



IBM Tivoli Workload Scheduler

Change password utility



© 2009 IBM Corporation
Updated July 20, 2009

IBM Tivoli Workload Scheduler

Change password utility

Abstract

- This session discusses function of the *chg_pass.sh* script, requirements, operation and implementation to facilitate WebSphere required procedures for Tivoli Workload Scheduler for TWS or DB user password changes. It will focus on:
 - ▶ Script function
 - ▶ Script requirements
 - ▶ Customizable Variables
 - ▶ Non-customizable Variables
 - ▶ Operation
 - ▶ Implementation
 - ▶ Troubleshooting

Note: A demonstration of the *chg_pass.sh* script will be also presented at the end of this session.

Abstract

This session discusses function of the *chg_pass.sh* script, requirements, operation and implementation to facilitate WebSphere required procedures for Tivoli Workload Scheduler for TWS or DB user password changes. It will focus on:

Script function

Script requirements

Customizable Variables

Non-customizable Variables

Operation

Implementation

Troubleshooting

Note: A demonstration of the *chg_pass.sh* script will be also presented at the end of this session.

Chg_pass.sh script function

The *chg_pass.sh* script is used to change WebSphere Security Properties when either TWS or the DB2® or Oracle Database user passwords have changed.

The *chg_pass.sh* script may be run from the UNIX® prompt or as a defined job on either a Tivoli Workload Scheduler Fault Tolerant Agent (FTA), Domain Manager (DM), Backup Domain Manager (BKM) or Master Domain Manager (MDM) workstation.

Chg_pass.sh script function

The *chg_pass.sh* script is used to change WebSphere Security Properties when either TWS or the DB2 or Oracle Database user passwords have changed.

The *chg_pass.sh* script may be run from the UNIX prompt or as a defined job on either a Tivoli Workload Scheduler Fault Tolerant Agent (FTA), Domain Manager (DM), Backup Domain Manager (BKM) or Master Domain Manager (MDM) workstation.

Script requirements

The script may exist in any directory.

The /tmp directory must exist on workstation since script created temporary files will be created in that directory.

Temporary files will be removed when script completes normally.

The “thiscpu” variable in the TWSHome/localopts file and entries in /etc/TWS/TWSRegistry.dat file for specified TWS user must be correct and valid.

The *chg_pass.sh* always requires the TWS user name, and TWS user password. If the DB user password changed, the TWS user name, TWS user password, DB user and DB password are required when executing script.

Script requirements

The script may exist in any directory.

The /tmp directory must exist on workstation since script created temporary files will be created in that directory.

Temporary files will be removed when script completes normally.

The “thiscpu” variable in the TWSHome/localopts file and entries in /etc/TWS/TWSRegistry.dat file for specified TWS user must be correct and valid.

The *chg_pass.sh* script requires the TWS user name. TWS user password is optional if only the DB user password changed. The DB password is required when DB user is specified.

Script requirements (continued)

Chg_pass.sh script must be executable and executed by the root user. This requirement is necessary since WebSphere operations performed by the script require root access.

The *chg_pass.sh* script **must run** shortly after password has changed for either the TWS or DB user.

If script is defined as a Tivoli Workload Scheduler job, script path must be specified in job definition and job logon user must be **root**.

Note: Defining *chg_pass.sh* script as a scheduled Tivoli Workload Scheduler job in advance permits scheduling execution of required WebSphere changes when the password changes for TWS or DB user otherwise script will need to be manually executed soon after password has changed.

Once the *chg_pass.sh* has completed successfully normal operations may resume.



Chg_pass.sh script must be executable and executed by the root user. This requirement is necessary since WebSphere operations performed by the script require root access.

The *chg_pass.sh* script **must run** shortly after password has changed for either the TWS or DB user.

If script is defined as a Tivoli Workload Scheduler job, script path must be specified in job definition and job logon user must be **root**.

Note: Defining *chg_pass.sh* script as a scheduled Tivoli Workload Scheduler job in advance permits scheduling execution of required WebSphere changes when the password changes for TWS or DB user otherwise script will need to be manually executed soon after password has changed.

Once the *chg_pass.sh* has completed successfully normal operations may resume.

Customizable variable - Debug

The debug option will provide a more verbose output of tasks performed by the script.

The debug option is invoked by setting "DEBUG" variable to "y" in the Debug Mode Option section.

The verbose output will default to stdout. If *chg_pass.sh* script is executed from a Tivoli Workload Scheduler job then stdout will be in the job stdlist. If executed from UNIX prompt, stdout may be re-directed to a file.

