IBM WebSphere® Data Interchange V3.3

Data Manipulation

@business on demand.

© 2007 IBM Corporation

This presentation will review WebSphere Data Interchange (WDI) Commands and Functions to manipulate data in a Send or Receive map .

Agenda

- Where do you start?

- Review Send/Receive mapping commands.

- Review Send/Receive mapping functions.

- Illustrate commands and functions.

- Summary and references

2

The presentation will discuss where you should start for help, review commands and functions, and illustrate the use of mapping commands and function..

Data Manipulation

- ## Where do you start?
  - ▸ WDI V3.3 Mapping Guide. Chapter 14
  - ▸ WDI Client Help contains the most up to date list of commands and functions.
    - ▪ Help->Contents->Mapping->Send Map->Mapping commands

The WDI Version 3.3 Mapping Guide, chapter 14, contains a list of commands and functions. The Client Help also contains this information.

Data Manipulation

The WDI Version 3.3 Mapping Guide.

Data Manipulation

WDI Client Help for Commands.

Section

WDI Mapping Commands

Data Manipulation

- Send/Receive Mapping Commands
    - Perform a specific action on the data.
    - Commands contain Expressions
    - The Syntax for an expression is token operator token
    - Special literal keywords supplied
    - Some literal keywords are specific to the Send or Receive map.

Mapping commands perform a specific action on the data. Commands contain Expressions. The Syntax for an expression is token operator token. Keywords are supplied to identify the mapping command or function. There are special keywords that do a specific mapping. Some keywords are available in a Send map and some are available in a Receive map.

Data Manipulation

- Expressions are literals, elements and variables that are combined using operators

- Expression yield a single value

- Expressions can use
    - Arithmetic operators: +, -, *, /
    - Logical operators: AND OR NOT
    - Relational operators: EQ, NE, LT, GT, LE, GE

- Expressions can be arbitrarily complex

- Example: &IF(Var1 NE 0 AND Var1 + Total < 500000)

Expression yield a single value. Expressions can use Arithmetic operators, Logical operators, and Relational operators. Expressions can be arbitrarily complex.

Data Manipulation

Commands are entered on the Literal or Mapping Command line. The Client mapping has field sensitive help which allows you to move to the literal line and press the F1 key on the keyboard to obtain help and navigate to the list of commands.

Data Manipulation

WebSphere Data Interchange for Multiplatforms V3.3

Contents | Index | Search

- WebSphere Data interchange Client
- Trading Partners
- Operational profiles
- Document definitions
- Rules and Usages
- Mapping
- Document Store
- Server Commands

Mapping Data Element Editor

The **Mapping Data Element Editor** is used to create and maintain most mappings in Send Maps and Receive Maps. It is opened from the Details tab page of the Send Map Editor and Receive Map Editor. Use the Mapping Data Element Editor to:

- Use an accumulator to store, count, and add values to a Data Format Field or Data Element. Accumulators allow you to perform such actions as counting Segments in a Transaction for later placement in a Transaction's trailer Segment.
- Associate a WebSphere Data Interchange literal or mapping commands with a Field or Data Element. Literals allow you to perform such actions as providing data to trading partners that your application does not contain.
- Use any of WebSphere Data Interchange's special handling options on a Field or Data Element. Special handling options allow you to perform such actions as editing dates, verifying data in a field against pre-defined lists, and converting data from one value to another.

The Data Format Path and Field fields identify a Data Format Field that is mapped to the Data Element. A value can not be entered into these fields. Instead, these fields are populated by dragging a field to the existing mapping and dropping it there, or when the mapping is created by dragging the Field to the Data Element and dropping it there. The Data Format Path field is always disabled. You can clear the two fields by selecting the Field field and then pressing the Delete key. Selecting the Field name from the drop-down in the Field field can restore the values.

Literals and mapping commands, as well as constants, are placed into the field

10

The Client mapping has field sensitive help which allows you to move to the literal line and press the F1 key on the keyboard to obtain help and navigate to the list of commands.

