

3 *System Use Case Model: Use Case Specifications*

3.1 *Copyright*

```
/*  
 * Copyright (c) 2007 IBM Corporation  
 * All rights reserved. This document is made available  
 * under the terms of the Eclipse Public License v1.0  
 * which accompanies this distribution, and is available at  
 * http://www.eclipse.org/legal/epl-v10.html  
 *  
 * Contributors:  
 *   IBM Corporation - initial use case documentation  
 */
```

3.2 *Interactive Use Cases*

3.2.1 *Offering Use Cases*

3.2.1.1 *UC2: Install a Componentized Offering*

Brief Description

User installs an offering using an interactive install.

Actors

1. User.

UC2 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. User logs onto system with authority to install the offering.
2. User accepts all defaults
3. There are no dependency issues or errors

UC2B.0: User initiates the install of an offering. System automatically proceeds to next step.

UC2B.1: System presents a list of languages available for use during installation. (If applicable, the product language files to install will be chosen later.). Language selection should default to the system locale, if applicable. User chooses language for the install.

UC2B.2: System presents progress while preliminary checks and initialization are performed. This step may involve multiple progress bars. System automatically proceeds to next step.

UC2B.3: System presents welcome message for product installation.

UC2B.4: System presents the license agreement and user accepts.

UC2B.5: System presents options to install only, create response file only or install and create a response file. User takes default action of install only without creating a response file.

UC2B.6: System presents progress while dependency checks are performed. System automatically proceeds to next step.

UC2B.7: System presents choices of installation type. User accepts default. In most cases, the default will be labeled Typical (if only some of the features are being installed) or Full (if all of the features are being installed). This step can also be replaced with install roles (UC2A30).

UC2B.8: System presents installation directory with the option to change it. User accepts default directory.

UC2B.9: System presents progress while dependency checks for features are performed. System automatically proceeds to next step.

UC2B.10: System presents summary screen that states what will be installed. User initiates install.

UC2B.11: System presents progress while install proceeds. System automatically proceeds to next step.

UC2B.12: System presents message that install was successful.

UC2 Alternative Flows

UC2A1. User does not have authority to install.

Previous step: UC2B.6.

UC2A1.1: System presents error message to user stating that user does not have authority to run install, but can create a response file, which can be silently installed by a user with authority.

UC2A1.2: System presents option to create a response file. User can choose response file and perform a silent install when logged in with proper authority.

Next step: UC2B.6.

UC2A2. Required prerequisites are not met so that install cannot run.

Description: This error message should occur only when the installer starts but cannot run enough to display the first panel.

Previous step: In UC2B.0, installer cannot start.

UC2A2.1: System presents error message such as “<Product Name> cannot be installed. <Failure Details>”.

Next step: Exit.

UC2A3. Configuration is needed during installation.

Description: Configuration steps could be presented at various places in the install flow. Normally, they would be presented after the path has been chosen and before the summary or, if there is a separate config tool, it can be launched after install to appear more seamless. The specifics of this flow are very dependent on the application being installed, and the designs should reflect good usability principles for presenting and eliciting the required information.

Previous step: Typically UC2B.8, but possibly others as needed.

UC2A3.1: System presents message with configuration options.

Next step: UC2B.9, in most cases, but might be needed before the installation path is specified (for example, when node name is in path) or other appropriate location.

UC2A4. Install of an offering when initial configuration fails. User reviews failure information and is offered choice to review log file.

Previous step: UC2B.2.

UC2A4.1: System presents message with details of the failure in wording understandable by the user.

Next step: Exit.

UC2A5. Prerequisite needed is not installed and offering install does not include the prerequisite.

Description: This flow occurs when a dependency is not met for either installing the offering itself or a feature and the dependency cannot be installed by the installer. Missing dependencies should be presented as a single list since there could be more than one dependency failure.

Previous step: A missing dependency is found in UC2B.6.

UC2A5.1: System presents message listing the dependencies.

Next step: Either exit or proceed to UC2B.7, depending on dependencies missing.

UC2A6. Install of an offering when soft dependencies are not met.

Description: This flow applies to situations where the current configuration meets the minimum, but not the recommended level: for example, when the computer has less memory installed than is recommended. This function depends on having the distinction between hard and soft requirements.

UC2A7. Install of an offering when selected features have conflicting dependencies.

Description: If at all possible, this situation should be avoided. It can be avoided by designing the interface so that the features are mutually exclusive when users are selecting features (e.g., radio buttons) or by separating mutually exclusive features as different roles in UC2A30.

Previous step: Features selected in UC2A28.1 conflict with each other.

UC2A7.1: System presents message stating that all of the conflicting dependencies will be deselected.

Next step: Return to previous step (UC2A28.1) with both features unselected, but enabled if possible.

UC2A8. Install of an offering fails when some parts have been installed or partially installed and some have not.

Description: The message should explain the reason for the failure, then initiate an uninstall if appropriate. If there is a reason that users would want to keep the updated version (for example, a fully functional application server), users can be prompted with the second message (UC2A48) giving them the option to retain the installed update.

Previous step: UC2B.11.

UC2A8.1: System presents message that installation failed, with reason for failure.

Next step: Exit or, if user has the option to keep installed parts, go to UC2A48.

UC2A9. Interactive offering install is performed using post-GA dependency list.

Description: This flow has been moved to UC51.

UC2A10. Dependency analysis results indicate that prerequisites are met only if one or more of the contained, but not currently selected, features are installed.

Description: In this flow, the user has selected a feature that has a prerequisite or corequisite, but has not selected the required feature. When the user selects a feature, any prerequisite or corequisite feature not selected should automatically be selected.

Previous step: User chooses custom install type but then, during the feature selection step (UC2A28.1), does not select its prerequisite.

Next step: Automatically select any prerequisite or corequisite features when user selects a feature that requires it (UC2A28.1).

UC2A11. Install of an offering that requires that combinations of choices be iteratively checked for validity.

Description: In this flow, selecting some options will preclude features that are presented subsequently. This should be done in one of three ways, depending on the circumstances:

1. If the selections are at a high enough level and can be fit into a role, consider a role-based install (UC2A30).
2. If the exclusive options are at the levels of features, make the exclusivity apparent through interface design.
3. If neither of these options applies, then UC2A11 should apply.

Previous step: UC2B.6.

UC2A11.1: System presents appropriate options that will filter features available. User chooses option.

Next step: UC2A28.1.

UC2A12. Install of an offering fails when some parts have been installed or partially installed and some have not.

Description: This flow should be identical to UC2A8 for the user.

UC2A13. Install of an offering that upgrades an earlier version and defers migration until after the install (version or release has changed).

Previous step: During UC2B.6, system detects that migration will be necessary.

UC2A13.1: System presents user with message that offering will be updated.

Next step: Continue install at UC2B.7, unless there are required migration-related steps.

UC2A14. Install of an offering that upgrades and migrates data from an earlier version.

Description: For users, this should be the same as UC2A13.

UC2A15. Install of an offering when missing topology dependency can be met by installing additional software included with the offering.

Previous step: UC2B.6.

UC2A38.1: System presents message describing change.

Next step: Exit or UC2B.7.

UC2A16. Install of an offering that upgrades from an earlier version, upgrades a prerequisite, and defers migration.

Description: For users, this flow is the same as UC2A13.

UC2A17. Install of an offering when migration fails.

Description: This alternate flow occurs when migration fails while current files are being backed up and before the files are migrated. In this case, installation should be canceled. If the failure occurs during or after migration, UC2A59 should apply.

Previous step: User selects migration-related options, and then migration fails.

UC2A17.1: System presents message stating reason for failure.

Next step: Exit.

UC2A18. Install of an offering that requires software that is already installed on the system and that software does not allow additional copies to be installed and is shareable.

Description: The user interface for this flow should be identical to the basic flow. The application should be configured to work with the shared software without prompting the user.

UC2A19. Install of an offering that requires software that is already installed on the system and that software allows multiple installs and is not shareable.

Description: The user interface for this flow should be identical to the main flow. The component should be installed without prompting the user.

UC2A20. Install of an offering that requires software that is already installed on the system, and that software allows multiple installs and is shareable.

Description: By default, the software that is already on the system should be shared unless there is a reason to do otherwise.

UC2A21. Install of an offering that carries a component to update the product (for example, General Maintenance Installer) as a common component on a system where it has not already been installed or is downlevel.

Description: In this case, the update component should be installed if it has not been installed already, or updated if the installed version is downlevel. There are no additional steps for this screen flow, unless upgrading will present a conflict.

UC2A22. Elementary check cannot be performed. Installation should stop at this point.

Previous step: Failure during UC2B.6, UC2B.9 or UC2B.11.

UC2A22.1: System presents message describing cause and stating that installation cannot proceed. If the cause of the failure is known, provide details about the cause.

Next step: Exit.

UC2A23. Install where user has skipped dependency checking, and one is missing

Description: In this flow, the application will not be able to run since it is missing a dependency. A note should be made in the log that the check was skipped, but since the dependency check is not run during install, the user will not know until runtime, when the offering will probably not run.

This alternate flow needs to be supported for support purposes and should be initiated in a non-documented manner (for example, a non-documented command line). It should not be used by end users in most cases.

UC2A24. User cancels install prior to file copy.

UC2A25. User cancels install during file copy.

Description: Both of these flows are identical for users. The system should present the message when they cancel either before any of the files have been copied, or before all of the files have been copied (before or during UC2B.11). They should have the option to return to the installer or exit installation.

Previous step: User cancels installation at any time before file copy is complete (during UC2B.11).

UC2A24.1/UC2A25.1: System presents message confirming cancel.

Next step: Exit and initiate uninstall.

UC2A26. User cancels install after file copy, but before config is complete.

Description: This flow would most likely apply to an aggregated offering where fully usable versions of one or more offerings have already been installed.

Previous step: User clicks Cancel button any time after file copy is complete, but before configuration complete (during or after UC2B.11).

UC2A26.1: System presents message confirming that user wants to cancel installation.

UC2A26.1: System presents message giving users option of either fully uninstalling all of the offerings or keeping any usable offerings installed during the installation.

Next step: Exit and initiate uninstall.

UC2A27. User input is required during file copy, for example, file already exists.

Description: If possible any confirmations should be presented before the pre-install summary (UC2A.10). If it is not possible to avoid this situation, then users should be presented with the appropriate message describing the options that are possible.

UC2A28. User picks Custom install and is presented choice of features to install.

Previous step: User chooses Custom path in UC2B.7.

UC2A28.1: System presents features in feature selection options. User chooses features.

UC2A28.1: System presents languages. User chooses languages.

Next step: UC2B.8.

UC2A29. User picks other install type such as Compact or Quick.

Description: In this flow, users choose an option other than Typical and Custom in UC2B.7. Many of these options, such as Compact and Quick, will immediately proceed to UC2B.8, but will install a different set of options. If users choose Custom, the next step should be UC2A28.1.

Previous step: User chooses Compact, Quick, or installation type other than Custom or Typical in step UC2B.1.

Next step: UC2B.8 or UC2A18.1.

UC2A30. User picks particular install role (when Typical or Custom options are not offered).

Description: This step is an alternative for step UC2B.7, and would be used when the feature set can better be described as a role: for example, Client, or Server.

Previous step: Installer uses install roles instead of standard options (Typical, Custom, Compact); replaces step UC2B.7.

UC2A30.1: User chooses install type.

UC2A30.2: User chooses languages.

Next step: UC2B.8.

UC2A31. Processes must be stopped or started in order for install to be completed.

Description: Installation requires processes to be stopped or started. Users in this case should have two choices:

1. Have system wait while users manually stop processes, and then check again.
2. Have system terminate these processes (do a hard stop).

Previous step: UC2B.9.

UC2A31.1: System lists detected running processes that will prevent successful install, and offers to stop these processes automatically or allow the user to stop them.

Next step: UC2B.10.

UC2A32. User wants to restart or stop running processes that were started or stopped after install was completed.

Description: Processes that have been stopped for install (see UC2A31) can be restarted at the end of install. Since the system stopped the processes, this step helps the user get the system back to the condition it was in before install.