Variable: DEBUG

```
#***** Customizable options *****
#
#Debug Mode Option
#To place in debug mode set DEBUG to y
DEBUG="";export DEBUG
```

Customizable variable – Debug

The debug option will provide a more verbose output of tasks performed by the script.

The debug option is invoked by setting "DEBUG" variable to "y" in the Debug Mode Option section.

The verbose output will default to stdout. If *chg_pass.sh* script is executed from a Tivoli Workload Scheduler job then stdout will be in the job stdlist. If executed from UNIX prompt, stdout may be re-directed to a file.

This slide shows a section for the Variable DEBUG option.

Redirecting STDOUT

If *chg_pass.sh* script is executed from UNIX prompt, stdout may be re-directed to a flat file by performing the following steps:

- 1) Edit *chg_pass.sh* script and set `DEBUG="y"`
- 2) Issue the UNIX "script" command and specify file that will contain output.

Example:

```
script -a filename
```

- 3) Execute script:

Example:

```
chg_pass.sh -twuser m83rtm -twspass passw0rd
```

- 4) Once script completes, issue a `<ctrl> d` to terminate capturing of stdout to file.

Example:

```
<ctrl> d
```

Redirecting STDOUT

If *chg_pass.sh* script is executed from UNIX prompt, stdout may be re-directed to a flat file by performing the following steps:

- 1) Edit *chg_pass.sh* script and set `DEBUG="y"`
- 2) Issue the UNIX "script" command and specify file that will contain output.

Example:

```
script -a filename
```

- 3) Execute script:

Example:

```
chg_pass.sh -twuser m83rtm -twspass passw0rd
```

- 4) Once script completes, issue a `<ctrl> d` to terminate capturing of stdout to file.

Example:

```
<ctrl> d
```

Non-customizable variables

The following variables must not be customized, changes to these variables will cause problems with the script:

```

USAGE="Usage:\nFor TWS password change:\n-twsuser <twsuser-id> -twspass <password> \n\nFor DB password
change:\n-twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password> \n\nFor TWS and DB
password change:\n-twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password> \n";export
USAGE

#Capture the original provided options
ORIG_SYNTAX="$*";export ORIG_SYNTAX

#Valid TWS versions
TWS_CURR_VERSIONS="8.3.0\n8.4.0";export TWSVERSIONS

#Necessary WAS tool scripts
WAS_TOOL_SCRIPTS="\nstopWas.sh\nstartWas.sh\nshowSecurityProperties.sh\nchangeSecurityProperties.sh";export
WAS_TOOL_SCRIPTS

#Define the path of the TWSRegistry.dat file
TWS_REGISTRY_PATH="/etc/TWS";export TWS_REGISTRY_PATH
TWS_REGISTRY_FILE="{TWS_REGISTRY_PATH}/TWSRegistry.dat";export TWS_REGISTRY_FILE

ERR_MSG=" ";export ERR_MSG
RETVAL=0;export RETVAL

#Set initial values
TESTVAL_01="Logon as root          [ ]";export TESTVAL_01
TESTVAL_02="Valid OS              [ ]";export TESTVAL_02
..
..
TESTVAL_19="Encrypt TWSHome/.TWS/useropts_<twsuser> file[ ]";export TESTVAL_19
TESTVAL_20="Exiting $0           [ ]";export TESTVAL_20

```

Non-Customizable Variables

The following variables must not be customized, changes to these variables will cause problems with the script:

The following slide shows a list of variables that exist in the script. These variables should not be customized.

Chg_pass.sh script execution options

The *chg_pass.sh* script has three execution variations. The operations performed by script will depend on whether workstation is a Fault Tolerant Agent (FTA), Domain Manager (DM), Backup Domain Manager (BKM) or Master Domain Manager (MDM).

Chg_pass.sh script syntax variations consist of:

1. TWS user password changes on FTA, DM, BKM or MDM

Syntax:

```
chg_pass.sh -twsuser <twsuser-id> -twspass <password>
```

2. DB password changes on BKM or MDM

Syntax:

```
chg_pass.sh -twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password>
```

3. Both TWS and DB user password changes on BKM or MDM.

Syntax:

```
chg_pass.sh -twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password>
```

Chg_pass.sh script execution options

The *chg_pass.sh* script has three execution variations. The operations performed by script will depend on whether workstation is a Fault Tolerant Agent (FTA), Domain Manager (DM), Backup Domain Manager (BKM) or Master Domain Manager (MDM).