Data Manipulation

WebSphere Data Interchange for Multiplatforms V3.3

Hide   Back   Print   Options

Contents | Index | Search

- WebSphere Data interchange Client
- Trading Partners
- Operational profiles
- Document definitions
- Rules and Usages
- Mapping
- Document Store
- Server Commands

perform conditional processing, etc… Most commands require parameters, named variables, special variables, constants, and expressions. Mapping commands in Send Maps and Receive Maps are most commonly called **literals** or **special literals** because they are entered in the "Literal" field on the Mapping Data Element Editor. Literals sometimes refer to constants instead.

The following is a list of the mapping commands that are used in Send Maps and/or Receive Maps. These commands can be used in both map types unless otherwise noted. These commands are not used in Data Transformation Maps, Validation Maps, and Functional Acknowledgement Maps.

&ACFIELD	&ASSERT	&DATE
&DEFERRED	&E	&ERR
&FORCE	&FORMAT	&IF
&IFDATA	&IFNODATA	&IFNOVAR
&LOOPBREAK	&LOOPCHECK	&LSAVE
&LSET	&LSID	&SAMEAS
&SAVE	&SET	&THANDLE
&TIME	&TPID	&TPNICKN
&USE	&ZEROSIG	

start

11:23 AM
Thursday
3/1/2007

11

Data Manipulation

© 2007 IBM Corporation

You can select the command to view the format of the command or function.

Data Manipulation

WebSphere Data Interchange for Multiplatforms V3.3

Contents | Index | Search |

- WebSphere Data Interchange Client
- Trading Partners
- Operational profiles
- Document definitions
- Rules and Usages
- Mapping
- Document Store
- Server Commands

&DATE

Keyword	Keyword Description and Syntax
&DATE	Syntax: &DATE

Substitutes the system date. The length of the date field in the EDI Standard Data Element (Send Maps) or the Data Format Field (Receive Maps) determines whether the date is formatted as *yyyymmdd* or *yymmdd*. The date is then edited as requested by the date edit specified in mapping.

You can use the &DATE keyword as source data for any of the EDI Standard data types.

You can also combine the &DATE keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.

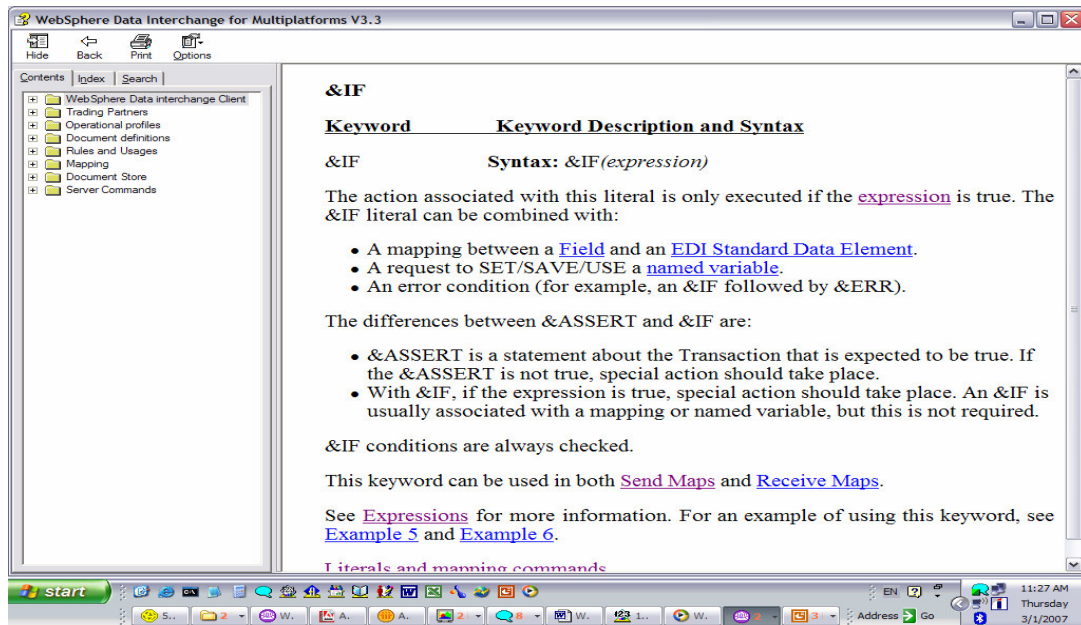This keyword can be used in both Send Maps and Receive Maps.