Previous step: Replaces post-install summary (UC2B.12) when processes have been stopped.

UC2A32.1: System presents message giving user the option of restarting the processes.

Next step: Exit.

UC2A33. Install of additional features to an offering.

Previous step: User runs the installer after installing one or more copies of the offering. This might occur when the user inserts install media, runs uninstall from the current install path or selects copy in Add/Remove Programs (Microsoft Windows Add/remove programs).

UC2A33.1: System presents welcome message for offering modify.

UC2A33.2: System presents post-install options. User chooses to modify.

UC2A33.3: System presents features. User chooses features.

UC2A33.4: System presents languages. User chooses languages.

Next step: Continue to installer main flows at UC2B.9.

UC2A34. Install new copy of offering (coexistence).

Previous step: User runs the installer after installing one or more copies and coexistence is supported. This might occur when the user inserts install media. In UC2A33.1, user chooses an option to add another copy.

UC2A34.1: System presents post-install options. User chooses option to install new copy.

Next step: User completes installer main flow starting at UC2B.4. The install should be identical to the main flow, except the default install path should have a number starting with an integer. The integer the first time this occurs should be a 2 and should increment by one for each installation.

UC2A35. User is given post-install options during post-install summary, such as launching First Steps, verifying installation, and configuring the installed offering.

Previous step: Replaces 2B.12 when post-install options are available.

UC2A35.1: System presents message providing available options.

Next step: Exit.

UC2A36. User launches install program from Setup Launchpad. Language selection will be presented prior to Setup Launchpad and can pass the language selection to the install program (skip 2B.1).

Previous step: User launches set from Launchpad.

Next step: Launch installer at UC2B.3 in Launchpad Language.

UC2A37. Install that updates related offerings, such as dependencies, without any migration or upgrade to the main offering: for example, WAS Enterprise ships WAS Base with interim fixes.

Previous step: UC2B.6.

UC2A38.1: System presents message. Users can choose to install or not install update.

Next step: Exit or UC2B.7.

UC2A38. Install that updates dependency, without any migration or upgrade to the main offering, but the updates of the dependency will 'break' other offerings that depend on it.

Description: For users, this is similar to UC2A7, but this occurs when an installed component conflicts with a prerequisite.

Previous step: UC2B.9.

UC2A38.1: System presents message listing software to be updated.

Next step: Exit or UC2B.10.

UC2A39. Install that requires action based on the dependency that was found.

Description: This flow will be application-dependent and occurs when user input is required due to the results of a check. For example, if the general dependency is on a database server, the install may do action A if it is IBM DB2 and action B if it's Oracle. These steps should occur either between UC2B.6 and UC2B.7 or between UC2B.9 and UC2B.10.

UC2A40. Install that migrates data from a competitive product.

Description: For users, this should be the same as UC2A13. In the first step, migrating from a competitive product should be the same as migrating from an IBM product. For subsequent steps, custom steps might be required to deal with any product-specific decisions users need to make.

UC2A41. Pieces are added to an offering post-GA, such as separate downloads that get registered as part of the offering or replace part of the offering (or both) without being a maintenance package.

Description: This alternate flow describes something like a Tech Preview or similar post-GA feature addition that might or might not be part of a maintenance package. The general principal should be that the piece is uninstalled the same way that it is installed. If it is installed as a maintenance package, it should be installed or uninstalled via the products' maintenance installer. If it has its own installer, it should have its own uninstaller and be registered with the operating system, such as through Add/Remove Programs (Windows only).

UC2A42. User specifies install directory that already exists and is not empty.

Description: This flow occurs in various locations (for example, UC2B.5 or UC2B.8) when the user specifies a path name that already exists and contains files. One option should allow the path to be selected. The other option should return the user to the step where the path was selected to allow the path to be changed.

Some products, like DB2, install multiple products to the same directory. In that case, this flow would not apply.

Previous step: User specifies a path or file name that already exists in UC2B.8, and possibly in other steps.

UC2A42.1: System presents message giving user the option to either overwrite any files or (manually) change the path.

Next step: Return to previous step, and either overwrite files or allow user to type new path.

UC2A43. User skips configuration during install, which results in the product not being able to run.

Description: In this case, the user is either offered the opportunity to run the configuration at the end of the install and does not run it or cancels install and keeps the installed components (UC2A48). If possible users should be warned that this will occur.

UC2A44. Dependency satisfied by multiple options. User must choose one.

Description: Multiple installed offerings can satisfy a prerequisite. For example, a user may have two different databases required.

Previous step: This should occur after UC2B.6 and before UC2B.10.

UC2A44.1: System presents message allowing users to choose one of the dependencies.

Next step: Continue to UC2B.7 and apply user selection.

UC2A45. User chooses option to view feature details associated with an installation type.

Previous step: User selects an install role (UC2B.7 or UC2A30.1) and initiates the details view, for example, by clicking a Details button.

UC2A45.1: System presents details.

Next step: Return to previous step.

UC2A46. Reboot message required after install.

Description: Users should not be forced to reboot if at all possible. If they do need to reboot, the reboot should replace the last step in installation or occur after the post-install summary. If they are going to be asked to reboot, any post-install options (for example, UC2A35) should not be included on the post-install summary.

Previous step: Install requires a reboot. Users should not be asked to reboot until UC2B.12.

UC2A46.1: System presents message that reboot is required and offers options of rebooting now or at a later time. User chooses reboot now.

Next step: Exit and reboot computer.

UC2A47. User needs to select installation of an offering when multiple ones exist.

Previous step: UC2A33.2 and multiple installed copies of an offering are installed, and user needs to choose one.

UC2A47.1: System presents message. User selects the appropriate offering.

Next step: Proceed to UC2A33.3 and continue applying changes to selected copy.

UC2A48. User cancels or install fails, but user has the option of keeping installed parts.

Previous step: UC2A8.1, possibly others.

UC2A48.1: System presents message giving users the option of keeping or removing some installed parts.

Next step: Exit.

UC2A49. Offering uses Express Install.

Description: This is the same as the main flow, but might have the response file step removed.

Note: This flow needs to be reviewed by the Express Team. It is expected that this flow would be developed by individual product install teams.

Previous step: User initiates the install of an offering directly from Launchpad.

UC2A49.1: System presents a list of languages available during installation. It should default to the locale, if applicable.

UC2A49.2: System presents progress while preliminary checks and initialization are performed. System automatically proceeds to next step.

UC2A49.3: System presents welcome message for offering install.

UC2A49.4: System presents the license agreement and user accepts.

UC2A49.5: System presents progress while dependency checks are performed. System automatically proceeds to next step.

UC2A49.6: System presents choices of installation type. User accepts default.

UC2A49.7: System presents installation directory with the option to change it. User accepts default directory.

UC2A49.8: System presents progress while dependency checks for features are performed. System automatically proceeds to next step.

UC2A49.9: System presents summary screen that states what will be installed. User initiates install.

UC2A49.10: System presents progress while install proceeds. System automatically proceeds to next step.

UC2A49.11: System presents message that install was successful.

UC2A50. Aggregated offering is made up of several offerings that can function as stand-alone offerings.

Description: This flow is covered by the UC2 basic flow and has been removed.

UC2A51. Processes automated stop fails.

Previous step: Installation requires processes to be stopped or started (UC2A31). User allows installation program to stop process, and hard stop fails.

UC2A51.1: System presents message that processes could not be stopped or started automatically. System can offer choice to try again or allow user to do it manually.

Next step: UC2B.10.

UC2A53. Install fails or user cancels install and user is offered option to leave or remove an offering that was originally installed as part of a second offering.

Previous step: UC2A24.1 or various error messages.

UC2A53.1: System presents message giving choice of keeping or removing the offering after the first offering is removed. User chooses to remove it.

UC2A53.2: System presents progress while uninstall proceeds.

UC2A53.3: System presents success message. System automatically proceeds to next step.

Next step: Exit.

UC2A54. Offering uses Express install and user chooses Custom install.

Previous step: User chooses Custom install type in UC2A49.6.

UC2A54.1: System presents features. User chooses features.

UC2A54.2: System presents languages. User chooses languages.

UC2A54.3: System presents installation directory with the option to change it. User accepts default directory.

Next step: UC2A49.7.

Note: This flow needs to be reviewed by the Express Team. It is expected that this flow would be developed by individual product install teams.

UC2A55. Dependency is no longer satisfied by an alternative as the result of an upgrade (if, for example, support is removed).

Description: For users, this is identical to UC2A5.

UC2A56. A new requirement for a later version of a prerequisite is added as the result of an upgrade.

Description: For users, this is identical to UC2A5.

UC2A57. A new requirement is added as the result of an upgrade.

Description: For users, this is identical to UC2A5.

UC2A58. Offering requires software that is already installed, does not allow additional copies to be installed and is not shareable.

Description: This is covered by UC2A38. If possible, this situation should be avoided.

UC2A59. Install of an offering when migration fails.

Previous step: User selects migration-related options, and then migration fails.

UC2A59.1: System presents message giving user the choice to continue or cancel installation.

Next step: Exit or continue with install, depending on user choice.

UC2A60. User without root permissions installs the offering.

Description: Users should be presented with a message as early as possible in the flow listing the limitations (for example, if DB2 Admin Server is installed as a non-root user, it might be possible to administrate only a local DB, not a remote DB). Additionally, some of the defaults might need to change in the flow (for example, path would use the user's path in DB2), and a check might need to be done to make sure the selected path can be written to. Note that this is a case where install can proceed, with limitations, as opposed to UC2A1, where users cannot proceed with install.

UC2A61. User has offering A installed, which contains a piece of offering B. User then installs full offering B.

Description: Offering is installed when a piece of that offering is already installed.

The offering B install should function as a full offering install (not a modify), but denote that a piece is already installed and what its relationships are. An example of this situation is when a client is already installed, and the full version is installed. This flow differs from UC2A5, because it might not apply across versions.

Previous step: UC2B.6 or UC2B.9.

UC2A61.1: System presents message telling user that a component of offering B is installed and offers upgrade, if appropriate.

Next step: UC2B.7 or UC2B.10.

UC2A62. Install of offering that ships another offering or component when newer copy of that offering or component is already installed.

Description: Users should be presented with this check as early in the flow as possible. The message should be displayed along with the suggestion to check for maintenance for the currently installed offering since an update to dependencies might confirm that the newer component will function with offering being installed. If possible, configure the newer copy to work with offering being installed.

UC2A63. User does not accept license agreement.

Previous step: User chooses "I do not accept..." in UC2B.4.

UC2A63.1: System presents message stating that accepting the license is required to install the product.

Next step: Return to UC2B.4.

UC2A64. Install path exceeds operating system limit for path characters.

Previous step: In UC2B.8 or other step where users type in a path to create, user types name that exceeds operating system limitation (number varies by operating system).

UC2A64.1: System presents message that path (repeat the path back to user) is not valid.

Next step: Return to UC2B.8 or previous step.

UC2A65. User specifies a response file name to create that already exists.

Description: This alternate flow occurs when the user chooses to create a response file with a name that already exists. Users can overwrite the existing file or create a new one by entering a new name.

Previous step: UC2B.5. User specifies a file name that already exists.

UC2A65.1: System presents message warning that the file already exists and allowing user to overwrite or cancel.

Next step: If user chooses to overwrite file, continue to next step. If user does not overwrite file, return to previous step, where user can change the name of the file.

UC2A66. User specifies an install path that is invalid.

Previous step: In UC2B.8 or other step where users type in a path to create, user types path that contains invalid characters..

UC2A64.1: System presents message that path (repeat the path back to user) is not valid.

Next step: Return to UC2B.8 or previous step.

UC2A67. User specifies a path or network address, and search for address times out because host cannot be reached.

Description: In a path field, user enters a remote path, but connection times out.

Previous step: Any panel where a user enters a path.

UC2A67.1: System presents error message stating that path could not be reached.

Next Step: Return to calling panel.

UC2A68. User specifies a path, but path cannot be found. This might be either a local or a remote path.

