-time event from the JD Edwards EnterpriseOne transaction server by invoking JD Edwards EnterpriseOne Dynamic Java Connector API. It generates artifacts for real-time event during enterprise metadata discovery and processes those real-time events in WebSphere runtime.

Section

Installation



This section will provide the installation preparation steps for WebSphere Adapter for JD Edwards EnterpriseOne.

Roadmap for configuring the module

- Package JDE Adapter Resource Archive file (CWYBC_JDE.rar) within WebSphere Integration Developer
 - ▶ Automatically import the adapter and create a Java EE connector project in the workspace
- Add EIS external dependencies to the RAR project
 - ▶ Many files, depending on JD Edwards version
- Add JDBC driver to access the database server



The WebSphere Adapter for JDE RAR file is now included in Resource Adapter directory of WebSphere Integration Developer. Once you have located the adapter, you will proceed with WebSphere Integration Developer and import the JDE archive into WebSphere Integration Developer. This creates a Java EE connector project in your workspace. Add any EIS external dependencies to the RAR project. You also need to add libraries from the connector module to the connector project class path.

Required client libraries

8.9 (SP1,SP2), 8.93	8.94	8.95 and 8.96	
kernel.jar	kernel.jar	connector.jar	ApplicationLogic_JAR.jar
connector.jar	connector.jar	JDBjBase_JAR.jar	jdeinterop.ini
database.jar	database.jar	JdbjInterfaces_JAR.jar	jde.ini
og4j.jar	log4j.jar	JdeNet_JAR.jar	jdelog.properties
xerces.jar	xerces.jar	Spec_JAR.jar	jdbj.ini
xalan.jar	xalan.jar	System_JAR.jar	Common_JAR.jar
deinterop.ini	jdeutil.jar	Base_JAR.jar	EventProcessor_EJB.jar
deLog.properties	jdbj.ini	log4j.jar	castor.jar
owra.jar	jdeinterop.ini	xerces.jar	JDBC driver
JDBC driver	jdelog.properties	xalan.jar	
	Common_JAR.jar	BizLogicContainer_JAR.jar	
	EventProcessor_EJB.jar	BizLogicContainerClient_JAR.jar	
	JDBC driver	ApplicationAPIs_JAR.jar	

- Copy files to a folder where adapter is installed
- See JD Edwards documentation – Connectors, and Interoperability
- jde.ini file needs to point to tnames.ora, and other props



Here is a list of required client libraries for the version of the JD Edwards software that you are using. Consult the JD Edwards documentation for connectors and interoperability properties. The jde.ini file needs to point to tnames.ora, along with other properties such as connection, user ID and password, and so on.

Section

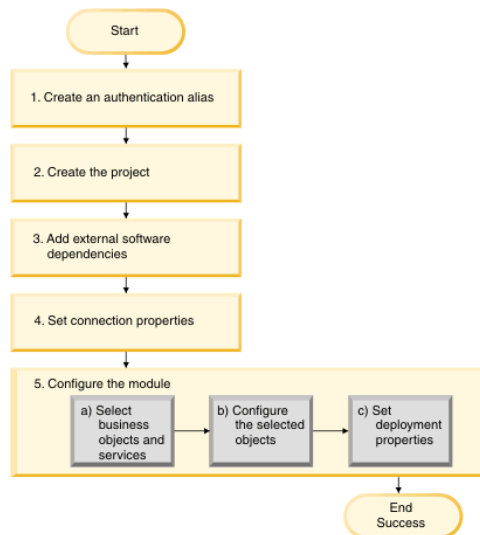
Enterprise metadata discovery



This section will provide an overview of the enterprise metadata discovery in JDE adapter.

Configuring the module

- Creates authentication alias
- Creates the project
- Adds external software dependencies
- Sets connection properties
- Configure for inbound or outbound processing



Before you can use WebSphere Adapter for JD Edwards EnterpriseOne V6.2 in a runtime environment, you must configure the module. You configure the module for the adapter to use by using the external service wizard in WebSphere Integration Developer. This figure illustrates the flow of the configuration task, and the steps that follow the figure describe this task at a high level.