Literals and mapping commands

start | EN | 11:24 AM Thursday 3/1/2007

12

Data Manipulation

This is the syntax for the &DATE function.

Data Manipulation

```
WebSphere Data Interchange for Multiplatforms V3.3

Hide   Back   Print   Options

Contents | Index | Search |

WebSphere Data Interchange Client
Trading Partners
Operational profiles
Document definitions
Rules and Usages
Mapping
Document Store
Server Commands
```

&IF

Keyword   Keyword Description and Syntax

&IF   **Syntax: &IF**(*expression*)

The action associated with this literal is only executed if the expression is true. The &IF literal can be combined with:

- A mapping between a Field and an EDI Standard Data Element.
- A request to SET/SAVE/USE a named variable.
- An error condition (for example, an &IF followed by &ERR).

The differences between &ASSERT and &IF are:

- &ASSERT is a statement about the Transaction that is expected to be true. If the &ASSERT is not true, special action should take place.
- With &IF, if the expression is true, special action should take place. An &IF is usually associated with a mapping or named variable, but this is not required.

&IF conditions are always checked.

This keyword can be used in both Send Maps and Receive Maps.

See Expressions for more information. For an example of using this keyword, see Example 5 and Example 6.

Literals and mapping commands

```
start                                    EN         11:27 AM
                                                    Thursday
                                                    3/1/2007
```

Data Manipulation

13

© 2007 IBM Corporation

The If command format contains an expression.

Data Manipulation

WebSphere Data Interchange for Multiplatforms V3.3

Contents | Index | Search |

- WebSphere Data interchange Client
- Trading Partners
- Operational profiles
- Document definitions
- Rules and Usages
- Mapping
- Document Store
- Server Commands

&TPNICKN

Keyword   Keyword Description and Syntax

&TPNICKN   Syntax: &TPNICKN

Substitutes the name of the Trading Partner profile.

You can also use &TPID keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.

This keyword can be used in Send Maps and Receive Maps.

Literals and mapping commands

Data Manipulation   © 2007 IBM Corporation

14

Some literal keywords do a specific mapping.

Data Manipulation

&FORCE

Keyword	Keyword Description and Syntax
&FORCE	Syntax: &FORCE *value*

Forces a value into a Data Format Field regardless of the EDI Standard Data Element's contents. A literal value specified for Receive Maps is normally used only if the Data Element does not contain any data, or if an error occurs while processing the data. See literals and mapping commands for special considerations.

For the &FORCE keyword to be effective, you must use it in a Data Element of a Segment that is present in the input Transaction.

This keyword can be used in Receive Maps only.

Literals and mapping commands

Data Manipulation

15

Some commands for example &FORCE are only available in Receive maps.

Section

WDI Mapping Functions

Data Manipulation

IBM Confidential
IBM Software Group

IBM

Data Manipulation

- Send Receive Mapping Functions
  - ▶ Used to manipulate the data or assign values.
  - ▶ All functions take zero or more arguments as input and return a value.
  - ▶ Some functions have optional parameters.

17

Mapping functions are used to manipulate data or assign values. Some functions have no arguments. All functions return a value.

Data Manipulation

The &USE function has some optional parameters or arguments for example position, length, and default value.

Data Manipulation

- Send Receive Mapping Functions: Field Manipulation
  - Scenario Outbound:
    - New purchasing system (outbound order) using legacy system for inbound invoice processing. Invoice is matched to order using PO number.
    - Legacy system PO number always begins with '4' and cannot exceed 7 positions. The new purchasing system PO number is 10 positions and does not begin with '4'.
    - Orders are mirrored to legacy system using the legacy notation for PO number.

19

© 2007 IBM Corporation

In this example, we have a new purchasing system which must match the purchase order number with the invoice system. The new purchasing system has purchase order number with length 10. The invoice system is expecting purchase order numbers with length 7 and beginning with the number 4.

