



WebSphere Product Center Release Notes for *Fix Pack 5.2.1.1*

Note!

Before using this information and the product it supports, read the information in “Notices” at the end of this document.

16 December 2005

This edition of this document applies to WebSphere Product Version, version 5.2.1.1, and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporations 2005. All rights reserved. Licensed Materials-Property of IBM US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Table of Contents

1. Overview	3
1.1 Fix Pack requirements	3
2. Enhancements	3
2.1 Data Entry (Multiple Edit) User Interface Enhancements	3
2.2 Ghost Item Issue	8
2.2.1 Workflow enhancements for ghost items prevention	9
2.2.2 Other enhancements in FixPack 5.2.1.1	10
2.2.3 Best practices for workflow graph design	11
2.2.4 Best practices for workflow step scripts	12
2.2.5 Best practices for workflow system administration	14
2.3 LDAP Integration Updates	16
2.3.1 Limitation	16
2.4 Portal Integration Updates	17
3. Product Fix History	17
3.1 5.2.1.1 Fixes	17
4. Issues and limitations	17
5. Installation	18
5.1 Pre-requisite for 5.2.1.1	18
5.1.1 Change permissions to integration directory	18
5.1.2 Schema migration for ghost items	19
5.2 Migration from 5.2.1 to 5.2.1.1	19
5.2.1 Migration script location	19
5.2.2 Prerequisites	20
5.2.3 Script description	20
5.2.4 LDAP migration	20
5.2.5 Usage	21
5.3 Preparation	21
5.3.1 Stopping the whole application on the local machine	21
5.3.2 Backup	21
5.3.3 Delete Tomcat working directory	21
5.4 Apply Fix Pack	22
5.4.1 Unpack tar file	22
5.4.2 Run WebSphere Application Server script	22
5.4.3 Verify configuration files	23
5.4.4 Run analyze_schema.sh	23
5.5 Test Installation	24
5.5.1 Start WebSphere Product Center	24
5.5.2 Check status	24
6. Appendix A – LDAP Integration Updates	26
6.1 Introduction	26

6.2 Feature Overview	26
6.3 Functional Overview	26
6.4 Assumptions.....	27
6.5 Pre-requisite.....	27
6.6 Configuring Tivoli Directory Server	27
6.6.1 Configure LDAP schema for users and roles	27
6.6.2 SASL - CONFIGURING DIGEST-MD5 on TDS	28
6.6.3 Configuration Notes	29
6.6.4 SSL Setup	29
6.7 Configuring WPC Login Script.....	31
6.7.1 Login Script Parameters.....	31
6.8 Securing Docstore Documents.....	33
6.9 Migration from 521 to 5211	34
6.9.1 Database Migration.....	34
6.9.2 Script Operations	35
6.10 Terms.....	36
7. Appendix B – Portal Integration Updates	38
7.1 Functionality / Features List.....	38
7.2 Multiple Catalog Support*	38
7.2.1 User Context Portlet*	38
7.3 Globalization support*.....	38
7.4 Attribute Types Support*.....	38
7.4.1 2.3.1. Utility Portlets*.....	39
7.5 Register and Approve Portal Users.....	39
7.5.1 User Registration Portlet *.....	39
7.5.2 User Approval Portlet	39
7.6 Browse an Item Catalog*	39
7.7 Work with a single Item	40
7.8 Work with multiple Items.....	40
7.9 Search for items*.....	40
7.10 Work with a collaboration area or business process.....	40
7.11 Upload Items defined in a comma-separated-values or MS-EXCEL file.....	41
7.12 Download Item details, into an MS-EXCEL file.....	41
7.12.1 Track completion status of Asynchronous operations.....	41
7.13 Saving User specific portlet settings*.....	42
7.14 Limitations.....	42
7.14.1 Catalog Support.....	42
7.14.2 Rendering of Item Details	42
7.14.3 Item Validation	42
7.14.4 Business Process Support.....	42
7.14.5 Transactions tracking.....	42

1. Overview

This document addresses enhancements, product fixes, installation instructions, and other updated information for WebSphere Product Center Fix Pack 5.2.1.1.

The following features were the main drivers for the delivery of this fix pack:

- Data entry UI enhancements
- Ghost item issues
- LDAP (Light Weight Directory Protocol) integration updates
- Portal integration updates

1.1 Fix Pack requirements

WebSphere Product Center 5.2.1 must be installed prior to the application of Fix Pack 5.2.1.1.

2. Enhancements

2.1 Data Entry (Multiple Edit) User Interface Enhancements

Several updates have been made to the user interface, specifically in the area related to performing multiple edits that greatly enhances a users experience when performing multiple edits to an item. This section details each of these updates.

Consistent Warning Messages for Lost Work

To prevent the loss of work during actions that would cause a user to lose unsaved work, additional warnings have been implemented to prompt users to save their work. For this Fix Pack, warnings are provided for the following actions:

1. Switching out of the current context. For example, by selecting an action from a pull-down menu or from a popup menu in the Left Nav Pane.
2. Changing from Single Edit page or Multiple Edit page to another page (For example, the Rich Search page).
3. Scrolling to another page of entries.
4. Signing out or exiting from the current WPC session.

Date Picker

The date and time picker allows the user to select a date and time from a pop-up style calendar. When a field has been defined as a date entry field, the user can click on the calendar icon to access the pop-up calendar, then select a month, date, year, and time.

Date Formatting

The option to specify an input or output format for date and time is now available in user settings. The default date/time format is "M/d/yy h:mm a". Users can specify a date/time format from a drop-down selection or create a user specific format.

Disable Auto Deselect on Copy Down

All entry selections are preserved when a copy down is made from the column headers.

Disable Auto Deselect on Save

All entry selections made are preserved when a save is performed.

Display Categorization in Multiple Edit

On the Single Edit page, the categorization is displayed to the users. However, such categorization information is not shown for the entries on the Multiple Edit page. Therefore, the addition of the same information to the entries is displayed on the Multiple Edit page.

Two icons have been added for each of the following type of entries:

- 1) The corresponding entry does not currently have any hierarchy mappings defined
- 2) The corresponding entry does have hierarchy mappings defined

When the mouse pointer hovers over the mapped hierarchy icon, a tool tip displays the hierarchy mappings information. The tool tip shows up to five mappings and indicates how many more are not displayed. The hierarchy mappings information shown in the tool tip also reflects the pending changes that have not been saved yet.

In addition to the tool tip, a popup is used to show the categorization information when the user clicks on the mapped hierarchy icon.

For this feature, a new user setting option was added " In Multi Edit, show hierarchy mappings for every entry displayed". This allows the users to control if the hierarchy mappings information should be collected and displayed in the Multiple Edit page. By default this option is false (i.e., the hierarchy mappings are not shown in Multi Edit).

Display Type of Node

A new user setting option was added, "Display the type of node in the Single Edit and Multiple Edit screens", which controls whether the icons signaling a node's type (dollar note, picture of flag, etc.) are displayed in the Multiple Edit column headers.

Goto First/Last Attribute Page

On the Multiple Edit page, a new set of UI controls allow the navigation of attributes to be displayed in either forward or backward manner. Currently, only "page backward" and "page

forward” are supported. Therefore, the addition of two controls allow the user to “jump to the beginning” or “jump to the end” to quickly move to the first or last “page” of attributes respectively.

Highlight Error Attribute Cell

Previously, cells where validation errors have occurred were highlighted in red only in single edit. The highlighting an error is now displayed in Multiple Edit.

Link Attribute Picker in Rich Search, Single Edit and Multi Edit

Currently, there is only an entry field for link attributes, which exposes users to the risk of an incorrect entry value due to typos. To address this issue, an attribute picker has been added for link attributes in various places in the product as described in more details below.

- **Rich Search screen** - The view and browse buttons are added to the link attributes. If the corresponding entry field has a value and the view button is selected, WPC displays the details on the item, providing the item exists in the destination catalog configured for the link attribute. The configuration of a link attribute is available through the Catalog Attributes function

The Browse allows the user to browse the destination catalog and pick an item in the destination catalog to be linked with the link attribute in the current catalog. The value of the selected item is used to set the value in the corresponding text field

- **Single Edit screen** - In a very similar fashion, as the Rich Search screen, the view button and browse button are added to the link attributes in the Single Edit screen
- **Multiple Edit screen** - A picker (entry field and the browse button) in the column header displays if the attribute is a link attribute

Lookup Table Popup

A lookup table editor (as found in the single edit screen) is available on each Multiple Edit cell with a column type Lookup Table. The editor is only available if the user “clicks” the cell. The editor is also available in the column header, so that it can be used to set multiple cell values at once.

Maintain Screen Location After Refresh/Save/Fill Down

The horizontal and vertical scrollbar remain at the same position after a refresh, save or fill down action.

Max Rows per Page in Multi Edit Setting

When processing items in a batch, the default maximum number of rows displayed in the Multiple Edit page is 20. In addition, in User Settings, the maximum value that can be selected from the “Rows per Page in Multi Edit” dropdown selection is 20 even though the maximum allowed as defined in common.properties is 50. Therefore, the user settings and the behavior of Multiple Edit with respect to “Rows per Page in Multi Edit” have been updated as follows:

1. User setting dropdown displays 2, 3, 4 and selections in multiples of 5 starting from 5, up to maximum of 50.
2. If the user settings have never been saved for a given company, the "table_rows_per_page_multi_edit" value in common.properties is used.
3. The number of rows specified in common.properties must be a positive multiple of 5 or be rounded up to the next multiple of 5 but not to exceed 50. The value specified in common.properties must be less than or equal to 50.
4. Once a selection is made in the user settings and the value specified for the "Rows per page in Multi Edit" is saved for a given company, it is this value that is used in Multiple Edit page until the user modifies its value in the user settings.