Description: User enters a path that does not exist when it does not make sense to create one: for example, if pointing to the location of a file to read. If the user is creating a file (for example, a response file), that path should be automatically created.

Previous step: Any panel where a user enters a path.

UC2A68.1: System presents error message stating that path or address could not be found.

Next Step: Return to calling panel.

UC2A69. User tries to run multiple installs or uninstalls at the same time.

Description: User tries to open another install while one is already running. The system should display a message stating that only one install can run at a time.

UC2A70. User selects to create a response file, and needs to input file name and directory.

Previous step: UC2B.5.

UC2A70.1: System presents a default response file name and directory. Users can override this default by entering a fully qualified file name, or if using a GUI to browse to a path and enter a file name. If file already exists, users are asked if they want to replace it.

Next step: UC2B.5.

UC2A71. User specifies a response file name that is invalid.

Description: Occurs when invalid characters entered or user specifies a string that cannot be a valid path.

Previous step: In UC2B.5, user enters a response file name that is invalid, because the name is too long, contains invalid characters or is otherwise invalid.

UC3A24.1: System presents error message indicating that entry is not a valid path.

Next step: Return to calling panel.

3.2.1.2 UC3: Uninstall offering

Brief Description

User uninstalls a specific offering using an interactive uninstall.

Actors

1. User.

UC3 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. Product installer logs on to system using an ID that has authority to perform update.
2. Product and uninstall program have been previously installed.
3. There are no dependency issues or errors

UC3B.0: User launches uninstall program.

UC3B.1: System presents a list of languages available during uninstall. It should default to the locale, if applicable.

UC3B.2: System presents progress while preliminary checks and initialization are performed. This step may involve multiple progress bars. System automatically proceeds to next step.

UC3B.3: System presents welcome message for offering uninstall.

UC3B.4: System presents choices such as remove or modify. User chooses to remove offering.

UC3B.5: System presents options to uninstall only, create response file only or uninstall and create a response file. User takes default action of uninstall only without creating a response file.

UC3B.6: System presents confirmation that states what will be uninstalled. User initiates uninstall.

UC3B.7: System presents progress while uninstall proceeds. System automatically proceeds to next step.

UC3B.8: System presents message that uninstall was successful.

UC3 Alternative Flows

UC3A1. User does not have authority to uninstall.

Previous step: UC3B.5.

UC3A1.1: System presents error message to user saying that user does not have authority to run install, but can create a response file, which can be silently installed by a user with authority.

UC3A1.2: System presents option to create a response file. User can choose response file and perform a silent install when logged in with proper authority.

Next step: UC3B.5.

UC3A2. Relationship information determines uninstall will corrupt system, or cause other programs to not function.

Previous step: After user chooses to uninstall (UC3B.4), it is determined that part of the offering to be uninstalled is shared by other offerings.

UC3A2.1: System presents message to user that gives user the name and path information about the affected offering and the choice of continuing or quitting.

Next step: Continue uninstall at UC3B.5 and deal with conflict as user specifies.

UC3A3. Not all installation directories or files could be removed.

Previous step: During uninstall (UC3B.7), not all files can be removed. This might occur because they are being used by a running application.

UC3A3.1: System presents message to user (replaces UC3B.8).

Next step: Exit.

UC3A4. Unregistration of offering uninstall is not successful.

Previous step: Not all registry entries can be removed (UC3B.7).

UC3A4.1: System presents message to user (replaces UC3B.8).

Next step: Exit.

UC3A5. Offering uninstall has updates applied.

UC3A6. Offering uninstall has fixes applied.

UC3A7. Offering uninstall has both updates and fixes applied.

Description: The flows for UC3A5, UC3A6 and UC3A7 should be identical to the main flow, unless there are issues such as dependencies (UC3A2) or feature uninstall (UC3A9). Fixes and updates should be removed before uninstall of offering proceeds, if needed.

UC3A8. User selects copy to uninstall.

Description: If the specific copy to be uninstalled is not known, then this would apply. This only occurs if it is not clear which copy is intended to be uninstalled. For example, if the user launches the uninstall from within one of the products' directories, then this flow would not apply.

Previous step: User selects to uninstall (UC3B.4), and multiple copies of an offering exist.

UC3A8.1: System presents message to user asking which copy to uninstall.

Next step: UC3B.5.

UC3A9. User selects to uninstall features of an offering.

UC3A10. User selects languages to uninstall.

Description: If these options are available, they should be an option in step UC3B.4.

UC3A11. Processes must be stopped or started in order for uninstall to be completed.

Description: Installation requires processes to be stopped or started. User in this case should have two choices:

1. Have system wait while user manually stops services, then check again.
2. Have system terminate these processes (do a hard stop).

Previous step: Before UC3B.6, requires processes to be stopped or started.

UC3A11.1: System lists detected running processes that will prevent successful install, and offers to stop these processes automatically or allow the user to stop them.

Next step: Continue uninstall at UC3B.6 and deal with processes as user specifies.

UC3A12. User wants to restart or stop running processes that were started or stopped after uninstall has been completed.

Description: Processes that have been stopped for install (see UC3A11) can be restarted at end of install. Since the system stopped the processes, this step helps user get system back to condition it was in before install.

Previous step: Replaces Finish Step (UC3B.8) when processes have been stopped.

UC3A12.1: System provides option to restart running processes that were stopped.

Next step: Finish and exit install.

UC3A13. Uninstall when part of offering (shared component) has been upgraded by another offering.

Description: This flow should be identical to the main flow, unless there are issues such as dependencies or feature uninstall. The shared component should not be removed, and a message should be added to the summary information at the end.

UC3A14. Uninstall fails. System presents failure info and user is offered choice to review log file.

Previous step: Failure during uninstall is detected (UC3B.7).

UC3A14.1: System presents message to user with failure details (replaces UC3B.8).

Next step: Exit.

UC3A15. Uninstall asks users if they would like to remove user data.

Previous step: User has option of removing user data after uninstall (UC3B.5).

UC3A15.1: System presents message to user giving the choice of keeping or removing user data.

Next step: Keep or remove files depending on user choice and proceed to post-install summary (UC3B.6).

UC3A16. When offering installed a now-shared component, inform user that the component is shared.

Description: This flow is identical to UC3A2 for users.

UC3A17. Uninstall when there are multiple copies of the offering.

Description: This flow has been removed and is covered by UC3A8.

UC3A18. Reboot message required after uninstall.

Description: Users should not be forced to reboot if at all possible. If they do need to reboot, the reboot should replace the last step in installation or occur after the post-install summary.

Previous step: Install requires a reboot. Users should not be asked to reboot until or after UC3B.8.

UC3A18.1: System presents message that reboot is required and offers options of rebooting now or at a later time.

Next step: Exit and reboot if selected.

UC3A19. Offering uses Express uninstall.

Description: This is the same as the main flow, without steps to choose a language or create a response file.

Note: This flow needs to be reviewed by the Express Team.

Previous step: User launches uninstall.

3A19.1: System presents welcome message for uninstall of offering.

3A19.2: System presents post-install options. User chooses to remove offering.

3A19.3: System presents confirmation message that states what will be uninstalled. User proceeds with uninstall.

3A19.4: System presents progress while uninstall proceeds. System automatically proceeds to next step.

3A19.5: System presents message that uninstall was successful.

UC3A20. Processes' automated stop fails.

Previous step: Installation requires processes to be stopped or started. User selects automatic stop, and stop fails during UC3B.7.

UC3A20.1: System presents message to user. See UC2A51.

Next step: UC3B.7

UC3A21. User is offered option to leave or remove an offering that was originally installed as part of a second offering.

Description: This flow is generally for offerings that users might have a reason to keep, such as Solution Installation or IBM HTTP server for WAS.

Previous step: UC3B.8.

UC3A21.1: System presents message giving choice of keeping or removing the offering after the first offering is removed. User chooses to remove it.

UC3A21.2: System presents progress while uninstall proceeds. System automatically proceeds to next step.

UC3A21.3: System presents success message.

Next step: Exit.

UC3A22. User uninstalls multiple offerings that are treated as an aggregated offering.

Description: This alternate flow describes a situation like DB2, which might aggregate multiple offerings and treat them as one, and so it makes sense to uninstall multiple offerings simultaneously. They could have been installed together or separately, but are located in the same directory, interdependent or otherwise grouped in a way that means it makes sense to uninstall them together.

In this case, there should be a step that lists all of the offerings that can be uninstalled from within that location.

UC3A23. User specifies a response file name to create that already exists.

Description: This alternate flow occurs when the user chooses to create a response file with a name that already exists. Users can overwrite the existing file or create a new one by entering a new name.

Previous step: In UC3B.5, user specifies a file name that already exists.

UC3A23.1: System presents message warning that the file already exists and allowing user to overwrite or cancel.

Next step: If user chooses to overwrite file, continue to next step. If user does not overwrite file, return to previous step, where user can change the name of the file.

UC3A24. User specifies a response file name that is invalid.

Description: Occurs when invalid characters entered or user specifies a string that cannot be a valid path.

Previous step: In UC3B.5, user enters a response file name that is invalid, because the name is too long, contains invalid characters or is otherwise invalid.

UC3A24.1: System presents error message indicating that entry is not a valid path.

Next step: Return to calling panel.

UC3A25. User specifies a response file path or network address, and search for address times out because host cannot be reached.

Description: In UC3B.5, user enters a response file name that cannot be reached.

Previous step: Any panel where a user can enter a path.

UC3A25.1: System presents error message stating that path could not be reached.

Next Step: Return to calling panel.

UC3A26. User tries to run multiple installs or uninstalls at the same time.

Description: User tries to open another install while one is already running. The system should display a message stating that only one install can run at a time.

UC3A27. User selects to create a response file, and needs to input file name and directory.

Previous step: UC3B.5.

UC3A27.1: System presents a default response file name and directory. Users can override this default by entering a fully qualified file name or, if using a GUI, to browse to a path and enter a file name. If file already exists, users are asked if they want to replace it.

Next step: UC3B.5.

3.2.2 Maintenance Use Cases

3.2.2.1 UC11: Install maintenance

Brief Description

User applies maintenance (installs maintenance package) to an existing offering using an interactive install.

This use case covers the interactive installation of a previously downloaded maintenance package where the user needs to specify what maintenance to install. A maintenance package can be an update (fix pack, refresh pack, for example) or fix (such as interim fix or test fix). If the maintenance process is changed so that selection and download of maintenance package is fully automated maintenance, an automated install maintenance use case would likely be different. It is necessary to support both manual and automated maintenance package installs.

Actors

1. User.

UC11 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. User logs on to system using an ID that has authority to perform update.
2. Maintenance installer is installed, or otherwise available for use.
3. User has direct access to maintenance package on the hard drive, CD or other media, or network distribution.
4. Basic flow assumes only one fix pack maintenance package can be installed at a time. Alternate flows cover the install of one or more fixes at the same time.
5. Maintenance package can be installed to only one offering or offering copy (instance) at a time.
6. Maintenance package is readable by maintenance installer.
7. There are no dependency issues or errors

UC11B.0: User initiates maintenance install. System automatically proceeds to next step.

UC11B.1 (UC19B.1): System presents a list of languages that can be used by the install program. (If applicable, the product language files to install will be chosen later.) Language selection should default to the user or system locale, if applicable. User chooses language for the install.

UC11B.2 (UC19B.2): System presents progress while preliminary checks and initializations are performed. This step may involve checks and progress may need to be presented in several substeps. System automatically proceeds to next step.

UC11B.3 (UC19B.3): System presents welcome message for maintenance activity.

UC11B.4: System presents options to install or roll back maintenance packages. User chooses install maintenance package.

UC11B.5: System presents list of offerings that are installed. User selects the offering to which maintenance package will be applied. Note that each copy of a coexisting offering should be treated as a different offering with a separate entry in this panel. If maintenance installer is specific to an offering, this skip may be skipped.

UC11B.6: System prompts for directory where maintenance packages for offering are located. If possible, defaults should be included if there are reasonable ones such as the last directory selected or those recommended for download in readme documents or on the product webpage.

UC11B.7: System presents message and progress while checking current system state and generating list of maintenance packages in the selected directory. System automatically proceeds to next step.