First, create an authentication alias to access the JD Edwards EnterpriseOne server. This step is optional, depending on your policy for handling passwords and IDs. You then perform this step using the server. Then start the external service wizard in WebSphere Integration Developer to begin the process of creating and deploying a module. This project is used to organize the files associated with the module. Adding the external software dependencies are required by JDE adapter to the project. These dependencies are also required when you export the module as an EAR file, and deploy the EAR file to the server. In addition, set connection properties to connect to the JD Edwards EnterpriseOne server for discovery of objects and services

Last, configure the module for inbound or outbound processing to find and select business objects and services from the JD Edwards EnterpriseOne server, and to generate business object definitions and related artifacts.

Outbound configuration

- Query filter properties
 - ▶ Selection of business functions, XML lists with tables specified by user
- JDE API does not have the capability of retrieving the tables for which XML lists can be generated
 - ▶ Requires to manually enter the names of the tables
 - Table information can be retrieved with JD Edwards tools such as the universal table browser
 - If the table listed in the “tables for XML lists” property, there is no list in the tree under XML lists node



During the enterprise metadata discovery, you can edit the query to display query filter properties for outbound processing. It allows selection of business functions or XML lists with tables.

For outbound processing, you need to specify the name of the table. For the XML lists with tables, the API does not have the capability of retrieving the tables; you must manually enter the names of the table from which to generate the XML List business objects. You can retrieve the table information using JD Edwards tools such as the universal table browser.

For more details on how to configure outbound processing in enterprise metadata discovery, a separate demonstration is provided.

Outbound configuration: Business functions

- ▶ These properties is empty by default. Valid values are set to the list of all the business functions selected
- ▶ You can add/remove business functions
- ▶ The order specified is retained in the container business object operation ASI

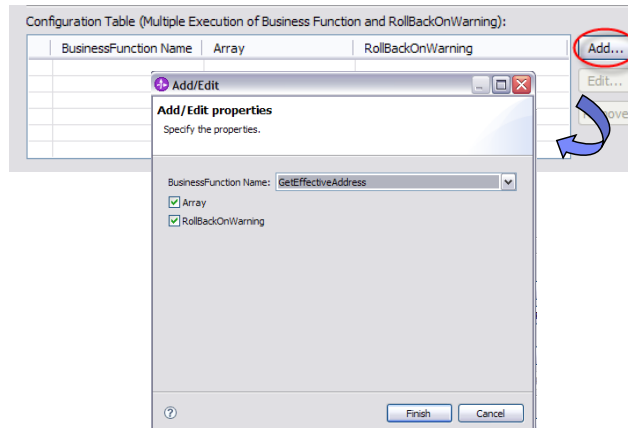
The screenshot shows a configuration window titled "Business Function container business objects". It contains three main sections:

- Container business object names:***: A text input field containing "GetMailingBF" with "Add..." and "Remove" buttons to its right.
- Operations:***: A list box containing "Create", "Delete", "Execute", "Retrieve", and "Update" with "Add..." and "Remove" buttons to its right.
- Business Functions:***: A text input field containing "GetMailingName" with "Add..." and "Remove" buttons to its right.

For the business function container business object, you will see the panel of enterprise metadata discovery, which allows you to provide the container business object name . You then need to add any business functions for Create, Retrieve, Update, Delete, and Execute operations and then you can select business functions and libraries from which to have your business object created. You can also add or remove these business functions. The order specified is retained in the “container business object operation application specific information.”

Outbound configurations: Multiple executions

- Support multiple executions of business function
 - ▶ Runtime and External Service Wizard
 - ▶ Allow to user to decide which business function to be execute multiply
 - ▶ Generated BO is marked as array type



The JD Edwards adapter supports multiple executions of business function. This new feature allows you to decide which business function is able to run multiple times. During enterprise metadata discovery, the generated business object definition marks the business function as array type; so that business function adds multiple instances in runtime and can be run multiple times. Refer to demonstration on how to configure this business function with multiple executions.