Navigation to Error Attributes

An improvement was made in the way validation errors are being displayed in Multiple Edit. To address the scenarios that an error icon does not display when a save occurs asynchronously, there is now an icon to indicate the entry "is being processed". The user must refresh the page to obtain the result of the save.

The following are the behaviors of error attributes in the Multiple Edit screen:

- The icon tool tip no longer contains the list of validation errors, but rather a simple text "Errors during save; click for details". Clicking on the icon brings up a popup that lists all the validation errors for this entry. Validation errors contain a link to the cell that contains the problematic value. The link sets focus to the particular cell
- In case the problematic cell is in a tab and the tab is not visible, clicking on the link will bring the tab to the foreground and focus on the problematic cell
- The background color of the problematic cells changes to red

To keep the Multiple Edit screen consistent with the Single Edit screen:

- When there are validation errors, and the selected view has tabs, there is no longer a clickable "error icon" between the tabs and the magnifier icon
- The state icon (upper left corner) is clickable and brings up the popup that the "error icon" used to display
- When there is more than one error, the tool tip of the state icon reads, "There are validation errors for this entry. Please click on the icon for a detailed list of errors."

Show All Validation Errors for an Item

Errors are not shown in the flashing Multiple Edit error icon tool tip at the row level. Instead, the tool tip now contains the following: "There are validation errors for this entry. Please click on the icon for a detailed list of errors."

Show Display Attribute in Left Column Grid

Note: Refer to Section 4 for limitations to this feature and a workaround.

In situations where there are too many attributes, a user may be working on a page of attributes without an easy handy reference to the display attribute (usually the primary key) of an item. A new feature has been added to allow the freezing of the display attribute while scrolling across to view other attributes. The behavior of this feature is as follows:

- 1) The addition of the option " In Multi Edit, show the display attribute in the left column grid" to the User Settings allows a user to control if the display attribute is displayed in the left column grid (i.e., the left most column in the Multi Edit page).
- 2) Based on the current user setting, the display attribute as configured for the catalog is displayed in the leftmost column in the Multi Edit page.

Available options:

- No (does not display attribute when scrolling across multiple attributes)
- Yes and truncate from left if too long
- Yes and truncate from right if too long

- 3) The display attribute of an entry also functions as a hyperlink. Clicking on it displays the corresponding entry in the Single Edit page.

Special Characters Support

Special characters such as single or double quotes in Multiple Edit are now supported for string enumeration and lookup table fields. Previously, these special characters were not supported properly and upon saving, an error would occur. Consequently, users had to switch to the Single Edit page to select a value from a string enumeration or lookup table, which includes any of these special characters.

Rendering of the lookup table window fails if the selected option from the select contains the special character "#". There can be other situations where the rendering of the lookup table picker window may fail. However, this occurs only under rare circumstances. It is advisable to avoid using # in the lookup table key string.

String/Number Enumeration dropdown

The string/number enumeration drop-down is available in the column header, so that it can be used to set multiple cell values at once. This is provided for number and string enumeration types only (i.e. this will not be available for string enumeration rule types).

Tabs in Multiple Edit

Multiple Edit can now display tabs. The pagination controls are still displayed when there are more than twelve attributes in the tab and attributes can be located in multiple tabs. Similarly to Single Edit, there is a magnify glass icon that helps the user to search a given attribute among the tabs. The search result provides a link back to the appropriate tab and sets the focus in the cell in the first row of the selected attribute.

Tab setup for the Multiple Edit screen is done in the same way as is done for the Single Edit screen. In the “Bulk Edit” section of the view definition screen there is a new button “Tab View” to bring up the tab definition screen.

Note: There are no changes in scripting operations or in the Environment Export Import package to support this feature.

Tab Ordering in Top Row

- Instead of cycling through the arrow icons (resize, sort) and then the input boxes separately for each column header (top row), now users can first cycle through all the arrow icons for each header, then the input boxes and associated icons (editor buttons, apply down arrows)
- The apply down icons have been moved to the right of input boxes rather than the left. This makes it more intuitive to enter a value in an input box then tab along to the apply down icon (and activate it by the keyboard) rather than potentially input a value then tab back to the apply down icon
- Previously, scrolling the main pane forces the column headers (top row) to also scroll, keeping them in sync. But scrolling the column headers (via tabbing) does not scroll the main value display at all. Now, scrolling the column headers will also scroll the main value display.

2.2 Ghost Item Issue

In FixPack 5.2.1.1, enhancements have been made to WPC workflow to drastically reduce the occurrence of the collaboration area data corruptions known as *ghost items*. Also, new best practices for workflow graph design, the writing of workflow step scripts, and the workflow-specific aspects of WPC system administration, have been formulated and are provided as part of these release notes. Section 2 below provides a brief description of the code changes, intended to impart only what workflow designers, step script writers, and users need to know about them. Section 3 contains recommended best practices for workflow graph design. Section 4 is best practices for step scripts. Section 5 is best practices for system administration.

These release notes will refer exclusively to ghost items, although for category (hierarchy) workflows there is the analogous issue of *ghost categories*. *Ghost categories has not been addressed in this FixPack*. The material here applies equally to both. Recall that in WPC an *entry* is either an item or a category; below we will sometimes say *entry*, sometimes *item*. Most material applies to entries, i.e. to both items and categories. When something applies only to one or the other, it should be clear from the context. Generally when we talk about categories, we are talking about something that

applies only to categories. When we say *container*, we mean “catalog” for the case of items, and “category tree”, i.e. “hierarchy”, for the case of categories.

2.2.1 Workflow enhancements for ghost items prevention

We will simply list the key changes that have been made here. They may seem to have little connection with each other. In fact, at the technical level, they are closely related.

2.2.1.1 Workflow processing transactions

First, in 5.2.1.1 the design and implementation of workflow events has been changed so that both endpoints of workflow graph moves (transitions from one step to another), and all the database updating associated with entering (checkout) and leaving (checkin, drop) a collaboration area, and “adds” and “imports” into workflow, are committed in a single database transaction. Formerly, leaving one step, and entering its successor step, were two separate transactions. This is the major step in 5.2.1.1 to address ghost items.

2.2.1.2 Connection isolation

Second, the step scripts are now run in separate database connections (and transactions) from the connection/transactions used by the workflowengine for graph moving. Thus the success or failure of an invocation of an IN(), OUT(), or TIMEOUT() function has no bearing on what step an item is in or will be.

2.2.1.3 Disable synchronous workflow processing

Third, previous releases of WPC have supported some scenarios of synchronous processing of workflow events in non-workflowengine WPC modules (e.g. appsvr, scheduler). For example, calling the checkoutEntries(...) script operation with its waitForStatus parameter = true is such a scenario in which synchronous execution of a workflow event is requested. Formerly, the workflow event spawned by such a script invocation was executed “inline” as a function call, rather than posted to the WPC workflow event queue and processed asynchronously by the workflowengine. This had brought about two flavors of workflow events inside WPC, those executed synchronously and those executed asynchronously. This situation is believed to have been a key cause of ghost items.

In 5.2.1.1 there is only the single flavor of asynchronous workflow events, all posted to the workflow event queue and processed asynchronously by the workflowengine. Scenarios such as checkoutEntries of requested synchronous behavior are supported by a new “busy waiting” capability of the WPC GUI, and of WPC script ops like checkoutEntries, to poll the workflow event queue until the event they are waiting for has finished processing.

2.2.1.4 One workflow engine

To reduce the possibility of conflicts on the database from workflow processing that might create ghost items, we now require that only one workflow engine is running in WPC.

If you try and run with multiple workflow engines you will get a critical error and will not be able to proceed with workflow processing.

2.2.1.5 General comments on ghost item prevention

As always, the workflow designer and the workflow user must be prepared that the processing of any workflow event may fail. When that happens, collaboration area entries will unexpectedly

appear to be “stuck” in certain workflow steps. A qualified step performer (i.e. a WPC user who is defined to be a performer in the relevant workflow step) can try to manually move the item out of the step by clicking one of the DONE buttons. This will almost always succeed in moving the item forward. However if it does not, the only recourse in WPC 5.2.1.1 is for a collaboration area administrator to send the items to the Fixit step, and then either check them in or remove them from the collaboration area.

We do not claim that ghost items cannot occur with this FixPack 5.2.1.1. In fact, on the contrary, we mention some ways below in which ghost items can be intentionally created (by violating certain of the best practices). Thus, in general, you may need to run the ghost items detect and clean scripts, which are separately distributed by WPC Level 2 Support. They are not provided by default in the Installer. Some documentation of these scripts is provided below in Section 5.2

Important note: As a result of the changes to address ghost item issues which moved all step transition logic to the end of the prior step, it is now possible to see an entry in the next step in the user interface before its IN() script has finished running.

Check for IN() script completion before you begin working with an item in a non-initial workflow step

2.2.2 Other enhancements in FixPack 5.2.1.1

2.2.2.1 Merge step processing

As part of the rework of the workflow step transition processing we took the opportunity to improve the way WPC handles merge steps in complex workflows.

A merge step is now able to keep track of how many times a particular item has entered that step, even beyond the amount that should trigger an exit. This allows for better recovery for example if an item enters the merge step but the exit is not triggered due to something like a system failure.