Chg_pass.sh script syntax variations consist of:

1. TWS user password changes on FTA, DM, BKM or MDM

Syntax:

```
chg_pass.sh -twsuser <twsuser-id> -twspass <password>
```

2. DB password changes on BKM or MDM

Syntax:

```
chg_pass.sh -twsuser <twsuser-id> -dbuser <dbuser-id> -dbpass <password>
```

3. Both TWS and DB user password changes on BKM or MDM.

Syntax:

```
chg_pass.sh -twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password>
```

Workstation specific task checklist

Task

Logon as root
 Valid OS
 Valid options
 twsuser <twsuser> is valid
 TWS Instance version 8.3.0 or greater
 TWS Agent type for <cpu> is <agenttype> and valid
 Symphony file exists on this <agenttype>
 Websphere WAS tools exist
 Websphere Active
 Websphere stopped
 TWSHome/.TWS/useropts_<twsuser> exists
 TWSHome/.TWS/useropts_ <twsuser> removed
 WAS Security Properties extracted
 WAS Security Properties parms adjusted
 WAS Security Properties password changed
 WAS Security Properties updated
 Websphere started
 Create TWSHome/.TWS/useropts_ <twsuser> file
 Encrypt TWSHome/.TWS/useropts_ <twsuser> file
 Exiting chg_pass.sh

Workstation

FTA/MDM/BDM
 FTA/MDM/BDM
 FTA/MDM/BDM
 FTA/MDM/BDM
 FTA/MDM/BDM
 FTA/MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 MDM/BDM
 FTA/MDM/BDM
 FTA/MDM/BDM
 FTA/MDM/BDM



Workstation specific task checklist

This slide shows the tasks that will be performed on a workstation by type.

Script operation

Tasks performed by the script must wait for completion of commands executed on either Tivoli Workload Scheduler, WebSphere or DB2 and will take several minutes to complete, be patient.

The script will verify that TWS user exists before proceeding with execution of script.

The values specified with the *chg_pass.sh* script will determine operations and tasks performed by the script.

The script will terminate immediately if specified TWS user is not valid.

The script will generate temporary files in the /tmp directory. The generated files will have a prefix of **ch_** and include the process id/pid as an extension. These generated files will be removed as soon as the script completes.

Script operation

Tasks performed by the script must wait for completion of commands executed on either Tivoli Workload Scheduler, WebSphere or DB2 and will take several minutes to complete, be patient.

The script will verify that TWS user exists before proceeding with execution of script.

The values specified with the *chg_pass.sh* script will determine operations and tasks performed by the script.

The script will terminate immediately if specified TWS user is not valid.

The script will generate temporary files in the /tmp directory. The generated files will have a prefix of **ch_** and include the process id/pid as an extension. These generated files will be removed as soon as the script completes.

Script operation (continued)

The script will use the TWS user to determine workstation type and appropriate procedures to perform.

The script uses information from TWSHome/TWSRegistry.dat file to determine if workstation is an FTA, BKM or MDM, therefore information for the TWS user must be valid and correct.

Domain manager instances are identified as an FTA in the TWSRegistry.dat file.

The script will also attempt to re-validate the TWS user.

Re-validation of the TWS user is required, otherwise MakePlan, adhoc submitted jobs or job streams that require access to scheduling objects in Tivoli Workstation Scheduler database will "ABEND".

The re-validation task in script labeled "Encrypting TWSHome/.TWS/useropts_<twuser_id>" is complete if *chg_pass.sh* script has completed successfully.

Note: If the script "Encrypt.." step fails, required password changes have completed and script may be considered successful. The TWS user will need to be re-validated manually at next UNIX login when invoking conman or composer.

The script will use the TWS user to determine workstation type and appropriate procedures to perform.

The script uses information from TWSHome/TWSRegistry.dat file to determine if workstation is an FTA, BKM or MDM, therefore information for the TWS user must be valid and correct.

Domain manager instances are identified as an FTA in the TWSRegistry.dat file.

The script will also attempt to re-validate the TWS user.

Re-validation of the TWS user is required, otherwise MakePlan, adhoc submitted jobs or job streams that require access to scheduling objects in Tivoli Workstation Scheduler database will "ABEND".

The re-validation task in script labeled "Encrypting TWSHome/.TWS/useropts_<twuser_id>" is complete if *chg_pass.sh* script has completed successfully.

Note: If the script "Encrypt.." step fails, required password changes have completed and script may be considered successful. The TWS user will need to be re-validated manually at next UNIX login when invoking conman or composer.