UC11B.8: System presents list of maintenance packages. (If maintenance list cannot be automated to provide latest maintenance packages that are appropriate for system, then maintenance packages should be separated into fixes and updates, since multiple fixes can be installed at once and only one update can be installed at once.) Information about each maintenance package is available, such as build release date and install status. The most recent update maintenance package should appear at the top of the list and be selected by default to encourage users to install latest level. User chooses the appropriate update maintenance package to be installed.

UC11B.9: System presents options to install only, create a response file only, or install and create a response file. User takes default action of install only, without creating response file.

UC11B.10: System presents status while dependency checks are performed.

UC11B.11: System presents summary listing maintenance packages to be installed and the offering they will be applied to (offering is designated by offering name and discriminant, such as install directory, since more than one copy of the offering might be installed).

UC11B.12: System presents status while the maintenance package is installed. System automatically proceeds to next step.

UC11B.13: System presents message stating the installation has been completed successfully and presents choice to finish or perform another maintenance activity. User chooses to finish.

UC11 Alternative Flows

UC11A1. User does not have authority to install. System presents message that informs user of authority limitation.

Previous step: UC11B.10.

UC11A1.1: System presents error message to user stating that user does not have authority to run uninstall but can create a response file, which can be silently run by a user with authority.

UC11A1.2: System presents option to create a response file. User can choose response file and perform a silent install when logged in with proper authority.

Next step: UC11B.10.

UC11A2. Maintenance installer is not on the system.

Description: This flow has been removed.

UC11A3. Maintenance installer not capable of installing maintenance package.

Description: This flow is specific case of a missing dependency (see UC11A4), though unique in that the installer itself is the dependency. If possible, the maintenance installer should update itself so that it is capable of installing maintenance packages.

Previous step: UC11B.10.

UC11A3.1: System presents message listing missing dependencies.

Next step: UC11B.10.

UC11A4. Dependency missing for maintenance package install and dependency cannot be installed.

Description: In this case, a dependency for a fix is not met, and cannot be met by the maintenance installer automatically (for example, by installing another fix. In this case, UC11A6 should apply).

Previous step: UC11B.10.

UC11A4.1: System either presents message listing missing dependencies or automatically selects them.

Next step: UC11B.10.

UC11A5. User selects to install fixes, which allows multiple fixes to be selected for install.

Description: This step replaces step UC11B.9. It allows multiple selection of fix maintenance packages, whereas UC11B.9 covers install of a single update maintenance package.

Previous step: UC11B.8.

UC11A5.1: System presents list of installable fixes.

Next step: UC11B.9.

UC11A6. Dependency (prerequisite or corequisite) fix is in same path but not selected for install.

Description: Fix A has a dependency on fix B. If fix B is in the specified maintenance package directory or otherwise appears on the maintenance package list, then it should be automatically selected when fix A is selected, if possible. If this is not possible, users should be informed that they need to manually select the other maintenance package. If fix B is not in the selected directory, then UC11A4 should apply.

Previous step: UC11B.10.

UC11A6.1: Automatically select prerequisite or corequisite fixes if possible. Currently they must be in the same directory, since only one location is scanned for maintenance packages. If multiple locations or automatic scanning for fixes is available, any dependencies should be installed if possible. If dependency cannot be met, then system presents message stating the install cannot be completed and which dependency is required.

Next step: UC11B.10.

UC11A7. System detects that running processes must be stopped (or started) in order for install to be completed.

Description: Installation requires processes to be stopped or started. Users should get a warning before any processes are stopped, giving them the opportunity to manually stop them before proceeding.

Previous step: UC11B.10.

UC11A7.1: System lists detected running processes that need to be stopped or started, and gives users the opportunity to manually stop them before proceeding.

Next step: UC11B.10.

UC11A8. Running processes that were stopped need to be restarted after install is completed.

Description: Processes that have been stopped for install (see UC11A7) can be restarted at end of install. Since the system stopped the services, this step helps user get system back to condition it was in before install.

Previous step: UC11B.13.

UC11A8.1: System provides option to restart running processes that were stopped.

Next step: Finish and exit install.

UC11A9. Display details of offering or offering copy.

Previous step: UC11B.5, UC11A42.1.

UC11A9.1: System presents details about offering copy that user selects.

Next step: Return to previous step.

UC11A10. Install a fix whose only purpose is to change the dependency information registered for an offering.

Description: This flow has been moved to UC51.

UC11A11. Install maintenance package when underlying infrastructure, such as Solution Installation, is not available.

Description: In this flow, the maintenance installer cannot run because the installation infrastructure is not available, and therefore the installer cannot know what can and cannot be installed. Unless there is a way to detect that this has occurred, this situation would be the same as not having any offerings on the system (UC11A54).

UC11A12. Offering fix pack installs maintenance package for shared software that has not been tested with other offerings that share that software, and may ‘break’ other offerings.

Description: If an offering will update shared software, the dependencies for that software should be checked to determine if other offerings that share it can use the newer level. For example: A WAS fix pack updates component X that is used by WPS also. WPS does not list the newer level of component X as a dependency alternative. When installing the WAS fix pack, the user should be informed that WPS may not work with the newer level of component X.

The situation where maintenance package installs cause breakages should be avoided if at all possible. Having an automated way to update the dependencies for all installed offerings is one way to avoid this.

Previous step: UC11B.10.

UC11A12.1: System presents message that fix pack is not tested with the following products that depend on this product, and prompts user to continue or cancel. User continues.

Next step: UC11B.10.

UC11A13. User chooses to view maintenance package details for maintenance package selected.

Description: Details about a maintenance package help users determine which maintenance packages they want to install. Details might include short and long description, list of defects/APARs fixes, build number, size of maintenance package, disk space required to install, link to readme, and any dependencies.

Previous step: UC11B.9 or UC11A5.1.

UC11A13.1: System presents detailed information about the selected maintenance package.

Next step: User returns to maintenance package list and selects maintenance packages to install.

UC11A14. Install new language with maintenance package.

Description: Languages can be provided post-GA as part of maintenance package that contains other files, such as a fix pack. This is not the preferred solution since update maintenance packages are cumulative, so if the language was added it would have to be part of each subsequent update also. The long-term solution where languages are treated as a separate type of maintenance package is shown in UC20.

Previous step: UC11B.9.

UC11A14.1: System presents list of languages that can be added with maintenance package. User selects languages they want to install.

Next step: UC11B.10.

UC11A15. Maintenance package is an integrated maintenance package where some of the fixes included in that package need to be excluded because they have been found to be defective (PE in z/OS terms).

Description: This flow has been removed.

UC11A16. Install of update maintenance package when fixes are installed.

Description: Fixes can be installed prior to fix pack or refresh pack install. These fixes can be newer or older than the update maintenance package. If they need to be uninstalled, they should be automatically re-installed if possible. If not possible, the user should be presented with a list of fixes that will need to be reinstalled between UC11B.10 and UC11B.11.

UC11A17. Install of update when earlier update is installed.

Description: This should be identical to the main flow. Updates, such as fix packs, should be installable on any previous level.

UC11A18. Install of maintenance package that conflicts with another installed maintenance package or maintenance package to be installed.

Description: Fix A has a conflict with fix B. If fix B is in the specified maintenance package directory or otherwise appears on the maintenance package list. If this occurs, Fix B should be automatically deselected when fix A is selected, if possible. If this is not possible, users should be informed that they cannot install the two maintenance packages together as soon as possible.

UC11A19. Install of fixes when fix pack is installed.

Description: This flow depends on whether the fix is later than or earlier than and whether the fix is included or superseded in the update (fix pack or refresh pack). If it is earlier and included in an installed update or superseded by one included in an update, it should not appear in the list in UC11A5.1. If it is later, the fix should be shown as installable in UC11A5.1.

UC11A20. Install of update maintenance package contains optional install features.

Description: Some maintenance packages can contain optional features. Users should be able to select which of the optional features they would like to install.

Previous step: UC11B.10.

UC11A20.1: System presents list of optional features that can be installed. Some optional features might be selected by default. Users select which features they want to install.

Next step: UC11B.10.

UC11A21. Install of maintenance package requires configuration change or additional action.

Description: This flow will be maintenance package dependent. In general, configuration change should be avoided in a maintenance package install stream.

Previous step: UC11B.10.

UC11A21.1: System informs user that configuration (or other) change will occur prior to starting the install. User can continue or cancel install at this point.

Next step: UC11B.11.

UC11A22. Maintenance package has license agreement.

Description: Maintenance package, such as refresh pack, might require user acceptance of a license agreement prior to install.

Previous step: UC11B.10.

UC11A22.1: System presents license agreement and options for user to accept or decline. Users accept agreement to continue. If they do not accept the agreement, install is stopped (UC11A55).

UC11A23. User selects to create a response file, and needs to input file name and directory.

Previous step: 11B.9.

UC11A23.1: System presents a default response file name and directory. Users can override this default by entering a fully qualified file name or, if using a GUI, to browse to a path and enter a file name. If file already exists, users are asked if they want to replace it (UC11A59.1).

Next step: 11B.9.

UC11A24. User installs delta fix pack.

Description: This flow removed.

UC11A25. User installs fix pack when critical fix pack installed.

Description: This flow removed.

UC11A26. Reboot required after maintenance package install.

Description: Users should not be forced to reboot if at all possible. If they do need to reboot, the reboot should replace the last step in installation or occur after the post-install summary. If they are going to be asked to reboot, any post-install options should not be included on the post-install summary.

Previous step: UC11B.13.

UC11A26.1: System presents message that reboot is required and offers options of rebooting now or at a later time.

Next step: Exit and reboot if selected.

UC11A27. Maintenance package includes fixes that are not as new as fixes already installed.

Description: This situation can occur when an updated version of a fix is released.

Previous step: UC11B.10.

UC11A27.1: System presents message stating that the selected packages are older than the ones currently installed and giving the option to either uninstall current ones and replace them with older ones or skip installing the older ones.

Next step: UC11B.10.

UC11A28. Maintenance package for offering that is on multiple machines, or part of a topology.

Description: A maintenance package might affect more than just a single machine, particularly if offering is part of a topology: for example, updating files from base node that are federated into deployment manager on WAS v5. If there is a specific ordering of maintenance installs that needs to occur, users should be informed. For example, with WAS v5, they should be informed that the deployment managers needs to be updated before updating base nodes.

Previous step: UC11B.10.

UC11A28.1: System presents message informing users of actions they need to take in order to update topology.

Next step: UC11B.10.

UC11A29. Maintenance install fails during installation transfer phase.

Description: Maintenance install can fail for a variety of reasons. As many checks as possible should be done prior to starting the install to help ensure it will be successful. When it fails, users should be informed of the failure and actions to take. Both summary and detailed information about the failure should be provided. If using a GUI, users could also be presented with the option to view the log file containing the information about the failure. If users can try the install again without exiting the installer, then that option should be provided.

Previous step: UC11B.12.

UC11A29.1: System presents message that describes the failure and user actions that can be performed to address it.

Next step: UC11B.12.

UC11A30. Install maintenance package when multiple installations of an offering exist.

Description: This flow has been removed.

UC11A31. Install maintenance package that updates selected offering and another offering, such as a dependency.

Description: In this case, applying a single maintenance package to an offering requires an update to another offering. For example, applying maintenance to WebSphere Application Server often requires updating IBM HTTP Server. Users should be provided with a warning that the second offering is going to be updated before the fix is applied.

Previous step: UC11B.10.

UC11A31.1: System presents warning that the second offering will be updated.

Next step: UC11B.10.

UC11A32. Install maintenance package that updates offering and another offering, such as a dependency that will 'break' other offerings using the dependency.

Previous step: UC11B.10.

UC11A32.1: System presents warning about the offering that will be broken.

Next step: UC11B.10.

UC11A33. Updates to an offering that extends another offering.