The other new feature is that the collaboration area administrator can now click on a merge step in the collaboration console and then click “Done” or “Fixit” to cause the entry to exit from the step.

Here are two examples of why this might be necessary:

Recall that for a merge step with n incoming steps, the item will exit the merge step once per n entrances of that item.

Admin triggers premature exit from Merge step

Say we have three routes into a merge step, A, B, and C. Item X has entered the merge step along routes B and C but not along A; there is some reason why getting the item to traverse path A is very difficult. In this scenario the collaboration area admin clicks on the merge step, and clicks DONE. The item will be removed from the merge step and inserted into the next step

Admin triggers exit from merge that should have happened but did not

Imagine we have a workflow with a loop; entries can pass around the workflow and eventually re-enter the merge step. Imagine that we have a merge step with 3 steps entering it, and we have had a

particular entry arrive in the merge step 8 times. You would normally expect that 8 entrances would equate to $3+3+2 = 2$ exits from the merge step, and that one more entrance will trigger a third exit.

Let us say that some problem with the system prevented these exits from happening. In the new code at 5.2.1.1, the system does not “lose” these entrances once they have occurred, as a counter is stored in the database and will store the value of 8 in this case.

To rectify this situation, the collaboration area admin goes into the merge step and clicks Done once. This will reduce the internal counter to 5 and trigger an exit from merge; however because the counter is still greater than zero, the entry will still be in the merge step even though a copy of it has left.

The admin can then click Done again, this will trigger a second exit from the merge, reducing the counter to 2, and again leaving the entry in the merge step.

If the admin were to click Done a third time, the entry would leave the merge step and be deleted from the step.

Warning: We recommend that this ability to move items manually out of a merge step is used with great caution; only in cases where system failure has prevented normal workflow processing and where it is not possible to go back and correct the problem in the earlier step that is preventing merge (e.g. in example 1 above, it would be better to go back and fix the problem on route A)

2.2.2.2 Nested workflows

We also took the opportunity to correct an issue with nested workflows that meant that upon exit from an inner workflow, any copies of the item in outer workflows would be deleted.

The policy now is that only copies of the item within the inner workflow (or workflows within the inner workflow) will be deleted when exiting that inner workflow.

2.2.3 Best practices for workflow graph design

WPC supports the creation of a rich set of workflow graphs, both through the GUI and through scripts. For example, any exit value of a step may lead to multiple next steps (a “split”), loops are permitted, and a single step of one workflow can contain an entire nested workflow.

2.2.3.1 Avoid splits leading to loops.

Recall that a *split* coming out of a step S is an exit value for S for which there are multiple next steps, i.e. multiple “outgoing branches”. Recall also that a *loop based at S* is a sequence of steps, starting at a step S, for which each successive pair $S_i \rightarrow S_j$ is connected by some exit value of S_i that leads to S_j , and where the last S_j in the sequence is just S again. The workflow graph designer should avoid having a split where one or more of the outgoing branches is in fact the beginning of a loop based at S.

As an example of this construction, we might have S going to S1, S2 for some exit value, and then S1 goes back to S for some exit value. Or in the most extreme form, we could have S going both to S2 and to itself, for some exit value (self loops are legal). Such a construct of “split + loop”, while legal, almost always leads to very confusing behavior for the users, for the reason that it enables the

total count of items in the workflow, i.e. the total count of items displayed in the collaboration console, to steadily increase, for no apparent reason.

Furthermore, suppose we have again the case of S going to S2 and itself, if S2 is a non-automated step (e.g. Modify, General), then the item may still be sitting S2, when a second exit from S sends it again to S2, i.e. we have a “repeated entrance” to S2. Such repeated entrances are also highly confusing to users, because they do not increase the item count in the re-entered step. In fact in 5.2.1.1 FixPack, an error message will be printed in the workflowengine’s workflow.log whenever a repeated entrance occurs.

2.2.3.2 Merge steps should not loop back to themselves.

This may produce unexpected behavior in complex workflows. If the graph designer believes it is needed, at least some additional intermediate steps should be inserted so that the loop is not directly from a Merge step to itself.

2.2.4 Best practices for workflow step scripts

In principle, any WPC script operation can be used in the IN(), OUT(), or TIMEOUT() functions of a workflow step script. However, the expressive power of the (large set of) WPC script operations, combined with the richness of possible WPC workflow graphs, makes it possible to create workflows and step scripts that will have deadlocks and similar problems. The workflow graph designer and the step script writer must coordinate and review their combined activities to avoid such pitfalls.

2.2.4.1 Minimize accessing of other containers in step scripts

It can be useful and powerful for a workflow step script to access other containers (e.g. lookup tables, or catalogs and category trees unrelated to this source catalog or collaboration area), and the items or categories they contain. However the risks of doing so cannot be overlooked. Through splits in the workflow graph, for example, there can be a high degree of concurrency of workflow event processing, which of course means that multiple step scripts can be running concurrently. If several of them are accessing the same container or its elements, there is the potential for deadlock, and it will not be automatically detected.

This (accessing other containers and their entries) is an area where the workflow graph designer and the step script writer must work carefully together, to keep track of the big picture represented by the whole suite of a workflow’s step scripts. They must at all times assure that, with respect to this suite of step scripts of all workflow steps, there is no pair of them such that deadlock can occur if both are running at the same time.

2.2.4.2 Be aware of the other step scripts

This is an expansion of the previous point. In workflow step scripts more than anywhere else in WPC, it is necessary to take into account from the outset, the possibility that other step scripts will run before, concurrently, or after this one. Also, that any particular step script may run more than once: this might happen if the workflowengine is stopped and then restarted.

Thus it is dangerous to assume without checking, for example, that an OUT script can simply insert a particular new primary key into a lookup table and commit it. The workflowengine might have been stopped while this script was running, but after it had done its commit. Thus the item in step

S whose OUT script this was would still be in S when the workflowengine is restarted, and then a user might click a “done” button in S again, and the OUT script would run again.

2.2.4.3 Be aware that the state of the collaboration area and its entries may change while the script is running

A step script runs in its own database transaction, which is independent of the database transactions used by the workflowengine to carry out actions on the workflow graph (and independent of the database transactions of other step scripts). The step script can do commits on its database transaction. The workflowengine neither knows about them nor can intervene in them. In particular, the workflowengine cannot roll back a commit done by a step script. And of course, commits done by a step script have no bearing on the behavior of the workflowengine or other step scripts (unless some of these guidelines are violated, and e.g. a step script unilaterally deletes an item from the collaboration area’s catalog).

A concrete instance of this issue is the case where one or more of the entries in the step script’s entrySet is dropped or checked in from the collaboration area, while the step script is running. The running step script will not know about this. Thus the workflowengine may drop an item from the collaboration area’s container, and a moment later the step script might do a saveCtgItem() on it, creating a ghost item.

2.2.4.4 Steps scripts must manage their own database transactions

Step scripts run in the same context as the workflow event the step uses, so they share the database connection with workflow event at the step. However, step scripts can perform their tasks in their own transactions over the same database connection. If there is any transaction still open after execution of the scripts, workflow event process will commit it and re-open a new transaction. In general, step scripts should use the useTransaction() script operation to establish their transaction boundaries.

2.2.4.5 Avoid self-referential workflow script operations in entry build scripts

An important class of WPC scripts which must necessarily be included in the discussion of workflow step scripts are *entry build scripts*, e.g. pre-processing scripts, post-processing scripts, and post-save scripts. We must take account of them because both workflow checkout, and workflow checkin, events will cause such scripts to be run, for the entries being checked out or in. For example, when an item is checked out, a clone of it is created in the collaboration area’s catalog, and then saved, during which entry build scripts will run. It should be clear that self-referential workflow script ops running in such an entry build script could cause deadlock. For example, if an entry build script were to call CollaborationArea::dropEntries() on itself, then it would be trying to drop itself from the collaboration area before it had been fully saved there.

2.2.4.6 Use CollaborationArea::addEntryIntoColArea() to add entries to a collaboration area’s container

Do not attempt to get new items into a collaboration area by the following means: “item = new CtgItem(ctg); item.saveCtgItem();” where “ctg” is the collaboration area’s container. Or “category = new Category(ctr); ctr.saveCategoryTree();” for a hierarchy workflow where “ctr” is the collaboration area’s container. Instead, use CollaborationArea::addEntryIntoColArea().

Using either of these deprecated usages will, in fact, create a ghost item because the item will be added to the collaboration area container, but not to the database tables, which maintain the state of the workflow graph (and which drive the collaboration console).

2.2.4.7 Do not use `Category::deleteCategory()` on a collaboration area's category tree

Category workflows have very specific rules for building and maintaining the “partial clone” of the source category tree, that is the collaboration area's container. It is appropriate to use `CollaborationArea::dropEntries()` to remove categories from the workflow, but do not attempt more direct manipulation in a step script, such as by `Category::deleteCategory()`.

In an item workflow as well, it is a bad idea to use `deleteCategory()`.

2.2.4.8 Use `CollaborationArea::moveEntriesToColArea` with care

This script operation is inherently complex and lengthy. Nevertheless if used intelligently it is a valuable tool. You will need to handle the possibility of its failing, for example if the entries being moved cannot be checked out, for whatever reason, to the requested destination collaboration area.