IBM Software Group

Data Manipulation

- Send Receive Mapping Functions: Field Manipulation
  - ▶ Solution:
    - Change the PO Number on the outbound order.
    - PO Number = 4 + last 6 digits of PO number.

    - Data Example:
    - PO number before = 5200000799
    - PO number after = 4000799

One solution is to change the purchase order number on the outbound order so the purchase order number in the returning invoice will match what is in the order.

Data Manipulation

- Send/Receive Mapping Functions: Field Manipulation
  - ▶ Steps Send/Receive Map:
    - ✓ Save the PO number.
      - ✓ &SAVE PONUMBER
    - ✓ Use scaling to get the last 6 characters of the PO number.
      - ✓ &SET PONUMBER &E(CHAR(PONUMBER) SC 5.6)
    - ✓ Concatenate '4' with the last 6 characters of the PO number.
      - ✓ &SET PONUMBER &E('4' + CHAR(PONUMBER))
    - ✓ Map the result
      - ✓ &USE PONUMBER

The steps to manipulate the purchase order number are:

1. Save the purchase order number in a variable.

2. Save the last six characters of the purchase order number using the Scaling operator

3. Concatenate the number four with the last six characters of the purchase order number

4. Map the result

Data Manipulation

First you need to save the purchase order number to the variable SPONUMBER. Variable names may not begin with the letter P. To save the value, use the drag and drop on the purchase order number and enter the Literal Command &SAVE.

Data Manipulation

Next you get the last 6 digits from the purchase order number. To add another mapping command double click on the element in the right window.

Data Manipulation

Save the Result ←————— Evaluate (expression)

▪ &SET SPONUMBER &E(CHAR(SPONUMBER) SC 5.6)

Variable Name   Force Character type

Substring position 5 for length 6

This is the command that was entered on the literal or command line. The &E is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. A value with all digits is normalized to numeric data type automatically. The CHAR is an operator that forces the variable SPONUMBER which contains the purchase order number we saved in the previous mapping command, to be treated as a character value. The SC operator is a special operator used for scaling real numbers but can be used when dealing with character data to provide a substring capability. The 5.6 is the position and length for the substring. The &SET keyword indicates to save the result in the variable SPONUMBER.

Next you concatenate the value '4' to the purchase order number.

Data Manipulation

Save the Result ←———————— Evaluate (expression)

- &SET SPONUMBER &E('4' + CHAR(SPONUMBER))

Variable Name · Force Character type

Concatenate operator

This is the command entered on the literal or command line. The &E is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. A value with all digits is normalized to numeric data type automatically. The CHAR is an operator that forces the variable SPONUMBER which contains the purchase order number we saved in the previous mapping command, to be treated as a character value. The plus operator is a numeric operator but can be used when dealing with character data to provide a concatenation capability. The &SET keyword indicates to save the result in the variable SPONUMBER.

Data Manipulation

The &USE keyword will use the value in the variable SPONUMBER.

Data Manipulation

The completed mapping.

IBM Software Group

Data Manipulation

- Send Receive Mapping Functions: Field Manipulation
  - ▶ Scenario Outbound:
    - My trading partner needs machine number and model number in the PO line item segment for all 7 digit buyer's part numbers. Buyer's part numbers can be up to 35 characters.
    - If the buyer's part number is 7 digits the first 4 digits of the part number is machine number and the last 3 digits of the part number is model number

In this example, my trading partner needs the machine number and model number in the purchase order line item segment for all 7 digit buyer's part numbers. The Buyer's part numbers can be up to 35 characters. If the part number is 7 digits, the first 4 digits is machine number and the last 3 digits are the model number.

Data Manipulation

- Send Receive Mapping Functions: Field Manipulation
  - ▶ Solution:
    - If the buyer's part number is 7 digits
    - Machine Number = first 4 digits
    - Model Number = last 3 digits

    - Data Example:
      - Buyer's part number = 6400XYZ
      - STD before = PO1*00010*1*EA*0.01*1 *BP*6400XYZ!
      - STD after = PO1*00010*1*EA*0.01*1 *BP*6400XYZ*****MA*6400*MN*XYZ!