Description: Some offering editions might replace or extend another edition. For example, an enterprise level extending a base-level product, then user might have to install both base fix pack and enterprise fix pack in a certain order or, if possible, a single fix pack that updates both editions. If the base and enterprise updates are separate maintenance packages, then user should be informed if any specific order of install is necessary and if mixed levels are supported: for example, Does user need to install base fix pack prior to enterprise fix pack? Can system work with base fp1 and enterprise fp2 levels, or does system need to have both base fp2 and enterprise fp2?

Previous step: UC11B.10.

UC11A33.1: If possible, provide a message describing what users need to do.

Next step: UC11B.10.

UC11A34. Install that requires update to underlying infrastructure, such as SI.

Description: Updates to the underlying infrastructure should happen automatically, such as through the SI bootstrap. If this is not instantaneous, a progress bar should be shown. If the update to the infrastructure requires a maintenance package, then users should be told where to get the maintenance package.

UC11A35. Install of maintenance package when some optional features are backlevel.

Description: This flow occurs if offering features are added from the original media after the offering was updated. When the maintenance package that will bring the backlevel features up to date, the system should indicate that the package "Reinstall needed" when users are selecting maintenance to install (UC11B.8 or UC11A5.1).

UC11A36. Maintenance install launched by double-clicking file.

Description: This flow requires a file extension to be registered with the operating system. If the operating system has file or program associations (like Microsoft Windows), then maintenance install can be launched by double-clicking a maintenance package file, which will launch the maintenance install with the maintenance package as a selection.

Previous step: User double-clicks maintenance package file name.

UC11A36.1 (UC11B. 2): System presents progress while preliminary checks and initializations are performed. System automatically proceeds to next step.

UC11A36.2 (UC11B.7): System presents message and progress while checking current system state and generating list of maintenance packages in the selected directory. System automatically proceeds to next step.

UC11A36.3 (UC11B.10): System presents status while dependency checks are performed.

Next step: UC11B.11.

UC11A37. Automatic start or stop of processes fails.

Description: Installation requires processes to be stopped or started and user chooses to do so automatically (see UC11A7 and UC11A8), but it fails.

Previous step: UC11A7.1 or UC11A8.1.

UC11A37.1: System presents message that processes could not be stopped or started automatically.

Next step: System can offer choice to try again or allow user to do it manually.

UC11A38. User has specified an offering and a directory, and no maintenance package is available or installable.

Description: No maintenance packages exist in the directory at all or none exist that are installable for the chosen offering.

Previous step: UC11B.7.

UC11A38.1: System presents message that no maintenance packages are found in the directory to apply to the offering.

Next step: User can exit, or go back and change a previous selection.

UC11A39. Fix pack contains an optional feature that was offered in a previous fix pack and has already been installed.

Description: Since fix packs are cumulative, any optional features should be presented to users each time a fix pack is installed until the user installs the feature. Once the feature is installed, the option should not be surfaced to users, or if it is surfaced it should be selectable but not editable.

Previous step: UC11B.10.

UC11A39.1: System presents optional features.

Next step: UC11B.10.

UC11A40. User launches maintenance installation using a shortcut within an application that specifies the discriminant.

Description: This alternate flow is identical to the main flow, except the user does not have to specify the offering (skip UC11B.5).

Previous step: User clicks the link, menu item, or other shortcut that launches the maintenance installer.

UC11A40.1 (UC11B. 2): System presents progress while preliminary checks and initializations are performed. System automatically proceeds to next step.

UC11A40.2 (UC11B.3): System presents welcome message for maintenance activity.

UC11A40.3 (UC11B.4): System presents options to install or uninstall maintenance packages..

Next step: UC11B.6.

UC11A41. User selects maintenance path without first selecting offering via a command line (maintenance first path).

Description: This flow has been removed.

UC11A42. User launches a maintenance installation by double-clicking a maintenance package file and maintenance package can apply to multiple offerings.

Description: Maintenance packages might be installable to more than one offering copy. If user chooses to install WAS fix pack 2 and there are 2 copies of WAS, one at GA level and one at fix pack 1 level, then both copies are valid offerings to install fix pack 2. Only valid offerings should be presented. For example, if the maintenance package is fix pack 3 for DB2 and there are 2 copies of DB2, one at fix pack level 2 and one at fix pack level 4, only the copy at fix pack level 2 is valid to install fix pack 3.

Previous step: UC11A36.3.

UC11A42.1: System presents the offerings that the maintenance packages can be applied to. User chooses offering.

Next step: UC11A36.3.

UC11A43. User launches a maintenance installation by double-clicking a maintenance package file and maintenance package that is already installed to all appropriate offerings.

Description: User selects maintenance first path, and selected maintenance packages are already installed on any applicable offerings on the system.

Previous step: UC11A36.3 or UC11A42.1.

UC11A43.1: System presents message that maintenance package is already installed and confirms that users want to reinstall it.

Next step: User can continue and reinstall, or go back and change previous selections.

UC11A44. User launches a maintenance installation by double-clicking a maintenance package file and the maintenance package is not applicable or installable to any offerings on the system.

Previous step: UC11A36.3.

UC11A44.1: System presents message that selected maintenance package cannot be installed because no applicable offerings were found on the system.

Next step: Exit.

UC11A45. User types command line with maintenance package as a parameter.

This flow has been removed

UC11A46. User runs maintenance install when only one copy of one offering is installed on the system.

Description: This flow has been removed. This flow should be identical to the main flow.

UC11A47. Dependency needed, and can be installed.

Description: Dependency for a maintenance package is missing but is included in maintenance package and is installable. Since dependency can be installed, it is noted in UC11B.10. Otherwise, the flow is same as basic flow.

Previous step: UC11B.10.

UC11A48. Dependency needed but cannot be installed.

Description: This flow has been removed and is identical to UC11A4.

UC11A49. Dependency installed, at an earlier level than required, although can be upgraded.

Description: Correct level of dependency for a maintenance package is missing, although an earlier version is installed. The maintenance package contains the newer level and can install it. Since dependency can be installed, user is given a choice to install the dependency if the install will overwrite the earlier version.

Previous step: UC11B.10.

UC11A49.1: System presents message that dependency is missing.

Next step: UC11B.10

UC11A50. Maintenance package updates Eclipse plug-in.

Description: This flow should be identical to the main flow for users.

UC11A51. Install of a maintenance package that is platform dependent on the wrong platform.

Description: Some maintenance packages might be platform dependent. Preferably, these packages should not be shown in the maintenance list at all. If not, then this case should be treated as a dependency failure (UC11A4)

UC11A52. User chooses a maintenance package location path that contains no files that are maintenance packages.

Description: User selects a path that does not contain maintenance packages.

Previous step: UC11B.7.

UC11A52.1: System presents message that no maintenance packages are available in this path.

Next step: UC11B.7: User chooses another location.

UC11A53. User chooses a local maintenance location path that does not exist.

Description: In most cases, this will occur when a user enters the wrong path or selects a path from the path list that no longer exists.

Previous step: UC11B.7.

UC11A53.1: System presents message that location does not exist.

Next step: UC11B.7: User chooses another location.

UC11A54. No offerings on system that maintenance installer can apply maintenance to.

Description: User tries to run maintenance installer when offerings have been uninstalled.

Previous step: UC11B.2.

UC11A54.1: System presents message that no offerings were found to apply maintenance to. This message should appear as early as possible in the flow.

Next step: User can exit, or go back and change a previous selection.

UC11A55. User does not accept license agreement.

Description: Install cannot continue since user hasn't accepted the license agreement.

Previous step: User chooses "I do not accept..." in UC11A22.1.

UC11A55.1: System presents message stating that accepting the license is required to install the product.

Next step: Return to previous step.

UC11A56. Install path for response file exceeds operating system limit for path characters.

Description: User types fully qualified path or file name (or both) that is too long for operating system. Calculation should be based on real operating system path, not variable.

Previous step: UC11B.9.

UC11A56.1: System presents message that the path (repeat the path back to user) is too long and denotes the maximum path length for the operating system and the entered path length.

Next step: UC11B.9.

UC11A57. Install of a maintenance package when only part of an offering (one or few features) is installed.

Description: Maintenance package install should be successful regardless of number of features selected to install. It is possible that the maintenance package will have nothing to update if the applicable features are not installed. In this case, the user should be presented with a message stating that the package will not be installed because the applicable features are not present.

UC11A58. User chooses path and offering, and all of the maintenance packages are already installed for that combination.

Description: If maintenance packages are already installed, the status shown in step UC11B.8 should reflect the status of the maintenance packages. If users proceed beyond this step, the maintenance packages should be reinstalled.

UC11A59. User enters response file name to create that already exists.

Description: This alternate flow occurs when the user chooses to create a response file with a name that already exists. Users can overwrite the existing file or create a new one by entering a new name.

Previous step: UC11B.9.

UC11A59.1: System presents message that file already exists, and offers choices to overwrite or not.

Next step: If user chooses to overwrite file, continue to next step. If user does not overwrite file, return to previous step (UC11B.9), where user can change the name of the file. This flow should be repeated until the user either clicks Yes to overwrite the file name or enters a new filename.

UC11A60. User launches maintenance install command line with discriminant and maintenance package file name as parameters.

Description: This flow has been removed.

UC11A61. User launches maintenance install using a command line with offering as a parameter.

Description: This flow has been removed.

UC11A62. User launches maintenance install using a command line with offering and the discriminant as parameters.

Description: This flow has been removed.

UC11A63. User launches maintenance install using a command line with offering and the maintenance package file name as parameters.

Description: This flow has been removed.

UC11A64. User launches maintenance install using a command line with offering, discriminant and the maintenance package file name as parameters.

Description: This flow has been removed.

UC11A65. User launches maintenance install using a command line with offering, discriminant or maintenance package file name as parameters, and one of these parameter names is not valid.

Description: This flow has been removed.

UC11A66. User specifies a response file path or name with invalid characters.

Description: Occurs when invalid characters entered or user specifies a string that cannot be a valid path.

Previous step: User enters a path with invalid characters, most likely when specifying the response file (UC11B.9).

UC11A66.1: System presents error message indicating that entry is not a valid path.

Next step: Return to previous step.

UC11A67. User specifies a remote path, and search for path times out because host cannot be reached.

Description: In a path field, user enters a remote path but connection times out.

Previous step: User enters a path and path is not found most likely when specifying the response file (UC11B.9) or choosing the maintenance location (UC11B.6).

UC11A67.1: System presents error message stating that path could not be reached.

Next Step: Return to previous step.

UC11A68. User specifies a remote path and host is found, but path cannot be found.

Description: User enters a path that does not exist on a server that can be reached. If the user is creating a file (for example, a response file), that path should be created, so this flow should not apply.

Previous step: User enters a path and path is not found, when choosing the maintenance location (UC11B.6).

UC11A68.1: System presents error message stating that path/address could not be found.

Next Step: Return to previous step.

UC11A69. An instance of the maintenance installer is launched when either another instance is running or an installer is running that uses the same installation database or other resources.

Description: This should only be an issue if there is no method of locking the shared resources. Once an installer is executing a change, it should block any other change management operations from being initiated by other installers. If this is not possible, then users should be presented with a message for each installer opened after the first one that tells them that only one can run at a time.

UC11A70. User cancels maintenance install prior to file copy.

Description: When users cancel, there should be a confirmation message confirming that they want to cancel and having an option to return to the installation. Any changes made while the installation was running should be undone.

UC11A71. Users cancel maintenance install during file copy.

Description: When users cancel, there should be a confirmation message confirming that they want to cancel and having an option to return to the installation. Any changes made while the installation was running should be undone.

UC11A72. One or more errors that prevent a maintenance package from being installed are found during dependency checking, but one or more other valid maintenance package can still be installed.

Description: This flow occurs after one or more dependency failures occur, but installation can still proceed. If none of the maintenance packages can be installed, then UC11A73 should apply.

Previous step: One or more dependency failures that prevents one or more maintenance package from being installed (for example, UC11A4.1, UC11A28, and so on.

UC11A72.1: System presents message TBD.

Next step: UC11B.10.

UC11A73. One or more errors that prevent a maintenance package from being installed are found during dependency checking, and there are no remaining valid maintenance packages to be installed.

Description: This flow occurs after one or more dependency failures occur and installation cannot proceed. If one or more of the maintenance packages can be installed, then UC11A72 should apply.