2.2.4.9 Manipulate collaboration area container processing options in step scripts with care.

Again, bear in mind that multiple steps scripts may be running simultaneously. If they each manipulate collaboration area container processing options of the same container (e.g. the collaboration area container), then great care is needed by the script writer.

2.2.5 Best practices for workflow system administration

2.2.5.1 General engine-level usage guidelines

- You should only run one workflow engine, against any database. Its number of threads is at your discretion, subject to constraints you can find in the WPC Installation Guide and also in comments in `common.properties`.
- If the workflowengine crashes unexpectedly, or you have reason to suspect that an explicit shutdown did not go smoothly, run the ghost items detect and clean scripts before attempting to restart it.
- Use the `stop_workflowengine.sh` shell script to gracefully shut down the workflowengine. This will allow all currently processing workflow events to complete, then stops all threads.
- If the workflowengine is hung, i.e. in a deadlock, when you run `stop_workflowengine.sh`, then you will need to use `abort_workflowengine.sh`. You should then run the ghost items detect and clean scripts before attempting to restart the engine.

2.2.5.2 Ghost items detection and cleanup

Before running any of the scripts, first do a graceful shutdown of the workflow engine using `stop_workflowengine.sh`. The workflow engine should be stopped when you run any of the ghost items detect/clean/restore scripts. In fact, it is best to shut down the entire WPC application, however as long as no edits are made to items and categories when running the ghost items scripts, it should be sufficient to only shut down the workflowengine.

Next, be sure you have created the “backup tables” in the database schema. These are special tables used by the cleanup and restore scripts. Use the `drop_backup_tables.sh` and

create_backup_tables.sh SQL scripts to create these tables. Note that you only have to run the create_backup_tables.sh script once; thereafter, you can freely run the report/clean/restore scripts as often as you wish, and the backup tables will be automatically (re)used.

Next is to download the compressed file containing the scripts and uncompress them into the \$TOP/src/db/workflow directory.

FTP location: <ftp://ftp.software.ibm.com/software/websphere/wpc/fixpacks/5.2.1.1/ghostitems.zip>

Uncompress to: \$TOP/src/db/workflow, the script files can be found in \$TOP/src/db/workflow/ghostitems/oracle

The following describes the Oracle versions of these scripts; the DB2 versions have quite similar names.

There are three shell scripts that make it possible to run each of the three functions from the UNIX command line:

- doreport.sh, doreportall.sh
- doclean.sh, docleanall.sh
- dorestore.sh, dorestoreall.sh

Instructions for their use are contained in the header comments of the scripts themselves.

It is equally feasible to run the SQL scripts and PL/SQL procedures directly from SQL*PLUS or equivalent. The relevant scripts are:

- Drop_views.sql
- Create_views.sql
- Procs.sql
- Report.sql
- Clean.sql
- Restore.sql

Instructions for their use are contained in the header comments of the scripts themselves. Or you can read the above shell scripts to see how they make use of the SQL scripts.

One general note to bear in mind about ghost items cleanup, it was said above that the workflowengine should be shut down when running these scripts, and in some cases, the shutdown might not have been graceful, e.g. a power failure. Thus in general we cannot know the exact status of WPC scripts that were being run by the workflowengine, such as workflow step scripts, when the shutdown occurred. In particular, such scripts may have done "intermediate" database commits., If the items associated with such scripts are found to be ghost items, and removed from the

collaboration area by the cleanup script, i.e. their checkouts are undone, there will still be no general way for all intermediate commits to be undone. In this sense the cleanup script is not able to truly rollback partially completed work.

2.2.5.3 Run analyze_schema.sh

It is good practice to run analyze_schema.sh on a regular basis but if problems occur with workflow functionality, run analyze_schema.sh to see if it resolves the issue.

Location: \$TOP/src/db/schema/util/ analyze_schema.sh

```
cd $TOP/src/db/schema/util
./analyze_schema.sh
```

To run analyze schema in DB2, refer to the WPC technote: Analyzing schema and collecting statistics in DB2 - <http://www-1.ibm.com/support/docview.wss?uid=swg21205939>

To run analyze schema in Oracle, refer to the WPC technote: Analyzing WebSphere Product Center schema and collecting statistics in Oracle - <http://www-1.ibm.com/support/docview.wss?uid=swg21206376>

To run analyze schema in Oracle, refer to the WPC technote: Analyzing WebSphere Product Center schema and collecting statistics in Oracle - <http://www-1.ibm.com/support/docview.wss?uid=swg21206376>

2.3 LDAP Integration Updates

NOTE: Currently LDAP is certified with IBM Tivoli Server version 5.2. For LDAP integration details, refer to Section 6 of this documentation.

LDAP integration methods have been updated to provide a tooling layer for managing User Authentication within WPC, primarily using scripts for Login and Logout processing. Note that the ldap_config.xml will no longer be used for LDAP configuration and similarly the enable_ldap attribute in common.properties is also not applicable. All these parameters should now be configured from the Login Script.

The LDAP update provides a flexible mechanism that is independent of a customer's LDAP directory setup and User Directory Structure. This maintenance release focuses on an implementation that builds the authentication framework based around both LDAP and WPC's own internal User Registry as Authentication mechanisms.

2.3.1 Limitation

For LDAP Users in WPC, the user details displayed do not include any personal details (First Name, Last Name, email etc.) because currently this information is not fetched from LDAP. This limitation will be addressed in a future release.

2.4 Portal Integration Updates

Several significant additions and updates to the existing Portal Enablement functionality have been implemented in this release. These functions were built on top of the WPC Portal Enablement 5.2.1 and are described in Section 7 of this document.

Refer to the updated Portal Enablement documents that have been provided with WebSphere Product Center 5.2.1.1.

3. Product Fix History

3.1 5.2.1.1 Fixes

Tracking Number	Description
JR22520	Organization Deletion Causes Organization Hierarchy Data Corruption
19738	Issue when changing WSDL in portal - Unacceptable Character ">" throws Exception.
JR21767	Scripts - getCategorySet() returns set with wrong size
JR21520	Uniqueness validation fails for multi-occurring attributes
JR22566	WPC produces incorrect XML for Commerce integration
JR22571	Custom Tool Does Not Render As Home Page
JR21932	WebServices: request fails validation when using xsd namespace
JR22020	FETCHING AN EXTERNAL URL WHEN INVOKING A WEB SERVICE NOT WORK
JR22209	Hierarchy pre processing script and the post processing script
JR21740	running delta syndications causes 'Unexpected Version' in syn
JR21996	REALTIME SEARCH FAILS WHEN THE SEARCH COUNT IS MORE THAN 100 (To be done by 5.2.0-IF006 integration)
JR21985	isStringSingleByte() script operation returns true for a string
JR22517	Can not view the files named with localized language letters
JR21996	Realtime search fails when the search count is more than 100

4. Issues and limitations

This maintenance release contains the following known issues (which will be addressed in a future maintenance release):

Tracking Number	Description
21016	<p>If display name is not contained in the current view of Multiple Edit page, it cannot be displayed in left column grid</p> <p>Problem Description: The primary key instead of the value of the display attribute is shown in the first grid column on the Multiple Edit page if the current view (be it the system default view or a</p>

	<p>user-defined view) does not include the display attribute.</p> <p>Workaround: In order for the value of the display attribute to be shown in the first grid column in the Multiple Edit page, at least one of the following conditions is required.</p> <ol style="list-style-type: none"> 1) Include the display attribute in the view being used 2) Include the display attribute in the User Defined Core Attribute Collection (accessible through Hierarchy Attributes for hierarchies or Catalog Attributes for catalogs). <p>If the display attribute is the primary key, the aforementioned problem will not happen.</p>
21047	<p>Limitation related to the Show Display Attribute in Left Column Grid feature. If the collaboration area has been instantiated before the user changes the display attribute, the previous display attribute will still be used when the item is displayed in the multi-edit page under the workflow context. This is because the information has been cached. The workaround is to restart the WPC session.</p>
21243	<p>Edit page didn't appear after clicking on "Reserve & Edit" button in workflow. Note: Workaround is provided in the Portal Enablement Installation Guide Provided with this release.</p>
21267	<p>Rendering of lookup table window fails if the selected option from the select contains the special character "#". There can be other situations where the rendering of the lookup table picker window may fail. However, this occurs only under rare circumstances. It is advisable to avoid using # in the lookup table key string.</p>
21139	<p>DB2 query size problem. If running large queries that return no records, it is necessary to adjust the STMTHEAP size for DB2.</p> <p>The following DB2 commands need to be run on the DB2 server as the DB2 instance owner to adjust the STMTHEAP size to an appropriate values to handle larger queries:</p> <pre style="margin-left: 40px;">db2 connect to <dbname> db2 update db cfg for <dbname> using STMTHEAP 10000</pre>

5. Installation

This section provides general guidelines to apply a maintenance release to WebSphere Product Center. Some information may differ depending on the methods used for previous installations. Contact your support representative for WebSphere Product Center with any installation issues.

5.1 Pre-requisite for 5.2.1.1

NOTE: Only applicable to WebSphere Product Center 5.2.1.1

5.1.1 Change permissions to integration directory

NOTE: This must be done prior to applying maintenance release.

On the system where the 5.2.1.1 Maintenance Release is to be installed, perform the following operation before applying the 5.2.1.1 Maintenance Release:

```
chmod -R +w $TOP/integration
```

5.1.2 Schema migration for ghost items

NOTE: This must be done after applying maintenance release.