The solution is to check the length of the part number. If the length is 7, you need to split the part number to create the machine number and model number.

IBM Software Group

Data Manipulation

- Send/Receive Mapping Functions: Field Manipulation
  - ▶ Steps Send/Receive:
    - Save the buyer's part number.
      - &SAVE partnum
    - Check to see if length 7
      - &SET mapit &E(CHAR(partnum) SC 8.1)
    - Use scaling to get machine number
      - &IF (mapit = '') &SET machinetype &E(CHAR(partnum) SC 1.4)
    - Use scaling to get model number
      - &IF (mapit = '') &SET modeltype &E(CHAR(partnum) SC 5.3)
    - Map machine number
      - &USE machinetype
    - Map model number
      - &USE modeltype

Data Manipulation   © 2007 IBM Corporation

The steps to manipulate the part number are:

1. Save the part number in a variable

2. Check to see if the length is 7.

3. Use the scaling operator to get the machine number

4. Use the scaling operator to get the model number

5. Map the results

Data Manipulation

First save the buyer's part number to the variable bpartnum. Variable names may not begin with the letter P. To save a value, use the drag and drop on the buyer's part number and enter the Literal Command &SAVE.
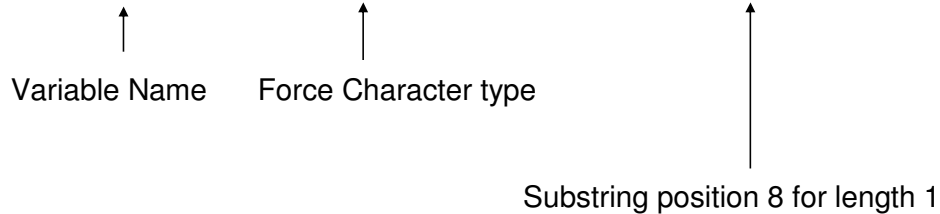
Data Manipulation

Next you need to try and determine the length of the part number. To add another mapping command double click on the element in the right window.

Data Manipulation

Save the Result ←————— Evaluate (expression)

• &SET mapit &E(CHAR(bpartnum) SC 8.1)

Variable Name   Force Character type

Substring position 8 for length 1

This is the command entered on the literal or command line. The &E is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. A value with all digits is normalized to numeric data type automatically. The CHAR is an operator that forces the variable bpartnum which contains the purchase order number we saved in the previous mapping command, to be treated as a character value. The SC operator is a special operator used for scaling real numbers but can be used when dealing with character data to provide a substring capability. The 8.1 is the position and length for the substring. The &SET keyword indicates to save the result in the variable mapit. If there is a character is position 8 then the variable mapit will contain a value which will indicate the part number length is greater than 7.

Data Manipulation

Now you can add an If command to see if we should split the part number. This command will set the machine number in the variable machinetype

Data Manipulation

Save the Result ← Evaluate (expression)

If (true)

- &IF (mapit = '') &SET machinetype &E(CHAR(bpartnum) SC 1.4)

Variable Name
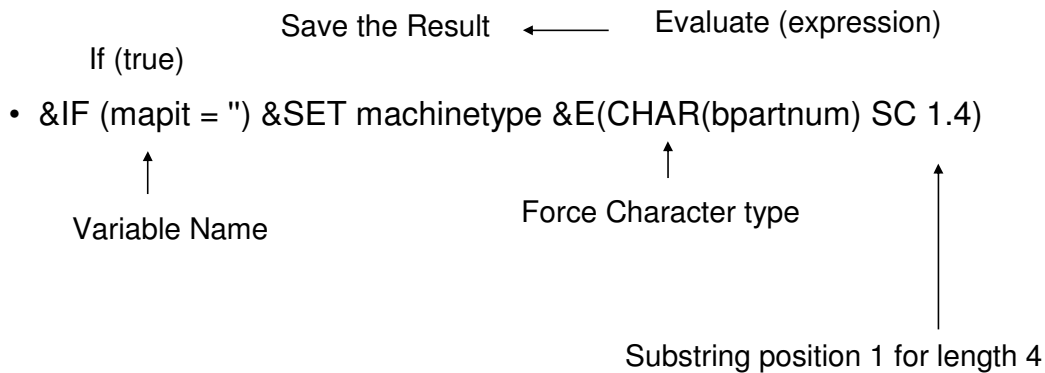
Force Character type

Substring position 1 for length 4

This is the command entered on the literal or command line. The &IF is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. In our example the variable mapit is null if there is not a digit in position 8. The mapit variable was set in the previous mapping command.