Previous step: One or more dependency failures that prevents one or more maintenance package from being installed (for example, UC11A4.1, UC11A28, and so on.

UC11A73.1: System presents message TBD.

Next step: Exit.

UC11A74. User types a command line where the discriminant and offering name specify two different products, but do not specify a maintenance package.

Description: This flow has been removed.

UC11A75. User types a command line where the discriminant or offering specifies a different product than the maintenance package applies to.

Description: This flow has been removed.

UC11A76. User enters a command line where the maintenance location path is invalid or does not exist.

Description: This flow has been removed.

UC11A77. User enters a command line where the maintenance file name is invalid or does not exist.

Description: This flow has been removed.

UC11A78. Install of maintenance package that has optional features and not all of them are installed.

Description: This flow occurs if a maintenance package with optional features is installed (UC11A20) and the user does not install all of the features. In this case, the package should maintain a state of partially installable.

UC11A79.

Description: This flow has been removed.

UC11A80.

Description: This flow has been removed.

UC11A81. User enters a URL that specifies the location of maintenance packages during the maintenance installer flow.

Description: In this case, the user specifies a URL that contains the maintenance packages during the maintenance installer flow. The assumption is that the appropriate code to make this work is integrated either client or server-side. If the path cannot be reached, either UC11A67 or UC11A68 should apply.

Previous step: User types a URL in step UC11B.6.

UC11A81.1: System displays progress while it attempts to connect to the site.

Next step: UC11B.7.

UC11A82. User launches dependency check for maintenance package without completing an install.

Description: In this flow, user types in a command line command that specifies the maintenance package, and either (or both) the discriminant offering name and an additional parameter to indicate it is a dependency check.

Previous step: User selects option to run dependency check only.

UC11A81.1: System presents progress while dependency checks are performed.

UC11A81.2: System presents message showing results of dependency check.

UC11A83. Maintenance installer allows option to rerun maintenance installer.

Previous step: The user is presented with the option to run the wizard again in the last step of the main flow (UC11B.13). This assumes that there are no actions that need to be taken before the maintenance installer can be run again (e.g., reboot, configuration, and so on). When the user selects this option, the maintenance installer should run again.

Next step: UC11B.4.

UC11A84. User creates a response file without performing a maintenance installation.

Description: In this flow, the user is not installing maintenance, but is only creating a response file. The dependency check should be skipped in this case.

Previous step: User only selects the option to create a response file (and does not select the option to install) in UC11B.9.

UC11A84.1: System presents pre-installation summary of items to be included in response file.

UC11A84.2: System presents post-install summary of items added to response file.

Next step: Exit or run wizard again, depending on user choice.

UC11A85. Maintenance package found during maintenance package query is uninstallable because the package has been corrupted.

Description: Users should be warned about this either by a warning message or by displaying the package in step UC11B.8, but disabling it and representing it as uninstallable.

3.2.2.2 UC19: Interactive roll back maintenance packages

Brief Description

Roll back of a maintenance package that has been applied to one offering.

Actor

1. User

UC19 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. User logs on to system using an ID that has authority to perform roll back.
2. Maintenance installer is installed, or otherwise available for use.
3. Basic flow assumes that only one update maintenance package can be rolled back at one time. Fix maintenance packages might require roll back of multiple fixes at one time, such as corequisite fixes.
4. Installed maintenance package for offering can be determined by maintenance installer.

UC19B.0: User initiates maintenance roll back. System automatically proceeds to next step.

UC19B.1 (UC11B.1.): System presents a list of languages available for use during the install program. (The product language files will be chosen later.) Language selection should default to the user locale, if applicable. User chooses language for the install.

UC19B.2 (UC11B.2.): System presents progress while preliminary checks and initialization are performed. This step may involve checks and progress may need to be presented in several substeps. System automatically proceeds to next step.

UC19B.3 (UC11B.3.): System presents welcome message for maintenance package activity.

UC19B.4: System presents options to install or roll back maintenance package. User chooses roll back maintenance package.

UC19B.5: System presents list of offerings that have maintenance packages installed. User selects an offering.

UC19B.6: System presents message and progress bar while checking current system state and generating list of maintenance packages that can be rolled back. System automatically proceeds to next step.

UC19B.7: System presents list of maintenance packages that can be rolled back for chosen offering. (This list should appear similar to list of installable maintenance packages.) Information about each maintenance package is available, such as install date, build date and install status. User chooses the appropriate update maintenance package to roll back. UC19B.8: System presents options to roll back only, create a response file, or roll back and create a response file. User takes default action of roll back, without creating a response file.

UC19B.8: System presents options to roll back only, create a response file, or roll back and create a response file. User takes default action of roll back, without creating a response file.

UC19B.9: System presents status while dependency checks are performed.

UC19B.10: System presents summary list of maintenance packages to be rolled back and the offering that they will be rolled back from. Offering is designated by offering name and install directory since more than one copy of the offering might be installed.

UC19B.11: System presents status while the maintenance package is rolled back. System automatically proceeds to next step.

UC19B.12: System presents message stating the roll back was completed successfully and presents the choice to finish or perform another maintenance activity. User chooses to finish.

UC19 Alternative Flows

UC19A1. User does not have authority to roll back. System presents message that informs user of authority limitation.

Previous step: UC19B.9.

UC19A1.1: System presents error message to user stating that user does not have authority to run roll back, but can create a response file, which can be silently run by a user with authority.

UC19A1.2: System presents option to create a response file. User can choose response file and perform a silent roll back when logged in with proper authority.

Next step: UC19B.9.

UC19A2. Maintenance installer not on the system.

Description: This flow has been removed.

UC19A3. Maintenance installer not capable of rolling back maintenance package due to dependency.

Description: If the maintenance install could install a maintenance package, it should be able to roll back the maintenance package. This flow can occur only if a back-level maintenance installer is somehow installed after the maintenance package has been installed.

Previous step: UC19B.9.

UC19A3.1: System presents error message to user stating that maintenance package cannot be rolled back.

Next step: UC19B.9.

UC19A4. User wants to roll back multiple fixes.

Description: This step is the same as UC19B.7, except user can choose more than one fix to roll back whereas UC19B.7 covers roll back of single fix pack maintenance package.

UC19A5. Corequisite fix is not selected for roll back.

Description: This flow should be handled like UC11A6. The corequisite fix should also be selected for roll back.

Previous step: UC19B.9.

UC19A5.1: Automatically select prerequisite or corequisite fixes if possible. If multiple locations or automatic scanning for fixes is available, any dependencies should be rolled back if possible. If dependency cannot be met, then system presents message stating the roll back cannot be completed and which dependency is required to also be rolled back.

Next step: UC19B.9.

UC19A6. Processes must be started or stopped in order for roll back to complete.

Description: Installation requires processes to be stopped or started. Users should get a warning before any processes are stopped, giving them the opportunity to manually stop them before proceeding.

Previous step: UC19B.9.

UC19A6.1: System lists detected processes that need to be stopped or started, and gives users the opportunity to manually stop them before proceeding.

Next step: UC19B.9.

UC19A7. User wants to restart or stop running services that were started or stopped after roll back is complete.

Description: Processes that have been stopped for install (see UC19A6) can be restarted at end of install. Since the system stopped the services, this step helps user get system back to condition it was in before install.

Previous step: UC19B.12

UC11A8.1: System provides option to restart running processes that were stopped.

Next step: Finish and exit install.

UC19A8. Display details of offering or offering copy.

Previous step: UC19B.5.

UC11A9.1: System presents details about offering copy that user selects.

Next step: Return to previous step.

UC19A9. Roll back of shared software, and may 'break' other offerings.

Previous step: UC19B.9.

UC11A12.1: System presents message that fix pack is not tested with the following products that depend on this product, and prompts user to continue or cancel. User continues.

Next step: UC19B.9.

UC19A10. User chooses to view maintenance package details for maintenance package selected.

Description: Details about a maintenance package help users determine which maintenance packages they want to install. Details might include short and long description, list of defects/APARs fixes, build number, size of maintenance package, disk space required to install, link to readme, and any dependencies.

Previous step: UC19B.8 or UC19A4.1.

UC11A13.1: System presents detailed information about the selected maintenance package.

Next step: User returns to maintenance package list and selects maintenance packages to install.

UC19A11. Uninstall of a language with maintenance package.

Description: Languages might be provided post-GA as part of a maintenance package. See UC11A14 for more information. The entire maintenance package must be uninstalled; the language or other optional features cannot be removed independently.

UC19A12. Roll back of maintenance package that is an integrated maintenance package where some of the fixes included in that package need to be excluded because they have been found to be defective (PE in z/OS terms).

Description: This flow has been removed.

UC19A13. Roll back of fix pack when fixes are installed post fix-pack install.

Description: The fix pack and any fixes that were installed after the fix pack should be rolled back. This flow is the same as the main flow; each maintenance package to be rolled back should be listed in summary.

UC19A14. Roll back of fix pack when earlier fix pack is installed.

Description: If an earlier fix pack is installed, the roll back should roll back to that level. This flow is the same as the main flow.

UC19A15. Roll back of fixes when fix pack is installed.

Description: Only fixes newer than and installed after the fix pack can be rolled back. Fixes that have been superseded cannot be rolled back until the superseding fix is removed. This flow is the same as the main flow.

UC19A16. Roll back of update maintenance package contains optional install features.

Description: Any optional features that were installed with the maintenance package should be rolled back with the maintenance package. These optional features can be listed on summary UC19B.9.

UC19A17. Roll back of maintenance package requires configuration roll back or additional action.

Description: This flow will be maintenance package dependent. In general, configuration changes should be avoided in maintenance streams. Any configuration steps would need to occur prior to roll back, since maintenance package code could be needed to do configuration roll back.

UC19A18. User selects to create a response file, and needs to input file name and directory.

Previous step: UC19B.6

UC19A18.1: System presents a default response filename and directory. Users can override this default by entering a fully qualified filename, or if using a GUI to browse to a file. If file already exists, user is asked if they want to replace it (UC19A29).

Next step: UC19B.6.

UC19A19. User roll backs delta fix pack.

Description: This flow should be identical to the main flow, since only the most recent fix pack should be able to be rolled back. Since the fix pack was delta, the previous level is restored.

UC19A20. Reboot required after maintenance package roll back.

Description: Users should not be forced to reboot if at all possible. If they do need to reboot, the reboot should replace the last step in installation or occur after the post-install summary. If they are going to be asked to reboot, any post-install options should not be included on the post-install summary.

Previous step: UC19B.6.

UC19A20.1: System presents message that reboot is required and offers options of rebooting now or at a later time.

Next step: Exit and reboot if selected.

UC19A21. Roll back of maintenance package for offering that is on multiple machines, or part of a topology.

Description: See UC11A28. Roll back should be reverse of install action.

UC19A22. Maintenance package roll back fails.

Description: Maintenance package roll back can fail for a variety of reasons. As many checks as possible should be done prior to starting the roll back, to help ensure it will be successful. When it fails, users should be informed of the failure and actions to take. Both summary and detailed information about the failure should be provided. If using GUI, users could also be presented an option to view the log file containing the information about the failure. If the user can try roll back again without exiting, then that option should be provided.

Previous step: UC19B.11.

UC19A22.1: System presents message that describes the failure and user actions that can be performed to address it.

Next step: Exit.

UC19A23. Roll back of maintenance package when multiple copies of the offering exist.

Description: This flow is same as UC19B.5 since all offerings and offering copies should appear in the same list.

UC19A24. Maintenance roll back launched by command line with maintenance package file as a parameter.

Description: This flow has been removed.

UC19A25. Installer has performed a custom action that cannot be undone.

Description: Actions that are performed as part of a maintenance package install should be able to be undone. If they are not, users should be clearly informed how to get their system back to the state it was in prior to the maintenance package install.

Previous step: UC19B.9.

UC19A3.1: System presents error message to user stating that action cannot be undone.

Next step: UC19B.9.

UC19A26. User specifies a response file name with path or network address, and search for address times out because host cannot be reached.