To address the issue of ghost items, the following schema migration is required:

Oracle

Run the following commands at the command prompt on the application server

1. Connect to the database. "dbusername", "dbpassword", "dbname" can be found in the WPC common.properties file.

```
sqlplus dbusername/dbpassword@dbname
```

2. Run the following command at SQL prompt

```
ALTER TABLE TUTL_JHB_JVM_HEARTBEAT ADD JHB_JVM_TYPE varchar2(200);
```

DB2

Run the following commands at the command prompt on the application server:

1. Connect to the database. "dbusername", "dbpassword", "dbname" can be found in WPC common.properties file

2. Run the following sql command

```
DB2 "ALTER TABLE TUTL_JHB_JVM_HEARTBEAT ADD COLUMN JHB_JVM_TYPE  
VARCHAR (200)"
```

5.2 Migration from 5.2.1 to 5.2.1.1

This section describes the pointers and pre-requisites of the 5.1.x -> 5.2.1.1 Migration enablement, which uses a SQL strategy. The script called **migrateto5211.sh** is used to migrate the database schema used by WPC prior to 5.2.1.1 to WPC 5.2.1.1.

5.2.1 Migration script location

The script is located in the \$TOP/bin folder.

Note: This script will cause permanent changes to the database schema use by WPC. This change is irreversible. We urge you take a backup of the database schema before you continue.

5.2.2 Prerequisites

This script causes permanent changes to the database schema. Hence since the change is irreversible, taking a backup of the database being migrated is recommended. Running the script sends out the following prompt for user action to ensure the above:

Do you want to continue? (y/n)

And asks one more confirmation:

Have you taken a backup of the database schema (y/n)?

Note: If the user clicks on N on either question, the script would exit since the user has chosen not to proceed.

5.2.3 Script description

The script does the following:

- Adds 2 new columns to the TSEC_SCU_USER table : SCU_LDAP_ENTRY_DN, SCU_LDAP_SVR_URL
- Adds a new column to the TUTL_JHB_JVM_HEARTBEAT table : JHB_JVM_TYPE
- Populates default values to the 2 new columns in the TSEC_SCU_USER table
- LDAP users migration

5.2.4 LDAP migration

For this LDAP migration, 3 arguments need to be provided to the script.

- ldap_url : the url of the LDAP server
- name_attribute : the naming attribute
- parent_dn : the parent DN

The script then does 2 things :

a) Updates the SCU_LDAP_ENTRY_DN column with a value computed as follows wherever the SCU_LDAP_FLAG column value is set to 'Y' :

```
<name_attribute>=<SCU_USERNAME>,<parent_dn>
```

where name_attribute and parent_dn values are supplied as arguments to the script and SCU_USERNAME is the username column in the TSEC_SCU_USER table

b) updates the SCU_LDAP_SVR_URL column with the ldap_url

5.2.5 Usage

- a) Just execute the `migrateto5211.sh` script from the command line
- b) For LDAP users migration, execute the script with the arguments as mentioned in the previous section in the following manner:

```
migrateto5211.sh ldap_url name_attribute parent_dn
```

5.3 Preparation

Before attempting to apply the latest Maintenance Release to WebSphere Product Center, the following preparation is recommended:

5.3.1 Stopping the whole application on the local machine

Complete the following steps to stop the WebSphere Product Center instance:

1. Check the scheduler to make sure there are no critical jobs that need to be completed. If the queue is clear, kill the scheduler manually by running the following script:

```
$TOP/bin/go/stop/stop_scheduler.sh
```

2. Abort the entire application by running the following script:

```
$TOP/bin/go/abort_local.sh
```

All services running on the local machine is aborted. The RMI registry is aborted.

Note: Check to make sure all processes have stopped using the 'ps' command.

Kill off any rogue "java" or "rmiregistry" processes that remain after shutting down the instance. Occasionally, it may take several attempts to kill off all java processes. Continue killing all java processes until they are all dead.

5.3.2 Backup

- **Create a full backup of the current WebSphere Product Center directories before applying the Fix Pack. The Fix Pack will overwrite files that have changed. If any issues occur, the backup will allow a rollback to a previous version**
- **It is recommended to apply the Fix Pack to a test system to identify any issues before applying the Fix Pack to a production system**
- **Perform a full backup of the database before applying the Fix Pack to a production system**

Note: Do not delete the old WebSphere Product Center version until performing thorough testing with the new installation.

5.3.3 Delete Tomcat working directory

For configurations using Tomcat, delete the Tomcat working directory using the following command:

```
rm -rf $TOP/etc/default/tomcat33/webapps/ccd
```

Once the working directory has been deleted, restart the application server and apply the Fix Pack.

5.4 Apply Fix Pack

To apply the Fix Pack to WebSphere Product Center, complete the following tasks:

- ❑ Unpack tar file
- ❑ Run WebSphere Application Server script
- ❑ Update configuration files
- ❑ Test installation

5.4.1 Unpack tar file

Purpose: To extract and update any new installation files into the current working directory

Note: GNU tar is needed to untar the WebSphere Product Center files.

1. Copy the WebSphere Product Center tar file to the user or temporary directory.

Example:

```
{HOME_OF_WPC}/tarballs
```

2. CD to \$TOP, the current working directory, and unpack the tar file:

Example:

Using GNU tar, the following command extracts and unzips the tar file using an absolute path:

```
tar zxvf  
/home/WPC/tarballs/wpc_5001_03_fixpak_from_5000_15_was5_db2.tgz
```

5.4.2 Run WebSphere Application Server script

After unpacking the tar file in the previous section, ensure that the default server (server1) is running and run the following WebSphere Application script:

```
$TOP/bin/websphere/install_war.sh
```

Note: Ensure that the default server (server1) is running, as it is required for the WebSphere Application Server script to work. If needed, start the WAS default server by issuing the following command as root:


```
${WAS_HOME}/bin/startServer.sh server1
```

5.4.3 Verify configuration files

Verify all configuration files required by the new installation and make any updates as needed. Refer to the backup copy of the configuration files for the previous installation if needed.

- ❑ common.properties
- ❑ admin_properties.xml
- ❑ init_ccd_vars.sh
- ❑ WebSphere Product Center license file

❖ **common.properties**

On startup, the system will use this file to read in all system level parameters. This file includes settings for the database layer (connection parameters), directory settings, default character sets, thread-pooling parameters, and other settings, which are documented in the file.

File location: \$TOP/etc/default

❖ **admin_properties.xml**

This file is used by the administrative utilities to configure clusters of the application.

File location: \$TOP/etc/default

❖ **init_ccd_vars.sh**

The initialization file is the shell script that initializes the shell variables used by the system.

File location: \$TOP/setup

❖ **WebSphere Product Center License File**

The license file contains the license keys needed to use the WebSphere Product Center application.

File location: \$TOP/etc/default

5.4.4 Run analyze_schema.sh

Recommended: It is recommended to run the analyze_schema.sh script once a week or when there is a twenty percent or more increase/change of data in the database.

Location: \$TOP/src/db/schema/util/analyze_schema.sh

The optimizer uses the catalog tables from a database to obtain information about the database, the amount of data in it, and other characteristics. It then uses this information to choose the best way to access the data. If current statistics are not available, the optimizer may choose an inefficient access plan based on inaccurate default statistics.

Therefore, it is highly recommended that you use the analyze_schema.sh script to collect current statistics on tables and indexes, especially if significant update activity has occurred since the last time the analyze_schema.sh was executed. This provides the optimizer with the most accurate information with which to determine the best access plan. On WPC databases it is recommended to run the analyze schema at least once a week.

To run analyze schema in DB2, refer to the WPC technote: Analyzing schema and collecting statistics in DB2 - <http://www-1.ibm.com/support/docview.wss?uid=swg21205939>

To run analyze schema in Oracle, refer to the WPC technote: Analyzing WebSphere Product Center schema and collecting statistics in Oracle - <http://www-1.ibm.com/support/docview.wss?uid=swg21206376>

5.5 Test Installation

5.5.1 Start WebSphere Product Center

To start the WebSphere Product Center, execute the following script:

```
$TOP/bin/go/start_local.sh
```

The script starts all the services needed to run WebSphere Product Center.

Note: This process should take approximately 30-40 seconds, depending on the speed of the processor.

5.5.2 Check status

Run the `$TOP/bin/go/rmi_status.sh` script that was provided by WebSphere Product Center and verify the following services have started correctly.

- admin_<machine name>
- appsvr_<machine name>
- eventprocessor
- queuemanager
- scheduler
- workflow

YOU HAVE SUCCESSFULLY APPLIED THE LATEST MAINTENANCE RELEASE FOR
WEBSHERE PRODUCT CENTER!

6. Appendix A – LDAP Integration Updates

6.1 Introduction

LDAP (Lightweight Directory Access Protocol) integration improves WebSphere Product Center's security infrastructure with the introduction of three function points into WebSphere Product Center:

- Real-time LDAP entitlements in WebSphere Product Center
- Ability to export a list of users from an LDAP server
- Ability to authenticate users imported from an LDAP server into WebSphere Product Center using the same LDAP server

The WPC-LDAP integration is a two stage process involving changes to WPC and the Tivoli server. This document details the setup and configuration steps that are required for successful deployment of ldap integration.