Outbound configuration: XML lists

- Add query capability
 - ▶ Sorting
 - ▶ Conditions
 - ▶ Where clause
 - ▶ Operator comparisons

Business object name: * F0116

Table type: OWTABLE

Table conversion version:

Operations: RetrieveAll

Queries

Query

Sorting conditions

Add sorting

Query Condition

Remove condition

Attribute: * Addressnumber

Clause: WHERE

Operator: Equal To

Use attribute value:

Default: 33937

Add Condition

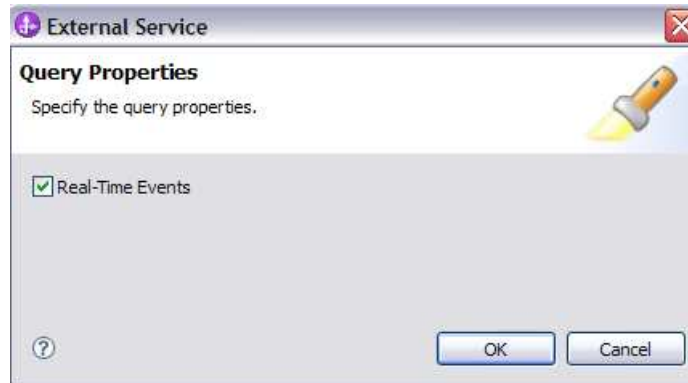
Remove Query

Add Query

For XML Lists, query capabilities include sorting, add conditions, the Where clause, operator comparisons and other default values. You also have the option to add multiple queries.

Inbound configuration: Real-time events

- Query filter properties
 - ▶ Real-time events

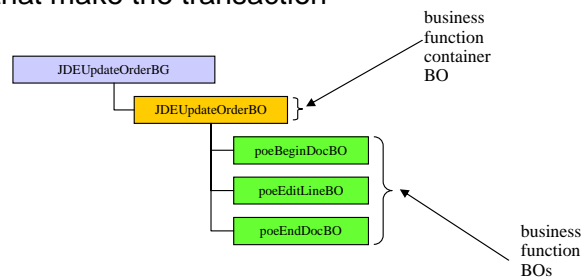


Similarly, you can also edit the query to display query filter properties for inbound processing. Real-time events is the only selection. You can just check that option for real-time event filtering.

For more details on how to configure inbound processing in enterprise metadata discovery, a separate demonstration is provided.

Outbound model – Business functions

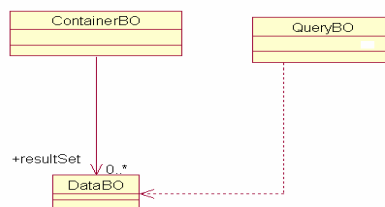
- The Business Function BOs as defined by the resource adapter are designed to support the execution of multiple EnterpriseOne business functions as a single transaction.
- The business function BO definitions map one-to-one to individual EnterpriseOne business functions. They contain attributes that map to the input and output parameters of the business function.
- Each business function container BO definition will contain references to the business function business objects that represent the business functions that make the transaction



Now look at the business object model for business functions and XML list. The business function business object, as defined by the resource adapter, are designed to support the execution of multiple EnterpriseOne business functions as a single transaction. So here you see the individual business function and individual business object. They are mapping one-to-one to those individual EnterpriseOne business functions. They contain attributes that map to the input and output parameters of the business function. Each business function container business object definition in the yellow box shown here will contain references to the business function business object that represent the business function that make the total transaction.