Description: In a path field, user enters a remote path, but connection times out.

Previous step: User enters a path and path is not found, most likely when specifying the response file (UC19B.8)

UC9A26.1: System presents error message stating that path could not be reached.

Next step: Return to previous step.

UC19A27. User tries to run multiple installs or roll backs at the same time.

Description: This flow is covered by UC11A69.

UC19A28. User specifies a response file path with invalid characters.

Description: Occurs when invalid characters are entered or user specifies a string that cannot be a valid path.

Previous step: User enters a path with invalid characters.

UC19A28.1: System presents error message indicating that entry is not a valid path.

Next step: Return to previous step.

UC19A29. User enters response file name to create that already exists.

Description: This alternate flow occurs when the user chooses to create a response file with a name that already exists. Users can overwrite the existing file or create a new one by entering a new name.

Previous step: UC19B.8.

UC19A29.1: System presents message that file already exists, and offers choices to overwrite or create a new one.

Next step: Return to previous step. If user does not overwrite file, return to previous step (UC19B.8), where user can change the name of the file. This flow should be repeated until the user either clicks Yes to overwrite the file name or enters a new filename.

UC19A30. User users option to rerun maintenance installer.

Previous step. The user is presented with the option to run the wizard again in the last step of the main flow (UC19B.12). This assumes that there are no actions that need to be taken before the maintenance installer can be run again (e.g., reboot, configuration, and so on). When the user selects this option, the maintenance installer should run again.

Next step: UC11B.4.

UC19A31. User creates a response file without performing a maintenance installation.

Description: In this flow, the user is not installing maintenance, but is only creating a response file. The dependency check should be skipped in this case.

Previous step: User only selects the option to create a response file (and does not select the option to install) in UC19B.8.

UC19A31.1: System presents pre-installation summary of items to be included in response file.

UC19A31.2: System presents post-install summary of items added to response file.

Next step: Exit or run wizard again, depending on user choice.

3.2.2.3 UC51: Update metadata

Brief Description

Updates are made to metadata after product has shipped.

Actors

1. User

UC51 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. Product installer logs on to system using an ID that has authority to perform install.
2. User has installed offering.
3. The maintenance package contains updates to metadata and is packaged as a fix.
4. Maintenance package is readable by maintenance installer.
5. There are no dependency issues or errors.

UC51B.0: This flow starts when users select a metadata update packaged as a fix in UC11A5.1.

UC51B.1: System presents list of other offerings on system that will be updated.

Next step: UC11B.10.

UC51 Alternative Flows

UC51A1. Update metadata with a fix after product is installed and update fails.

Previous step: UC11B.12.

UC51A1.1: System presents message stating that installation failed and returns system to previous state.

Next step: Exit.

UC51A2. Uninstall metadata after product is installed using a fix.

Description: This flow is identical to uninstalling a fix (UC19A4).

UC51A3. Uninstall metadata after product is installed using a fix and update fails.

Previous step: UC19B.9.

UC51A3: System presents message stating that installation failed and returns system to previous state.

Next step: Exit.

UC51A4. Manually update metadata to be used during product installation.

Description: In this flow, users are pointing to a new metadata file during installation. We would expect this to occur via a command line such as `install.exe -metadata <filename.mdv>`, then follow the UC2 basic flow.

This flow was formerly UC2A9.

UC51A5. Manually update metadata during installation for a newly supported operating system.

UC51A6. Manually updating metadata during installation fails.

Description: System presents message stating that installation failed and returns system to previous state.

Next step: Exit.

UC51A7. Update metadata during uninstall.

UC51A8. Update metadata in database directly using a command line.

UC51A9. Roll back metadata in database directly using a command line.

3.2.2.4 UC4: Install bundled maintenance

Brief Description

Customer installs maintenance that bundles maintenance packages for multiple products made available after GA.

Actors

1. User

UC4 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. Product installer logs on to system using an ID that has authority to perform install.
2. User has installed offering.
3. Maintenance package is readable by maintenance installer.
4. There are no dependency issues or errors.

UC4B.0: This flow starts when users select a bundled maintenance package in step UC11B.8.

UC4B.1: After the dependency check step (UC11B.8), the system presents list of other offerings on system that will be updated.

Next step: UC11B.10.

UC4 Alternative Flows

UC4A1. Multiple copies of an offering are installed, and users need to choose which one to update.

This alternate flow is replaced by UC11A42.

UC4A2. User installs bundled maintenance without filtering maintenance packages by the offering.

Description: This flow will be defined later.

3.2.2.5 UC17: Roll back bundled maintenance

Brief Description

Customer uninstalls maintenance that bundles maintenance packages for multiple products made available after GA.

Actor

1. User

U17 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. Product Installer logs on to system using an ID that has authority to perform install.
2. User has installed offering and wants to install bundled maintenance package.
3. Maintenance package is readable by maintenance installer.
4. There are no dependency issues or errors.

UC17B.0: This flow starts when users select a bundled maintenance package in step UC19B.7.

UC17B.1: After the dependency check step (UC19B.9), the user views list of other offerings on system affected by uninstall of the bundled maintenance package and then proceeds to next step.

Next Panel: UC19B.9.

U17 Alternate Flows

3.2.2.6 UC20: Install language pack

Brief Description

Customer installs localization pack made available after GA.

Note that in this use case, languages are delivered by means of localization packs rather than a part of maintenance package (UC11A22).

Note: Localization packs will be referred to as languages to users.

Actor

1. User

UC20 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

Preconditions

1. Product installer logs on to system using an ID that has authority to perform install.
2. User has installed offering and is adding a localization pack, or user wants to run install in new language supported by localization pack.
3. Localization pack provided to user as separate delivery vehicle, not as part of an update (such as a fix pack).
4. Updates after post-GA localization packs contain updates to all languages.

UC20B.0: This flow begins when a user selects a localization maintenance package in UC11B.9. This flow assumes that it is possible to install a language without a fix pack.

UC29B.1: System presents list of installable languages. User chooses language.

Next step: Back to UC11 main flow at UC11B.10.

UC20 Alternative Flows

UC20A1. User has installed fixes prior to localization pack install.

Previous step: User selects to install localization pack in UC20B.1, which requires fixes to be reinstalled.

UC20A1.1: System presents message with a list of fixes and fix packs that will need to be reinstalled.

Next step: UC11B10.

UC20A2. User has installed fix packs prior to localization pack install.

Previous step: User selects to install localization pack in UC20B.1, which requires a fix pack to be reinstalled.

UC20A2.1: System presents message with a list of fixes and fix packs that will need to be reinstalled.

Next step: UC11B10.

UC20A3. User chooses to run install in a newly supported language.

Description: This flow will be added later.

UC20A4. Localization pack install fails.

Previous step: Localization pack selected prior to UC20B.1 install fails for various reasons.

UC20A4.1: System presents error message telling user that install failed.

Next step: Exit.

3.2.2.7 UC81: Uninstall post-GA localization pack

Brief Description

Customer uninstalls localization pack made available after GA.

Note: Localization packs will be referred to as languages to the user.

Actor

1. User

UC81 Basic Flow

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next or pressing Enter.).

Preconditions

1. Product installer logs on to system using an ID that has authority to perform uninstall.
2. User chooses offering.
3. User has installed offering and added localization pack.
4. Localization pack provided to user as separate delivery vehicle, not as part of an update (such as a fix pack).

UC81B.0: This flow begins when a user selects a localization maintenance package in UC19B.8. This flow assumes that it is possible to install a language without a fix pack.

UC81B.1: System presents list of languages that can be uninstalled. User chooses language.

Next step: Back to UC19 main flow at UC19B.9.

UC81 Alternative Flows

UC81A1. Localization pack uninstall fails.

Previous step: Localization pack uninstall fails during UC19B.10 for a variety of reasons.

UC81A1.1: System presents error message telling users that uninstall failed.

Postcondition: Exit.

3.2.2.8 UC13: Install manufacturing refresh

To users this flow is identical to UC2.

3.2.3 Post-Install Use Cases

3.2.3.1 UC5: Post-install migration

Brief Description

Customer runs or reruns migration after install.

Actor

1. User

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

UC5 Basic Flow

Preconditions

1. Product installer logs on to system using an ID that has authority to perform install.
2. Migration conditions are in a deployment descriptor.
3. Previous version of offering that can be migrated is installed on the machine. Previous version could be N-1 or N-x.

Note: For the same version, lower editions (base or standard) might require migration to higher edition (enterprise), but this should be avoided. Offering data should be in a format that can be used by all editions.

4. User has installed new version of the offering. Data that can be migrated from the previous version exists (possibly as backup during upgrade install, or currently on the machine in coexistence scenario).

Note: These flows can be performed to new offering levels that are upgrades (which replace the previous version) or migration that allows coexistence of multiple offering levels (and does not replace the previous version).

5. Each product will have different needs for information to be migrated and whether they support post-install migration or not.

UC5B1: After install is completed, user runs migration tool that reads from the deployment descriptor to complete the migration.

UC5B2: System presents migration progress and messages. System automatically proceeds to next step.

UC5B3: System presents message that migration was successful.

Alternative Flows

UC5A1. Backup of previous version fails (which could have happened during running of migration tool or during offering install; see UC2).

UC5A1.1. System presents message describing reason for failure.

UC5A2. Restore fails.

UC5A2.1. System presents message describing reason for failure.

UC5A3. Offering is higher edition (enterprise) than previous version (base or standard).

UC5A4. Offering is at lower edition (base) than previous version (enterprise).

UC5A5. Migrated data not appropriate for newer level. (for example, spec change between versions.)

UC5A6. Offering is coexisting with another copy or version, so that migration data might cause conflicts.

UC5A7. Migration can be done from several previous versions (such as v3 and v4 to v5, or 2 copies of v4 to v5).

UC5A8. User offered option to select what data to migrate.

UC5A9. User informed that manual processing is needed after migration (for example, obsolete APIs that need replacement.).

UC5A10. User migrates data from other machine.

UC5A11. User migrates data from competitive product.

UC5A12. User must select from several options to migrate from, such as multiple instances that can be migrated.

3.2.3.2 UC6: Post-install configuration

Brief Description

Customer reruns configuration after install.

Actor

1. User

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next or pressing Enter).

UC6 Basic Flow

Preconditions

1. Product Installer logs on to system using an ID that has authority to perform install.
2. Configuration conditions are in the deployment descriptor.
3. User has installed version of the offering.

UC6B.1: After install is completed, user runs configuration tool that reads from the deployment descriptor to complete configuration.

UC6B.2: System presents warning that existing configuration will be replaced. User accepts.

UC6B.3: System presents configuration progress and messages. System automatically proceeds to next step.

UC6B.4: System presents message that configuration was successful.

Note: User could have a number of ways to rerun configuration, such as from a configuration wizard or administrative console. Whatever configuration tool the offering has should allow the user to restore the default configuration from the deployment descriptors, which is the key purpose of this use case. Note that the deployment descriptors might be updated after GA of the product, providing different configuration options than were available at GA.

UC6 Alternative Flows

UC6A1. System presents several options to configure, such as multiple instances that can be configured.

Previous step: UC6B.2.

Next step: UC6B.3.

UC6A2. Configuration fails.

Previous step: UC6B.3

Next step: Exit.

3.2.3.3 UC7: Post-install self-test

Brief Description

Customer runs or reruns self-test after install.

Actor

1. User

UC7 Basic Flow

Preconditions

1. Product installer logs on to system using an ID that has authority to perform install.
2. Self-test conditions are in the deployment descriptor.

UC7B.1: User installs offering.

UC7B.2: After install is completed, user runs self-test, either from First Steps or from the command line.

UC7B.3: System presents self-test progress and messages. System automatically proceeds to next step.

UC7B.4: System presents message that self-test was successful.

Alternative Flows

UC7A1. Self-test could not be performed.

UC71A1.1: System presents message describing error.

UC7A2. Self-test fails.

UC7A2.1: System presents message describing reason for failure.

UC7A3. User selects options for self-test.

Previous step: UC7B.2.

Next step: UC7B.3.