6.2 Feature Overview

LDAP integration provides the ability to use third party LDAP systems for authentication purposes. Given the complexity involved in using third party LDAP capabilities for authorization, the existing authorization infrastructure available within WebSphere Product Center 5.2 is used to authorize the LDAP users whereas the authentication will be in the realm of LDAP. The entitlements of LDAP users and roles into WPC are both runtime as well as based on user/system invoked script operations. The LDAP user in WebSphere Product Center is differentiated by entries in the 3 columns viz,

1. SCU_LDAP_FLAG set to Y
2. SCU_LDAP_ENTRY_DN
3. SCU_LDAP_SVR_URL

The integration of LDAP with WebSphere Product Center provides an improved security authorization infrastructure that allows the support of 1000+ casual users that require authorization for a variety of (internal and external) roles. For example, Category Managers would be an internal role and an Assistant Brand Manager would be an external role.

For WebSphere Product Center 5.2, LDAP integration is only certified with IBM Tivoli Directory Server version 5.2 (supporting LDAP v3).

6.3 Functional Overview

- If the user exists in WebSphere Product Center and the LDAP flag is not set, WebSphere Product Center authentication is performed
- If the user exists in WebSphere Product Center and the LDAP flag is set, authentication is performed with the LDAP Server. The WebSphere Product Center Role(s) set in the LDAP Server must match the User-Role mappings in WebSphere Product Center

- If the user exists on the LDAP Server and has WebSphere Product Center related roles set, but does not exist in WebSphere Product Center, the User will have the required entitlements created in WebSphere Product Center LDAP flag set

6.4 Assumptions

If a user is authenticated in a session then the user will continue to be authenticated until the end of the session, even if the user identity changes during that period (I.e. change in role, password, etc).

6.5 Pre-requisite

Below are the pre-requisites for this deployment

- Tivoli Directory Server is installed and verified as working correctly.
- WebSphere Product Center Instance is successfully created

6.6 Configuring Tivoli Directory Server

6.6.1 Configure LDAP schema for users and roles

Create a new realm

1. Create a new Realm from the IBM Tivoli Directory Server Web Administration Tool using the menu path **Realms and Templates > Add Realm**.
2. Complete all the required fields.
3. Select the Object Class *domain* as the Parent DN.

For example:

Relative DN Parent DN

cn=myrealm dc=wpcdomain.dc=isl.dc=com

Create a new user template

1. Create a new User Template from the IBM Tivoli Directory Server Web Administration Tool by clicking **Realms and Templates > Add User Template**.
2. Key in above created realm entry as the Parent DN. Select the Structural object class as *inetOrgPerson*.
3. Edit the Required attribute tab to include all the following list of required attributes:
 - Cn
 - Sn
 - Uid (This is the Naming Attribute)

- Mail
 - TelephoneNumber
 - TelexNumber
 - PostalAddress
4. Associate this User Template with the above created Realm using the menu path Realms and Templates > Manage Realms > Edit.

For example:

Parent DN

dc=wpcdomain,dc=isl,dc=com

cn=mytemplate,dc=wpcdomain,dc=isl,dc=com

Create a new user

1. Create a new User from the IBM Tivoli Directory Server Web Administration Tool using the menu path **Users and Groups > Add User**.
2. Select the above-created realm as Realm for this user.
3. Key in the "Required" attribute tab to include all the above-mentioned attributes.

Create a new group

1. Create a new Group from the IBM Tivoli Directory Server Web Administration Tool using the menu path **Users and Groups > Add Group**.
2. Select the previously created realm as Realm for this group. The Object class for the group is groupOfNames.
3. Associate the Users to Groups.

6.6.2 SASL - CONFIGURING DIGEST-MD5 on TDS

Under Server administration in the Web Administration console, expand the Manage security properties category in the navigation area of the Web Administration Tool and select the DIGEST-MD5 tab.

a) Under Server realm, you can use the pre selected Default setting, which is the fully qualified host name of the server, or you can click Realm and type the name of the realm that you want to configure the server as.

Note: If the ibm-slapd Digest Realm attribute in the configuration entry is set, the server uses that value instead of the default for the realm. In this case, the Realm button is pre selected and the realm value is displayed in the field. This realm name is used by the client to determine which user name and password to use.

b) Under Username attribute, you can use the pre selected Default setting, which is uid, or you can click Attribute and type the name of the attribute that you want the server to use to uniquely identify the user entry during DIGEST-MD5 SASL binds.

c) If you are logged in as the directory administrator, under Administrator username, type the administrator username. Members of the administrative group cannot edit this field. If the username specified on a DIGEST-MD5 SASL bind matches this string, the user is has the administrator.

d) When you are finished, click Apply to save your changes without exiting, or click OK to apply your changes and exit, or click Cancel to exit this panel without making any changes.

6.6.3 Configuration Notes

Password Encryption Support

The DIGEST-MD5 mechanism authenticates clients by comparing a hashed value sent by the client with a hash of the user's password. However, because the mechanism must read user passwords, all users wishing to be authenticated through DIGEST-MD5 must have {CLEAR} passwords in the directory. When storing {CLEAR} passwords in the directory, you must ensure that access to password values is properly restricted through ACIs, as described in Chapter 6, "Managing Access Control". You may wish to further protect {CLEAR} passwords by configuring attribute encryption in that suffix, as described in Encrypting Attribute Values.

Refer - <http://docs.sun.com/source/817-7613/ssl.html#wp14354>

Client Side Configuration

The hostname of the Tivoli Server should be entered in the hosts file of the Server.

6.6.4 SSL Setup

To enable security with server authentication, you can follow one of the given steps:

SSL Setup – Certificate from a Certifying Authority (CA) Self Signed Certificate

Create a public/private key pair and obtain and store a certificate from one of the predefined (well-known) Certificate Authorities. This procedure requires less setup because the key ring file is preconfigured with the CA root certificates required to identify the CAs from whom the certificate is issued.

SSL Setup – Self-Signed Certificate

To enable SSL security we can create a self-signed root certificate and store the certificate in the database and class files. To ensure maximum security for your site, you should only use a self-signed certificate for server authentication until you receive a CA-issued certificate.

You can use the ikmGUI utility to create a self-signed certificate to enable SSL sessions between clients and servers. The steps are essentially the same except that, in this case, you are the CA for

the keys you will be creating, and you will be creating your own root certificate. The advantage of using this type of certificate is a quick start, it is free, and you have no dependencies on other organizations. The drawback, on the other hand, is that each client or server using this kind of certificate needs to have the new root certificate imported, which may impose some administrative burden.

1. Create server key database (.kdb file).
 - a. Click **Key Database File** (Figure 15-6 on page 460).
 - b. Click **New**, from that dropdown that appears in point a above.
 - c. On the dialog window that pops up, select **CMS key database** file in the Key database type selection list and then type in the name and location of the key database file to be created. This file has an extension of .kdb, as, for example, in ldap_key.kdb. Then, click **OK** to close the dialog panel.
 - d. A new dialog pops up that requests your input for a password for the key database file, an optional expiration time, and whether or not the password is to be stashed to a file. Enter a password, an optional expiration time, and make sure that you check the check box next to **Stash the password to a file?** otherwise, you have to enter the password manually in the configuration file of the directory server. Click **OK** to close this dialog. The password is then encrypted and stored in a file with the same name as the key database file but with an extension of .sth.
2. Create a self-signed certificate.
 - a. Select **New Self-Signed Certificate...** from the Create pull-down menu in the main window (Figure 15-6 on page 460). In the dialog window that shows up, you will have to fill in the following information:
 - Key label (a clear, descriptive label for the certificate)
 - Key Version (normally X.509 V3, unless you have reasons for other versions)
 - Key size (512 or 1024, depending upon security requirements and country version of the ikmgui utility)
 - Common name
 - Organization and other pertinent information to identify the owner of the certificate
 - Validity period in days
3. Click **OK** to create the request. The .arm file so created contains the certificate request.
4. From the certificate just created above, you need to extract the root certificate that is necessary for other communication partners (clients and/or servers) to recognize the newly created certificate. Here are the steps for exporting the root certificate:
 - a. Select the new certificate's entry in the Personal Certificate list and click **Extract Certificate** at the bottom right on the main window.
 - b. Select **Base64-encoded ASCII data** from the Data type list and enter a file name (with a .arm extension) and a location (directory) for the new root certificate to be exported to. Then click **OK** to export the root certificate. (If

You have now created a file that holds your own root certificate. This must be imported to all communication partners that will connect to the server through SSL.

5. Use the following steps for importing the new root certificate into others' key database (using ikmgui):

- a. Make sure that the certificate extracted above, in the previous step, is made available to all the communication partners. You can transfer the file using ftp or a diskette or any other suitable media.
- b. Invoke the ikmgui utility on the receiving system.
- c. If not already done, create a key database file (see first step above for creating a self-signed certificate).
- d. In the Key database content portion of the window, select Signer Certificates from the selection list and click Add... on the right.
- e. Select Base64-encoded ASCII data from the Data type list and type the Certificate file name and location into the appropriate fields. Then, click OK to import the certificate.
- f. On the upcoming dialog, supply a label for this certificate and click OK.

The steps as described above need to be done on each machine that will communicate using this certificate with the machine from which the certificate was exported.

Each LDAP server should have an individual own certificate. Sharing certificates across multiple LDAP servers is not recommended. By using different certificates and private keys for each server, your security exposure is minimized should a keyring file for one of the servers be compromised.