Outbound model – XML list

- The XML List BOs are designed to support the retrieval of multiple records from a specific JDE table or view based on parameterized SQL-like queries.
- For query support these requirements have been taken into consideration:
 - ▶ Build time definition of query structure
 - ▶ Support for more than one query expression per table
 - ▶ Right-hand operand values in conditions to be specified at runtime
 - ▶ Default values for right operands to be specified at build time, for cases where a static condition is preferred
 - ▶ Support for multiple occurrences of a column in a query expression
 - ▶ Support for >2 cardinality operators: IN, NI, BW (between) and so forth...
 - ▶ As such, structural distinction had to be made between the query BOs and the data BOs to hold the query result



Here is the business object model for XML list. The XML list business objects are designed to support the retrieval of multiple records from a specific JD Edwards table or view, based on parameterized SQL-like queries. These listed requirements need to be taken into consideration for query support. In business structure requirement, the structural distinction has to be made between the query business object and the actual data business object to hold the query result.

Outbound model – XML list (continued)

- The XML List Data BO definitions map one-to-one to EnterpriseOne tables. They are generated based on a table/table conversion structure and are reused across multiple queries on the same table.
- Each XML List QueryBO is based on one JDE Table/Table Conversion and its attributes reference the associated table's data business object. There might be multiple QueryBOs that are based on the same table/table conversion.
- XML List Query BOs are created at build-time, through ESD based on the corresponding data BO definition and user definition of the query structure. They are semantically similar to SQL stored procedures.
- QueryBO ASI contains information about what XML List Data BO it is based on (in DataType element)
- The Container BO will hold the result set of a query. ESD will generate one container BO definition per Data business object.



Here is more information on the XML list data business object and query business object, concerning how they map to JD Edwards tables.

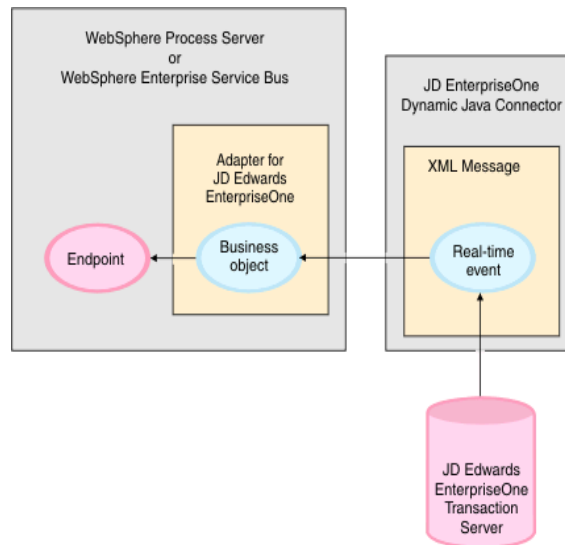
Outbound operations

- Business functions
 - ▶ Create
 - ▶ Retrieve
 - ▶ Update
 - ▶ Delete
 - ▶ Execute
- XML lists
 - ▶ RetrieveAll

When the adapter receives a request, it processes the request using the JD Edwards EnterpriseOne Dynamic Java connector to invoke either a business function or an XML list. The business functions include: Create, Retrieve, Update, Delete, and Execute. For XML lists, only RetrieveAll is available.

Inbound model – Real-time event

- Process real-time events
 - ▶ Invokes JDE dynamic Java connector API to get event
 - ▶ Parses the content of the event
 - ▶ Populates business object retrieving from the event
 - ▶ Sends generated business object to endpoint



The above figures illustrate how business objects are used by the adapter for JD Edwards EnterpriseOne for inbound processing. When the adapter gets a real-time event from the JD Edwards EnterpriseOne transaction server by invoking JD Edwards EnterpriseOne Dynamic Java Connector API, it parses the content of this real-time event and converts it into a business object. Then, the adapter populates the associated business object with the values it retrieves from the payload of this real-time event and sends the generated business object to the event endpoints. For example, if a company is updated, the JD Edwards EnterpriseOne server captures this change immediately, and one real-time event is generated by JD Edwards EnterpriseOne transaction server. The adapter then communicates with the JD Edwards EnterpriseOne transaction server, retrieves this real-time event, and processes it. After converting it into a business object, the adapter delivers this business object to the event endpoint.