3.2.4 Launchpad and First Steps Use Cases

3.2.4.1 UC21: Check offering dependencies

Brief Description

Allows user to check dependencies of an offering.

Install programs check dependencies for the offering they install. There are other times that checking dependencies can be useful. Users can check dependencies prior to installing an offering. Users might also want to check dependencies after the offering has been installed for a period of time to confirm that all the dependencies are still ok.

Actor

1. User

Unless otherwise noted, a user action initiates an advance to the next step in the flow (for example, by clicking Next, pressing Enter, and so on.).

UC21 Basic Flow

Preconditions

UC21B.0: User launches dependency checker from Setup Launchpad or command line.

UC21B.1: System displays status while dependency check is performed.

UC21B.2: System displays results.

UC21 Alternative Flows

UC21A1. Dependency checker could not be started.

UC53A1.1: System presents message that informs users why dependency checker could not be started and any actions they need to take so that it will run.

Next step: Exit.

UC21A2. Dependency checker could not be started.

UC53A1.1: System presents message that informs users why dependency checker could not be started and any actions they need to take so that it will run.

Next step: Exit.

3.2.4.2 UC53: Start Launchpad

Brief Description

Setup Launchpad is a pre-install tool that links to key pre-install information and actions.

A Setup Launchpad (also referred to as LaunchPad) tool provides the user access to important pre-install information and actions. Setup Launchpad options should be accessible and executable in other ways, such as through commands. This tool simply provides a centralized point to access key pre-install options.

Depending on the product media and target platform, Launchpad can be automatically started, such as on insertion of the offering CD on Windows systems. The Launchpad executable should be in the root directory of the media, so that users can easily find it.

Setup LaunchPad can include these options, and others:

- **Readme and release notes.** A readme and release notes provide key offering information, including late-breaking information that could not be incorporated into the product information prior to GA. These documents are generally provided in text or HTML format, and require a text viewer/editor or browser.
- **Offering documentation, especially installation related documentation** Install documentation, such as an installation guide or getting started, provides users with documentation instructing them how to plan, install and get started using the offering. Documentation can be provided in many different ways or formats, such as HTML, text or PDF. The install media should include some documentation or, at a minimum, a link to install documentation from the readme and Launchpad. Including links to (potentially more up-to-date) information on the Web is good idea. Which ones should be required? See install requirements.
- **Offering information, such as Getting Started or Quick Tour.** This might include a link to the product Web site or an interactive demo of the product, such as one pointing out what's new with this version.
- **Dependency information.** Each offering should have a Web site that contains the latest information about its dependencies. Where appropriate, these dependencies should be separated by platform so that users can quickly go to the platform of interest to them. Specific versions of dependencies should be stated. If the dependency is included with the offering package, it should be clearly stated.
- **Check dependencies.** Offerings should allow users to check dependencies prior to running the install to confirm that the system is prepared for install. Dependency checks could also be run after install to verify that all the dependencies are still present. For more information on check dependencies, see UC21.
- **Topology information**
- **Migration Information**
- **Link to First Steps**
- **Choose optional/additional components**
- **Install offering.** This initiates the offering install. For more information on offering install, see UC2.
- **Exit.** Exits the Launchpad.

Actor

1. User

UC53 Basic Flow

Preconditions

1. User has access to install media, either physical or electronic.

UC53B.1: Setup Launchpad is started, either from autorun or user execution of program such as launchpad.bat or launchpad.sh.

UC53B.2: System presents a list of languages available for use by the Launchpad. Language selection should default to locale, if applicable. User selects language for the Launchpad.

UC53B.3: Launchpad provides important pre-install options for offering being installed. These options might include view readme (UC73), view and check dependencies (UC71 and UC21), view install guide (UC75), or install product (UC2).

UC53B.4: User selects one or more options from the Launchpad. System performs option.

UC53B.5: User closes Launchpad by choosing Exit.

Alternative Flows

UC53A1. Launchpad could not be started.

UC53A1.1: System presents message that informs users why Launchpad could not be started and any actions they need to take so that it will run.

Next step: Exit.

UC53A2. One or more Launchpad options could not be performed.

Description: LaunchPad options vary and might require a browser or text viewer or execute a command. Therefore, if an option fails, the message may vary. For example, if the option links to a URL that is not found, the user will get a typical browser message for page not found.

UC53A2.1: System presents message that informs users why option could not be performed. If possible, include any actions they need to take.

UC53A3. User can launch more than one install from single Launchpad.

Description: Depending on how offerings are packaged, the Launchpad can start installs of more than one offering. Each separate install should appear as its own link on the Launchpad.

UC53A4. Autorun is disabled.

Description. If autorun is disabled on Windows, either through the user hitting the Shift key during media insertion or by changing the default registry entry, then Launchpad will not appear automatically. Users can still manually start Launchpad.

UC53A5. User removes install media while Launchpad is running.

Description: Launchpad is generally on one CD of the offering. Launchpad should still be able to run without the CD and request the proper CD if user selects something on another CD.

UC53A5.1: System presents message telling user to replace install media to continue. User replaces media.

3.2.4.3 UC54: Start First Steps

Brief Description

First Steps is a post-install tool, linking to key post-install information and functions to get the user started using the software.

A First Steps tool provides the user access to important post-install information and actions. First Steps tool options should be accessible and executable in other ways, such as through commands. This tool simply provides a centralized point to access key post-install options.

First Steps options can be based on features selected during install. For example, First Steps should not provide an option to run samples if samples have not been installed.

When multiple languages are installed, First Steps (and all other user-interface elements) should appear in the current locale language when possible. If the locale language was not installed, it should default to English.

Depending on the offering, First Steps can be automatically started at the end of install, or a preferably a user option can be provided during install whether to launch First Steps. First Steps should also be provided as a command, such as `firststeps.bat` or `firststeps.sh` in offering path and a shortcut in the Start menu.

First Steps can include these options, and others:

- **Offering documentation.** This would link to install documentation to get users using the product or customizing it. Documentation could either be local or on the Internet.
- **Offering information, such as Getting Started or Quick Tour.** This might include a link to the product Web site or an interactive demo of the product, such as one pointing out what's new with this version.
- **Register product.** Follow the standard IBM Registration process for this particular offering. This option assumes user purchased or acquired an offering that can be registered.
- **Self-test or installation verification.** This performs a test that validates that the offering was installed correctly and is ready for use. For more information on self-test, see UC7.
- **Migration.** This option allows user to migrate system or user data (or both) to the newly installed version from the previously installed version. This option should be provided only where appropriate. Migration might also occur as part of the install program. For more information on offering install, see UC5.
- **Start offering or offering administration.** There can be multiple options to start a key element of an offering, such as server or administrative console. These options can be executed in other ways, such as by command or from a Start menu. It is not necessary to provide the ability to start or run every element of the offering, only a few key ones.
- **Samples.** The option allows users to view the samples that are available for this offering as a way to help them get started using the offering. Provide this option only when samples have been installed.
- **Check for updates.** Allows users to check for maintenance package that can be applied to an offering.
- **Exit.** Exits the Launchpad.

Actor

1. User

UC54 Basic Flow

Preconditions

1. User has installed the offering.

UC54B.1: First Steps is started, either automatically at the completion of install, or a user choice can be provided during install to launch First Steps once install is completed.

UC54B.2: First Steps appears, in the language of locale.

UC54B.3: User selects one or more options from First Steps.

UC54B.4: User closes First Steps by choosing Exit.

UC54 Alternative Flows

UC54A1. First Steps could not be started.

UC54A1.1: System presents message that informs users why First Steps could not be started and any actions they need to take so that it will run.

UC54A2. One or more First Steps options could not be executed.

UC54A2.1: System presents message that informs users why option could not be performed and any actions they need to take.

UC54A3. Options that change based on system state.

Description: First Steps options that are provided or active might depend on the state of the system or other options. For example, if samples are not installed, then a link to the samples should not appear on First Steps. A start server link changes to starting the server while that process is working, then Stop the Server when the process is complete. Also, the administrative console or samples cannot be started until after the server has been started. In this case, the administrative console and samples links should be disabled or not made available until the server is started.

3.2.4.4 UC71: View Static Dependency List

Brief Description

View Dependency List links users to the URL that has the latest information about the supported dependencies for the offering version.

Each offering should have a Web site that lists its dependencies. Where appropriate, these dependencies should be separated by platform so that users can quickly go to the platform of interest to them. Specific versions of dependencies should be stated. If the dependency is included with the offering package, it should be clearly stated. A Web site is preferred since information can be updated over time.

Actor

1. User

UC71 Basic Flow

Preconditions

1. User has browser and is connected to Internet.

UC71B.1: User views dependency list, from Launchpad or through navigation from the offering Web page.

Alternative Flows

UC71A1. View dependency fails if user is not connected to Internet, does not have a browser, or page is not found.

Description: These error conditions would behave the same way for any URL, and should rely on the operating system-appropriate method for displaying the error to the user (such as a pop-up error message or command line error message).

3.3 Other Use Cases

3.3.1 Silent Use Cases

3.3.1.1 UC40: Silent install an offering

Brief Description

Customer needs to silently install a specific offering.

Actor

1. User

UC40 Basic Flow

Preconditions

1. Product installer logs on to system using an ID that has authority to perform install.
2. User has access to sample response file, or information on how to create a response file.
3. Response files might be created through the following means:
 - a. Editing the sample response file provided by the product.
 - b. Capturing user inputs during a GUI install recording session.
 - c. Having the install program generate the response file.
 - d. Using a utility to reverse engineer the response file based on a completed install.
 - e. Creating a response file from scratch based on product documentation.

UC40B.1: User prepares the response file with the chosen options.

UC40B.2: User initiates silent install program.

UC40B.3: User reviews log with final message saying the installation has been completed successfully.

UC40 Alternative Flows

Many alternate flows for UC40 silent install are similar to the alternate flows for UC2 interactive install, with the key difference being that any messages would be logged instead of displayed on the screen. See alternate flows for UC2 for possible error conditions. Because users are prevented from making certain errors with the GUI, there are additional alternate flows that apply only to silent install.

UC40A1. User overrides response file parameter with command line parameter.

UC40A2. Required parameter missing in response file.

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC40A3. Parameter has invalid setting. (It might be typo, like parm=treu, or other.)

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC40A4. Parameter conflicts with another parameter (user selects more than one install group or mutually exclusive features.)

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC40A5. User does not accept license agreement in response file.

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC40A6. Uninstall of entire offering.

Description: If uninstall simply uninstalls everything, then a response file might not be required since there are no options to specify

3.3.1.2 UC41: Silent maintenance activity***Brief Description***

Customer needs to silently install or uninstall maintenance package.

Actor

1. User

UC41 Basic Flow***Preconditions***

1. Product installer logs on to system using an ID that has authority to perform install.
2. User has access to sample response file, or information on how to create a response file.
3. Response files might be created through the following means:
 - a. Editing the sample response file provided by the product.
 - b. Capturing user inputs during a GUI install recording session.
 - c. Having the install program generate the response file.
 - d. Using a utility to reverse engineer the response file based on a completed install.
 - e. Creating a response file from scratch based on product documentation.

UC41B.1: User prepares the response file with the chosen options for install or uninstall.

UC41B.2: User initiates silent install or uninstall program.

UC41B.3: User reviews log with final message saying the installation or uninstall has been completed successfully.

UC41 Alternative Flows

Alternate flows for UC41 silent maintenance package activity are similar to the alternate flows for UC11 and UC19 interactive install and uninstall of maintenance packages, with the key difference being that any messages would be logged instead of displayed on the screen. See alternate flows for UC11 and UC19 for possible error conditions.

UC41A1. User overrides response file parameter with command line parameter.

UC41A2. Required parameter missing in response file.

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC41A3. Parameter has invalid setting. (It might be a keying error, like parm=treu, or other.)

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC41A4. Parameter conflicts with another parameter (user selects more than one install group or mutually exclusive features.)

Description: The user should be presented with an error message, and the cause of the failure should be noted in the log with enough detail for the user to correct it.

UC41A5. If uninstall simply uninstalls everything, then a response file might not be required since there are no options to specify.