SSL Setup – Client Side (WPC end)

- a) Get the .arm file generated in the above steps in to the WPC environment
- b) Use the keytool command to import this in to the JVM
Keytool -import -trustcacerts -keystore cacerts -storepass changeit -noprompt -alias mycert -file <cert_file_generated>
- c) Make an entry in hosts file to point to the SSL Tivoli instance.
- d) Change the location of the cacerts file in the keystore variable in the login script. If the file that was generated in the step2 was in /home/sgopan/cert/cacerts, make the value for the keyStore variable to this location.
- e) Set the bindType variable to ssl and sslBindType to either simple or DIGEST-MD5

6.7 Configuring WPC Login Script

6.7.1 Login Script Parameters

Parameter	Default Value	Description
wpcOnlyAuthentication	true	Flag that identifies the authentication mechanism. Set to false in case ldapAuthentication is required

ldap_naming_attribute	Cn	The naming attribute for the users in this LDAP Server
group_naming_attribute	cn	The naming attribute for the groups in this LDAP Server
ldap_urls		Array of LDAP Urls , in stage I , only single LDAP server supported
parent_DNs		The Parent DN where the users are likely to be found (Can be set to "") if you do not know the Parent DN
bindType	ssl	The bind type could be one of simple/sasl/ssl
sslBindType	DIGEST-MD5	The subtypes allowed in ssl bind. One of simple/DIGEST-MD5
keystore	/workspace2/sgopan/austin_521/ldap/cert/cacerts	The location of the cacerts file that has been imported in to the JVM
admin_user	cn=root	The root user in the ldap
admin_password	"root1234	The password of the root user
rootBindType	simple	Should be always simple
rootUserUrl		The SSL Tivoli url
personClass	objectclass=inetorgperson	The person class in the LDAP server
groupOfNamesClass	objectclass=groupOfNames	The groups class
groupOfUniqueNamesClass	objectclass=groupOfUniqueNames	The groups class
Logger_name	ldap	The logger variable configure in log.xml

Log.xml Settings to enable Custom LDAP Logging

To enable logging in the Login,Logout scripts include the following sections in the log.xml file

```
<appender name="ldap" class="org.apache.log4j.FileAppender">
  <param name="File" value="{TOP}/logs/ldap.log" />
  <param name="Append" value="true" />
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d [%t] %-5p %c %x- %m%n"/>
  </layout>
</appender>

<category name="ldap" additivity="false">
  <priority value="debug" />
  <appender-ref ref="ldap" />
</category>
```

</category>

If the entry is not made the logs will be under <appsrvr>/default.log file

6.8 Securing Docstore Documents

The Login, Logout and Trigger Scripts are all associated with User Authentication; therefore they need to be secured so that they have restricted access. To enable restricted access, the Object to ACG mapping is being introduced to individual files in the DocStore.

Note: Currently, this feature does not have the capability to secure Directory paths in the DocStore, but this is planned for a future release. For this maintenance release, this feature should be used only to secure Scripts related to User Authentication ONLY.

By design generated files will be available to be secured. But this feature enables a user to secure any other file in DocStore. The basic governing rules for securing files are driven by two privileges on DocStore i.e. View files and Delete Files. If the user has both these privileges for a file in DocStore via an ACG Map, then the user has full access to the file.

View Files privilege governs reads and writes access to the files.

If the User does not have the View Files privilege then the following will be the resultant in the Application's UI –

1. User will not be able to see the file in the DocStore listing
2. If the file is a script file it will not appear in the listings of Scripts Console

If the User does not have the View Files privilege then the following will be the impact on script operation which use the file in question –

1. Writer::save() throws an exception if there are no permissions.
2. getDocbyPath() throws an exception if there are no permissions.
3. getDocListbyPaths() to throw exception if some files have no permissions.
4. Listing of paths will not be blocked.
5. getScriptbyPath throws an exception if there are no permissions.
6. importEnv throws an exception if there are no permissions.
7. importXML throws an exception if there are no permissions.
8. importXSD throws an exception if there are no permissions.
9. new ExcelBook throws an exception if there are no permissions.
10. new Reader throws an exception if there are no permissions.
11. new XmlDocument throws an exception if there are no permissions.

12. new ZipArchive throws an exception if there are no permissions.
13. sendEmail throws an exception if there are no permissions on the attachments
14. sendFtp throws an exception if there are no permissions.
15. sendHTMLEmail throws an exception if there are no permissions on the attachments
16. sendHTTP throws an exception if there are no permissions on the file to be used
17. sendMultipartPost throws an exception if there are no permissions.
18. startAggregationByName throws an exception if there are no permissions for the data file
19. unzip throws an exception if there are no permissions on the zip file.
20. validateXML throws an exception if there are no permissions.
21. WebService::setImplScriptPath throws an exception if there are no permissions in the script file.
22. WorkflowStep::setWflStepScriptPath throws an exception if there are no permissions on the script file.
23. writeBinaryFile throws an exception if there are no permissions on the path.
24. zip throws an exception if there are no permissions on the files being zipped.
25. Catalog::loadCatalog throws an exception if there are no permissions on the data file.
26. Catalog::massUpdate throws an exception if there are no permissions.
27. Doc::copyDoc throws an exception if there are no permissions.
28. Doc::moveDoc throws an exception if there are no permissions.
29. ExcelBook::savetoDocstore throws an exception if there are no permissions.
30. Writer::writeFile throws an exception if there are no permissions.
31. ZipArchive::addCtgFile throws an exception if there are no permissions.

As per the current design the exception would be thrown by the Docstore API and it might flow upwards as an AustinException. Only the logs would reveal the exact issue. Customers should be made aware of this very clearly.

These script operations will only be affected in the manner described, if the customers attempt to use this feature extensively (which should not be allowed). This feature does not block execution of scripts on which there is no View Files privilege.

6.9 Migration from 521 to 5211

6.9.1 Database Migration

The migration script should be executed only after applying the fix pack on the 521 base version.

The migration script `migrateto5211.sh` is available in `$TOP/bin` directory. The script does the data migration of affected tables(SCU, JHB) in 521 to 5211 .

Usage

If LDAP Enabled flag is set to true in common.properties then the following 3 parameters should be supplied while running the script

- a. LDAP_URL
- b. Naming Attribute
- c. Parent DN of the 521 specific LDAP data

This can be sourced from ldap_config.xml in \$TOP/etc/default folder.

6.9.2 Script Operations

6.9.2.1 New script operations

The following script operations have been added in 5.2.1.

::authenticateWPCUser

Prototype:

Boolean ::authenticateWPCUser(String sUserName, String sPassword, String sCmpCode [, Boolean bEncodedPassword])

Description:

Provides authentication for a WPC user. Optional parameter bEncodedPassword indicates if the password is being passed already encoded. Default is false.

User::getLdapEntryDn

Prototype:

String User::getLdapEntryDn()

Description:

Returns the distinguished name field associated with an LDAP authenticated User.

User::setLdapEntryDn

Prototype: void User::setLdapEntryDn(String sEntryDN)

Description: Sets the distinguished name field associated with an LDAP authenticated User

User::getLdapServerUrl

Prototype: void User::setLdapServerUrl(String sServerUrl)

Description: Return the URL of the server providing this users' LDAP authentication.

User::setLdapServerUrl

Prototype: void User::setLdapServerUrl(String sServerUrl)

Description: Sets the URL of the server providing this user's LDAP authentication.

::populateSecurityContext

Prototype: AustinContext ::populateSecurityContext(User user[, String[] roleNames, InitialLdapContext ldapContext, ICategory organization])

Description: Returns the context for the given user by assigning the access privileges for the roles passed in roleNames. It has no effect on the current users context. If ldapContext is present then a handle of the context will be set in the returned context. If the user is not already present in WPC a new user will be created in the organization specified otherwise in the default organization of default organization hierarchy.

6.9.2.2 Deprecated script operations

The following script operations in 521 are removed. The implementation for the same has been provided in the LDAP Library script

getLdapUserInfo

Usage

getLDAPUserInfo(ctx,parentDN,query,attributes,filename)

- ctx – The Context of the root user in the Tivoli Server
- parentDN – The Parent DN of the User
- query – The LDAP Query that has the users naming attribute
- attributes – The array of required attributes e.g cn, homephone, mobile etc
- filename – The name of the file in docstore in which the user details will be written

getAllLdapUserInfo

Usage

getAllLDAPUsers(ctx,parentDN,searchFilter,attributes,filename)

- ctx – The Context of the root user in the Tivoli Server
- parentDN – The Parent DN of the User
- searchFilter – The LDAP Query with person class
- attributes – The array of required attributes e.g cn, homephone, mobile etc
- filename – The name of the file in docstore in which the all the user details will be written

6.10 Terms

Term	Definition
-------------	-------------------

GSKit	Global Security Kit (GSKit) Version 7a is a software package that is required only if Secure Sockets Layer (SSL) Security or Transport Layer Security (TLS) is required.
gsk7ikm	The graphical utility gsk7ikm (IBM Key Management GUI) is provided for IBM AIX, Windows NT, and a number of other IBM and non-IBM platforms to manage SSL X.509v3 certificate databases (also known as keyring files or keyring databases). Its use is required to configure and use Secure Sockets Layer (SSL). The gsk7ikm utility replaces other utilities, like ikeyman, used with earlier versions of IBM SSL support.
Keytool	A security tool to manage keystores and certificates. Included in the JDK

7. Appendix B – Portal Integration Updates

This appendix lists the features and limitations of the current WebSphere Product Center Portal Enablement.

Note: For more information, refer to the updated Portal Enablement documents that have been provided with this Fix Pack.