Script operation (continued)

The *chg_pass.sh* script displays a status of all completed tasks to stdout.

If an error is encountered while performing a task, script will echo error and exit script.

The script will display a task summary showing status for each performed task. The status will either be passed, failed, yes, no, or n/a. The script will perform only necessary tasks, those that are not applicable will be set to "n/a"

Any errors in script will also be displayed again after display of the completed task summary list.

The error message will assist in troubleshooting by identifying last successfully completed task and task that produced error, if script does not complete successfully.

The *chg_pass.sh* script displays a status of all completed tasks to stdout.

If an error is encountered while performing a task, script will echo error and exit script.

The script will display a task summary showing status for each performed task. The status will either be passed, failed, yes, no, or n/a. The script will perform only necessary tasks, those that are not applicable will be set to "n/a"

Any errors in script will also be displayed again after display of the completed task summary list.

The error message will assist in troubleshooting by identifying last successfully completed task and task that produced error, if script does not complete successfully.

Successful Chg_pass.sh Completion Task List

Logon as root	[passed]
Valid OS	[passed]
Valid options	[passed]
twuser m83ora is valid	[passed]
TWS Instance version 8.3.0 or greater	[passed]
TWS Agent type for m83ora is MDM and valid	[passed]
Symphony file exists on this MDM	[passed]
Websphere WAS tools exist	[passed]
Websphere Active	[yes]
Websphere stopped	[passed]
TWSHome/.TWS/useropts_m83ora exists	[no]
TWSHome/.TWS/useropts_m83ora removed	[n/a]
WAS Security Properties extracted	[passed]
WAS Security Properties parms adjusted	[passed]
WAS Security Properties password changed	[passed]
WAS Security Properties updated	[passed]
Websphere started	[passed]
Create TWSHome/.TWS/useropts_m83ora file	[passed]
Encrypt TWSHome/.TWS/useropts_m83ora file	[passed]
Exiting chg_pass.sh	[passed]

14

Change password utility

© 2009 IBM Corporation

Successful Chg_pass.sh Completion Task List

This slide shows all the tasks performed by the script and the status of each task. At this point all the tasks performed successfully, so the script ended successfully.

Unsuccessful chg_pass.sh completion task list

```
Logon as root [passed]
Valid OS [passed]
Valid options [passed]
twuser m821 is valid [passed]
TWS Instance version 8.3.0 or greater [failed]
TWS Agent is valid, [ ]
Symphony file exists on this MDM [ ]
WebSphere WAS tools exist [ ]
WebSphere Active [ ]
WebSphere stopped [ ]
TWSHome/.TWS/useropts_<twuser> exists [ ]
TWSHome/.TWS/useropts_<twuser> removed [ ]
WAS Security Properties extracted [ ]
WAS Security Properties parms adjusted [ ]
WAS Security Properties password changed [ ]
WAS Security Properties updated [ ]
WebSphere started [ ]
Create TWSHome/.TWS/useropts_m821 file [ ]
Encrypt TWSHome/.TWS/useropts_m821 file [ ]
Exiting chg_pass.sh [ ]
```

```
Error: TWS version for TWSuser m821 for TWS instance m821 is 8.2.1
and not valid. TWS version must be 8.3.0 or greater. Unable to
continue, exiting chg_pass.sh.
```

Unsuccessful chg_pass.sh completion task list

This slide shows that the first four tasks completed successfully, or passed. The fifth task, which verifies that the Tivoli Workload Scheduler instance is version 8.3.0 or greater failed.

At the end of the slide it also repeats the error stating that the user for m821 is for Tivoli Workload Scheduler instance 8.2.1 and not valid. The version must be 8.3 or greater.

chg_pass implementation procedures

The `chg_pass.sh` script may be executed from the UNIX prompt as the root user or via Tivoli Workload Scheduler as an adhoc job. It may be defined as a job and scheduled in advance to be used later whenever password changes for TWS or DB user.

To execute script from UNIX prompt:

1. Edit `chg_pass.sh` script and customize variables if necessary.
2. Change permission for script to include execute permissions.
3. Execute the `chg_pass.sh` script as root using options corresponding to password changed:

For TWS password change:

Syntax:

```
-twsuser <twsuser-id> -twspass <password>
```

For DB password change:

Syntax:

```
-twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password>
```

For TWS and DB password change:

Syntax:

```
-twsuser <twsuser-id> -twspass <password> -dbuser <dbuser-id> -dbpass <password>
```

chg_pass implementation procedures

The `chg_pass.sh` script may be executed from the UNIX prompt as the root user or via Tivoli Workload Scheduler as an adhoc job. It may be defined as a job and scheduled in advance to be used later whenever password changes for TWS or DB user.