The &E is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. A value with all digits is normalized to numeric data type automatically. The CHAR is an operator that forces the variable bpartnum which contains the purchase order number we saved in the previous mapping command, to be treated as a character value. The SC operator is a special operator used for scaling real numbers but can be used when dealing with character data to provide a substring capability. The 1.4 is the position and length for the substring. The &SET keyword indicates to save the result in the variable machinetype.

Data Manipulation

Now you can add an If command to see if we should split the part number. This command will set the model number in the variable modeltype.

Data Manipulation

Save the Result ◄─────── Evaluate (expression)

If (true)

- &IF (mapit = ") &SET modeltype &E(CHAR(bpartnum) SC 5.3)

Variable Name

Force Character type

Substring position 5 for length 3

This is the command entered on the literal or command line. The &IF is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. In our example the variable mapit is null if there is not a digit in position 8. The mapit variable was set in the previous mapping command.
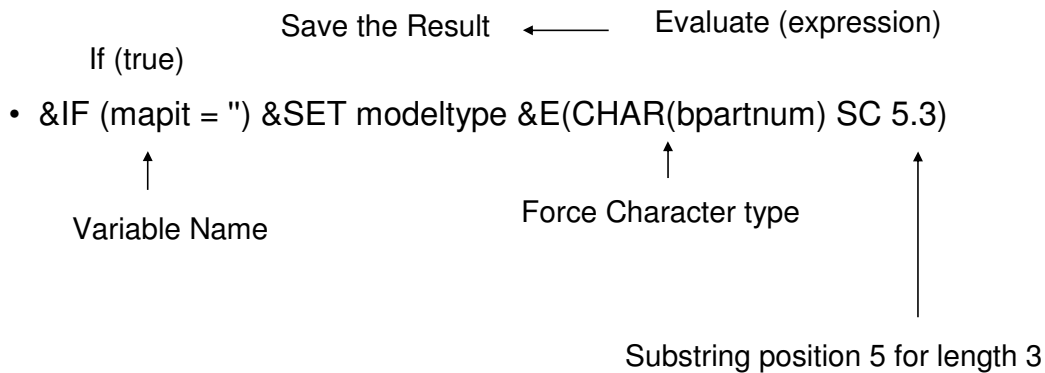
The &E is a mapping command and take an expression as an argument. It evaluates the expression and returns the result. The expression is within the first set of parentheses. A value with all digits is normalized to numeric data type automatically. The CHAR is an operator that forces the variable bpartnum which contains the purchase order number we saved in the previous mapping command, to be treated as a character value. The SC operator is a special operator used for scaling real numbers but can be used when dealing with character data to provide a substring capability. The 5.3 is the position and length for the substring. The &SET keyword indicates to save the result in the variable modeltype.

Data Manipulation

Next you map the results. The &USE keyword will use the value in the variable machinetype.

IBM Software Group

Data Manipulation

The &USE keyword will use the value in the variable modeltype.

The completed mapping.

Summary

- Where do you start?
  - ▸ WDI V3.3 Mapping Guide. Chapter 14
  - ▸ WDI Client Help contains the most up to date list of commands and functions.
    - Help->Contents->Mapping->Send Map->Mapping commands

42

More information can be found in the WebSphere Data Interchange Version 3.3 Mapping Guide chapter 14.

IBM

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	WMQ	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

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

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and 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 and/or changes in the product(s) and/or program(s) 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.

Information is provided "AS IS" without warranty of any kind. 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 (e.g., 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 2006. 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.

IBM Software Group