7.1 Functionality / Features List

The functionality present in this Portal Enablement release, along with the portlet that implements the functionality is listed in this section. This functionality list is built on top of WPC Portal Enablement 5.2.1 release, but with significant additions and updates to the existing functionality. The newly added or modified functionality is marked with an asterisk (*), and also in specific cases marked with “[New in WPC 5211]”.

7.2 Multiple Catalog Support*

The application works with a predefined set of catalogs, and all the portlets are multi-catalog aware. For instance, this ensures that users can browse for a catalog, and perform a search on that specific catalog.

7.2.1 User Context Portlet*

This portlet keeps track of the currently selected catalog. This portlet can be used to inform the user about this information, For instance, adding this portlet in a page, where the Search Portlet is present, informs the user about the currently selected catalog.

7.3 Globalization support*

This support improves on the existing functionality from the WPC 5.2.1 release. The Portal solution is aware of the user’s preferred locale, and if this is not set, it uses the browser’s locale. This locale information is shared with WPC. This ensures that the portal content, and the data picked up from WPC are locale specific.

The user can also select a locale to apply to specific portlet content; this is the “locale restriction” option. This option will ensure that localized attributes are rendered based on the locale restriction selection.

7.4 Attribute Types Support*

The WPC Attributes types are supported in Portal; rendering support, localized attributes support, and multi-occurrence support. This support is a significant update from the one that existed in WPC 5.2.1 release.

7.4.1 2.3.1. Utility Portlets*

The enhanced attribute support – support for Relationship and Lookup Attributes – require utility portlets to search, browse, and view content. These portlets are rendered as pop-up windows, and can only be used within that context.

- Search utility Portlet – this helps a user to search for an item or lookup values
- Browse Item Catalog utility Portlet – this helps a user to browse for an item, and not browse for lookup values
- Work with Item utility Portlet – this helps a user to view item content, and not view lookup values.

7.5 Register and Approve Portal Users

This provides an option to add a WPC portal user. Every WPC Portal user is associated with a Supplier GLN. Once a user registers with the application, the administrator uses the User Approval portlet to approve the user. This will ensure that the user can log-in to the application.

7.5.1 User Registration Portlet *

The User Profile portlet, present in WebSphere Portal Server 5.1 and higher, is modified for for WPC portal user registration. The Registration Portlet, available in WPC 5.2.1, is deprecated, and has been replaced with the User profile portlet. User registers on the Portal through this portlet. This portlet can be accessed by clicking on the Register link in the login page. This portlet can also be accessed to update the user profile.

7.5.2 User Approval Portlet

The User Approval portlet is accessible to an Administrator. This portlet is used to approve a user's registration request, and also to assign the user with a WPC user name, thus ensuring that the user is able to work with the Business Approval portlet

This portlet can be accessed by an administrator, and is present in the User Approval Tab.

7.6 Browse an Item Catalog*

Browse the Item Catalog, its hierarchies, categories in the hierarchies, and items in categories. Select a Category, and add an Item. Select an Item and view its details, clone, update, or delete it.

The Item Catalog provides support to work with multiple catalogs, and also to define the catalogs that the users need to work on.

This functionality is implemented by the Item Catalog Portlet

7.7 Work with a single Item

Provides the functionality to View, Add, Update, and Clone an Item. This also includes an option to delete an item and an option to download the item details into an EXCEL file. The Item details can be accessed by:

- Performing a search for items, selecting an Item from the search results, and viewing its details
- Selecting an Item from the Item Catalog.
- Selecting an item from the list of items in a Business Process' step

This functionality is provided by the Single-Item Edit portlet

7.8 Work with multiple Items

This provides an ability to update Item details, for a selected set of Items. The user can select the items from a search result, and proceed to perform a mass-update of the items' details. The Multi-Item Edit Portlet provides this functionality

The items details update is an asynchronous process, where the details are submitted to WPC, and the user can proceed with working with other portlets. The status of this asynchronous transaction is tracked in the Transactions portlet.

7.9 Search for items*

Search for Items, by specifying the search criteria. The user has the option of choosing the attributes for a search criterion and the attributes to be displayed as part of the search results. The search performed by a user is stored in a search history.

Search is available in two flavours - Quick Search Portlet and Advanced Search Portlet. Quick search only provides an option for the user to launch a search, by entering values in the search criteria. The actual search is delegated to the Advanced Search behind the scenes. Quick Search has a simpler user interface.

[New in WPC 5211] The Quick, and Advanced search use only the search criteria that the user has explicitly set, while performing a search. For instance, the search criteria consists of "GTIN" and "Short Description"; in this case the user only enters a value for Short Description, then only this field is considered for performing the search.

7.10 Work with a collaboration area or business process

The user is provided with an option to work with a Collaboration Area (Business process, or workflow). The user is presented with list of workflows, depending on the authorization. The user can opt to work with a workflow, by donning a role appropriate to him/her.

The selected workflow and its steps are presented to the user. The user can further drill down into a step, and view the items in the step.

The user has the option to reserve, reserve&edit, and un-reserve items in that step. The user can also perform actions (sets the EXIT value) on the items, to ensure that they move to the next step in the workflow.

7.11 Upload Items defined in a comma-separated-values or MS-EXCEL file

This provides an option for a user to upload items into WPC. The CSV file format should contain the attribute names on the first line, followed by the attribute values; should resemble a tabular format, as shown below.

```
AttrA    AttrB  AttrC  AttrD
value1   value2 value3 value4
value5   value6 value7 value8
```

Note: The attribute names should be fully qualified names - it should contain the path of the attribute and include the Spec name too. For example "ItemSpec/GroupAttribute/GTIN"

Similarly, an MS-EXCEL file containing Item details can be uploaded into WPC. The default format of the MS- EXCEL is similar to the tabular structure shown above. Refer to the WPC documentation on Excel Support, to better understand the various options for the MS-EXCEL upload format.

The upload operation is an asynchronous process, where the file is submitted to WPC, and the user can proceed with working with other portlets. The status of this asynchronous transaction is tracked in the Transactions portlet.

7.12 Download Item details, into an MS-EXCEL file

Items details can be downloaded into an MS- EXCEL file - the items details from the "Work with Item", "Mass Update", and "Search" portlets. All these portlets provide a download option.

Clicking on the download option opens up a "Download" portlet. The user can choose the attributes and their order to be downloaded as part of the Item details.

The download action is an asynchronous process, where the details are submitted to WPC, and the user can proceed with working with other portlets. The status of this asynchronous transaction is tracked in the Transactions portlet. Once successfully complete, the user can download the MS-EXCEL file from a link in the transaction details.

7.12.1 Track completion status of Asynchronous operations

The Transactions Status portlet tracks asynchronous operations, and their completion status.

The portlet constantly polls the transactions for completion status. Once complete, the user can view the details of the transaction - on whether the transaction was successful or not.

In the case of transactions related to downloading item details as MS- EXCEL files, the transaction details also contains a link to download the MS-EXCEL file, if the transaction was successful.

7.13 Saving User specific portlet settings*

The user updates the portlet's display and content using the portlet's "Edit" option. These options are now persisted across user sessions. The Search portlets, Business Approval portlet, and Multi-Item Edit portlet support this functionality.

7.14 Limitations

This sections purpose to inform the user about the limitations of the current release; this has to be viewed from the context of the features that have been implemented in the current release.

7.14.1 Catalog Support

An Item added, cloned, or deleted through the "Work with Item" portlet is not reflected in the Item Catalog. This requires a refresh of the Item Catalog, to reflect any changes.

7.14.2 Rendering of Item Details

- The "Work with Item" portlet's rendering has to be changed to accommodate an Item Specification, other than the one shipped with the product. This requires a modification of the "Work with Item" portlet's JSP. Dynamic rendering of Item details - tabs, and its contents - based on the WPC View specification is not supported.
- The other portlets, which render Item content, dynamically render the Item specification content. They need not be modified.

7.14.3 Item Validation

Item content validation is not performed in the portal application, and is delegated to WPC.

Item content validation, when uploading Item contents in a CSV or MS-EXCEL file, is performed on WPC. This will be performed on WPC, if the Item Specification has value-rules associated with it. Refer to the WPC related documentation for details.

7.14.4 Business Process Support

In the Item details, for a Workflow Step, the items are sortable only by GTIN. There is no support to sort items on any other parameters.

The attributes in the item specification are the ones for which a sort algorithm is applicable. This algorithm can be created in the WPC Script layer.

7.14.5 Transactions tracking

- Currently there is no automated option to archive or cleanup transactions. This has to be performed manually, by deleting the contents of the transactions file:

<WPS_ROOT>\installedApps\<Portal Enablement installation context>\<Portal Enablement WAR file name>\WEB-INF\transdb

- If the portal application is updated or re-installed on WPS, then the transactions list is reset. Take a back up of the file mentioned above, and replace it in the same directory, after the portal application update.
- The portlet auto-refreshes every 20 seconds. This refresh rate is not configurable, and is currently hard-coded in the portlet. This value can be changed in the <WPS_ROOT>\installedApps\<Portal Enablement installation context>\<Portal Enablement WAR file name>\WebContent\jsps\html\alert\transactionlist.jsp file.

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION AS IS WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Burlingame Laboratory
Director IBM Burlingame Laboratory
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not necessarily tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples may include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

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

IBM
the IBM logo
AIX
CrossWorlds
DB2
DB2 Universal Database
Domino
Lotus
Lotus Notes
MQIntegrator
MQSeries
Tivoli
WebSphere

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

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

Java and all Java-based trademarks 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.