Inbound model – Real-time event

- Supports event persistence using database eventstore
 - ▶ Set up data source using WebSphere Process Server
 - ▶ Assured-one delivery in case of failure
 - ▶ Set AssuredOnceDelivery property to True
- Supports event persistence using in-memory representation
 - ▶ No support for event recovery
 - ▶ Event processing is fast

Event delivery configuration	
Type of delivery:	ORDERED
Event types to process:	
Retry limit for failed events:	5
Number of connections for event delivery	
Minimum:	1
Maximum:	1
<input checked="" type="checkbox"/> AssuredOnceDelivery	
Event Persistence Data Source(JNDI) Name:	
<input checked="" type="checkbox"/> Create event persistence table	
The table with event persistence information:	JDETABLE
The schema name of the database where the table is stored:	
The user name used to connect to the database:	
The password used to connect to the database:	



The adapter supports event persistence for inbound processing in case of abrupt termination. Event persistence (or assured-once delivery) is a way to make sure that events are delivered once, and only once, to the endpoint in the case of a failure. During event processing, the adapter persists the event state in an event store located on the data source. You must set up this data source using WebSphere Process Server or WebSphere Enterprise Service Bus before you can create the event store. To use the recovery feature provided by WebSphere Process Server or WebSphere Enterprise Service Bus, you set the AssuredOnceDelivery property in the activation specification to true. This recovery feature is set to true by default.

The adapter also provides for event persistence using an in-memory representation of the event store. When you use this feature, you do not need to create a JNDI data source or an external event store, and event processing is faster. However, with this feature there is no support for event recovery. In the case of server failure, the in-memory event stores are lost.

To prevent the loss of events in the case of server failure, the recommended approach is to use the database event store. To use the in-memory event persistence capability of the adapter, you must not set the JNDI name property.

When a failed event occurs and the file can not be written to disk the JDE adapter will print the content of this failed event to a JDE trace file, to avoid event loss. The content is between two strings of ten hash marks and the trace level is INFO. The content can be copied into a new file and saved with the event ID as the file name in the directory "FailedEventFolder." When the status of the failed event record is changed from -1 to 0 in the event table, the event is handled again.

Inbound operation



- Operations
 - ▶ Create
 - ▶ Update
 - ▶ Delete



For Inbound; Create, Update, and Delete are the only three supported operations.

Section

Summary and references

The next section covers the summary and references.

Summary

- Summary
 - ▶ The JD Edwards EnterpriseOne resource adapter enables integration to JD Edwards EnterpriseOne applications
 - Introduced inbound support
 - ▶ Support for enterprise metadata discovery for discovering services
 - ▶ Learned different business object models
 - Business functions
 - XML lists
 - Real-time events

25

WebSphere Adapter for JD Edwards EnterpriseOne V6.2

© 2009 IBM Corporation

To summarize this presentation, you have learned that the JD Edwards EnterpriseOne Resource Adapter enables outbound and inbound connections for integration to JD Edwards EnterpriseOne Applications. Enterprise metadata discovery is used for discovery of services and creating the service description. It is also used to specify values for custom adapter properties and for the discovery of business objects. You also learned about different business object models for outbound and inbound processing through business functions, XML list calls and the real-time event mechanism.

Additional reference information can be found in the WebSphere Adapter for JD Edward EnterpriseOne user guide.

Reference information

- WebSphere adapter for JDE user guide
- Java connector architecture
 - ▶ <http://java.sun.com/j2ee/connector/index.jsp>
- Enterprise metadata discovery
 - ▶ <http://www.ibm.com/developerworks/java/library/j-emd/>
- WebSphere adapter information center
 - ▶ <http://www.ibm.com/software/integration/wbiadapters/library/infocenter/>
- WebSphere Process Integration information center
 - ▶ http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/topic/com.ibm.websphere.wps.620.doc/welcome_wps.html



Additional reference information can be found at these URLs.

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 e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WBIV62_IEA_AdapterJDE.ppt

This module is also available in PDF format at: [../WBIV62_IEA_AdapterJDE.pdf](..WBIV62_IEA_AdapterJDE.pdf)



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

Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:
WebSphere

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. 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>

Java, JDBC, and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2009. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.