To execute script from UNIX prompt:

1. Edit `chg_pass.sh` script and customize variables if necessary.
2. Change permission for script to include execute permissions.
3. Execute the `chg_pass.sh` script as root using options corresponding to password changed.

chg_pass implementation procedures

To execute script as a Tivoli Workload Scheduler job:

1. Edit *chg_pass.sh* script and customize variables if necessary.
2. Change permission for script to include execute permissions.
3. Define job that executes the *chg_pass.sh* script. The logon user must be "root".

The path must be provided for *chg_pass.sh* script. The password must be new password for TWS user.

Sample Job definition:

```
M83#CHG_TWS_PASS SCRIPTNAME "/tws/m83/scripts/chg_pass.sh -twsuser m83 -  
twspass passw0rd"  
STREAMLOGON "root"  
DESCRIPTION "Change pass job for TWS user"  
RECOVERY STOP
```

4. Submit the job using Job Scheduling Console or conman. Once the password has been changed increase the job priority to greater than "10" either via Job Scheduling Console or conman.
5. Verify that job completed successfully.

17

Change password utility

© 2009 IBM Corporation

To execute script as a Tivoli Workload Scheduler job:

1. Edit *chg_pass.sh* script and customize variables if necessary.
2. Change permission for script to include execute permissions.
3. Define job that executes the *chg_pass.sh* script. The logon user must be "root".

The path must be provided for *chg_pass.sh* script. The password must be the new password for TWS user.

This slide shows an example of the actual job definition.

4. Submit the job using Job Scheduling Console or conman. Once the password has been changed increase the job priority to greater than "10" either via Job Scheduling Console or conman.
5. Verify that job completed successfully.

chg_pass implementation procedures

To execute script using a Tivoli Workload Scheduler job stream:

- (Optional) Create a job stream that may be scheduled to run on the day password will change. The job stream should have a limit of 0. The scheduled job should have "at" time dependency. The "at" time dependency should be shortly after password will change, if time is known.

The job stream may also be defined as "ON REQUEST" and submitted on the day password will change before time of password change. The job stream should have a limit of "0".

Sample Job stream definition:

```
SCHEDULE M83#CHG_TWS_PASS_SCH
ON REQUEST (or on a specific date)
Limit 0
:
M83#chg_pass AT 0700
END
```

- Submit the job stream using the Job Scheduling Console or conman if not scheduled for a specific date.
- Once password has changed increase job stream limit to greater than "0" either via the Job Scheduling Console or conman
- Verify that job completed successfully.

To execute script using a Tivoli Workload Scheduler job stream, perform the following:

- Create a job stream that may be scheduled to run on the day password will change. The job stream should have a limit of 0. The scheduled job should have "at" time dependency. The "at" time dependency should be shortly after password will change, if time is known.

The job stream may also be defined as "ON REQUEST" and submitted on the day password will change before time of password change. The job stream should have a limit of "0".

This slide shows a sample job stream definition

- Submit the job stream using the Job Scheduling Console or conman if not scheduled for a specific date.
- Once password has changed increase job stream limit to greater than "0" either via the Job Scheduling Console or conman
- Verify that job completed successfully.

Troubleshooting

The script has specific requirements for execution. Review and verify the following:

1. *Chg_pass.sh* script must be executable and executed as root user.
2. A valid TWS user must be specified.
3. The correct options must be specified.
4. The /tmp directory must exist.
5. The /etc/TWSRegistry.dat file must exist and be valid and current for the TWS users.
6. The "thiscpu" variable in the TWSHome/localopts file must have correct Tivoli Workload Scheduler workstation name.
7. Verify that Tivoli Workload Scheduler file system has sufficient disk space.

Troubleshooting

The script has specific requirements for execution. Review and verify the following:

1. *Chg_pass.sh* script must be executable and executed as root user.
2. A valid TWS user must be specified.
3. The correct options must be specified.
4. The /tmp directory must exist.
5. The /etc/TWSRegistry.dat file must exist and be valid and current for the TWS users.
6. The "thiscpu" variable in the TWSHome/localopts file must have correct Tivoli Workload Scheduler workstation name.
7. Verify that Tivoli Workload Scheduler file system has sufficient disk space.

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_chg_pass.ppt

This module is also available in PDF format at: [../chg_pass.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:

DB2 Symphony Tivoli

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>

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

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.

