Using the Process Solution Command Line Interface

The PSI CLI interface is very similar to that supported in CCMDB 7.1.1.1.

The back-end logic is being updated to support the Maximo subsystem, new in CCMDB 7.1

Invoking the Process Solution Command Line Interface

A launch script is provided for starting the Process Solution Command Line Interface. The script is named solutionInstaller.bat and is deployed and configured in the %CTG_CCMDB_HOME%\bin directory. The Process Solution Command Line Interface executes on the Maximo Admin Workstation. The launch script is deployed and configured by the CCMDB Foundation Installer. No post-installation configuration is required. Simply invoke solutionInstaller.bat with the desired command string and the action is performed.

During processing of the command, the Process Solution Command Line Interface will write messages to the cstandard output of the command window from which the command was launched.

General Syntax

The syntax for invoking solutionInstaller is:

solutionInstaller.bat parameter-clause-1 parameter-clause-2 ... parameter-clause-n

- Each parameter-clause consists of either -parameterName parameterValue or -parameterName .
- -parameterName parameterValue is used for parameters that require a parameter value.
- -parameterName is used for parameters that represent switches or flags which do not require a parameter value.
- parameterName represents the name of one of the supported parameters.
- parameterNames are always prefaced with a dash.
- parameterValue represents the value associated with a particular parameter name.
- parameterValues that contained embedded spaces should be enclosed in double-quotes

Specifying the Action To Perform

A special parameter, -action , must be specified on each invocation of solutionInstaller. This parameter specifies the action or software lifecycle operation to be performed. Based on the value specified for this parameter, additional parameters may need to be specified. For example, when -action showinstall is specified, the type parameter must also be provided. The following table identifies the supported actions that may be specified for the Process Solution Command Line Interface.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Value of -action Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>List Installed Packages</td>
<td>showinstalled</td>
</tr>
</tbody>
</table>
List Available Packages showavail
Base Install install
Incremental Update/FixPack upgrade
Undo Incremental Update/FixPack undo
Uninstall uninstall
Apply Interim Fix applyfix
Undo Interim Fix undofix
List Installed Fixes for a Package showfixes
Refresh Language Support Files for a Package refreshlangs

**Summary of Supported Parameters**

The collection of supported parameters for the Process Solution Command Line Interface are described in the following table.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-action</td>
<td>Specify the function or software lifecycle operation to perform.</td>
</tr>
<tr>
<td>-dbpwd</td>
<td>Specifies the password of the database user ID that is used to access the Maximo database.</td>
</tr>
<tr>
<td>-dbuser</td>
<td>Specifies the database user ID that is used to access the Maximo database.</td>
</tr>
<tr>
<td>-fixid</td>
<td>Specifies the unique identifier of an interim fix/patch that you want processed.</td>
</tr>
<tr>
<td>-force</td>
<td>Specifies whether to continue on with a deployment operation even if there are one or more unsatisfied requirements associated with the package being processed.</td>
</tr>
<tr>
<td>-license</td>
<td>Specifies if you want to automatically accept the license agreement or be prompted for the acceptance or rejection of the license agreement by using one of the following values: accept or prompt.</td>
</tr>
<tr>
<td>-loadlanguages</td>
<td>Specifies whether options Language Support files for the package should be loaded into the Maximo Database.</td>
</tr>
<tr>
<td>-loadsampdata</td>
<td>Specifies whether to load sample or demonstration data associated with the package being processed.</td>
</tr>
<tr>
<td>-maxpwd</td>
<td>Specifies the password of the Maximo administrative user that is used to access the Maximo application.</td>
</tr>
<tr>
<td>-maxuser</td>
<td>Specifies the Maximo administrative user ID that is used to access the Maximo application.</td>
</tr>
<tr>
<td>-pkgpath</td>
<td>Specifies the file path of a process solution package. Paths that have embedded spaces should be enclosed in double quotes.</td>
</tr>
<tr>
<td>-pkguuid</td>
<td>Specifies the unique identifier of the process solution package that you want processed.</td>
</tr>
<tr>
<td>-pkgver</td>
<td>Specifies the version of the process solution package that you want processed.</td>
</tr>
<tr>
<td>-skipdbcfg</td>
<td>Specifies whether to defer the update of the Maximo Database during package deployment.</td>
</tr>
<tr>
<td>-skipj2eecfg</td>
<td>Specifies whether to defer the regeneration and redeployment of the Maximo J2EE Applications during package deployment.</td>
</tr>
<tr>
<td>-type</td>
<td>Specify one of the following types of solution element package to be returned.</td>
</tr>
</tbody>
</table>
when the showavail or showinstalled actions are invoked. Valid values are processmodule, integrationmodule, or all.

-waspwd      Specifies the WebSphere Application Server administrator password.
-wasrxapwd    Specifies the password for the user ID under which remote access to the WebSphere Deployment Manager system is performed.
-wasrxauser   Specifies the user ID under which remote access to the WebSphere Deployment Manager system is performed.
-wasuser      Specifies the WebSphere Application Server administrator user ID.

Process Solution Command Line Interface Reference

showinstalled Action - List Installed Packages

**Action:** showinstalled

**Purpose:** List information about the currently installed process solution packages.

**Syntax:**

```
solutionInstaller  
  -action showinstalled  
  -type {processmodule|integrationmodule|all}
```

**Description:** The showinstalled action displays a list of the currently installed process solution packages that have been installed into the IBM Tivoli Change and Configuration Management Database foundation. Information about each installed package is written to the command prompt.

The following columns of information are displayed for each installed package:

1. **Name** identifies the display name of the installed package.
2. **Version** identifies the version-release-modification level of the installed package.
3. **Solution Type** identifies whether the installed package is a Process Manager Product or a System Integration Module.
4. **Package Type** identifies whether the installed package is a Base Install, Incremental Update, or Full Update package.
5. **Unique Identifier** identifies the 32 character hexadecimal value that uniquely identifies the package.

**Preconditions:** None

**Sample Usage:** The following example will display information about all the currently installed Process Manager Products.

```
solutionInstaller -action showinstalled -type processmodule
```

showavail Action - List Available Packages
**Action:** showavail

**Purpose:** List information about process solution package Zipfiles residing in the CCMDB Install Tree.

**Syntax:**

```
solutionInstaller
   -action showavail
   -type {processmodule|integrationmodule|all}
```

**Description:** The showavail action displays information about the process solution package Zipfiles that are located under the CCMDB Install Tree. This command will examine the process solution package Zipfiles that are located in the `%CTG_CCMDB_HOME%\installableApps\solutions` directory. When the showavail action is initiated, the process solution installation service will query this directory and display information about each process solution package Zipfile in the directory.

The following columns of information are displayed for each package:

1. **Name** identifies the display name of the process solution package.
2. **Version/Fix Name** identifies the version-release-modification level of the process solution package for a Base Install, Incremental Update, or Full Update package and the fix identifier for a Fix Package.
3. **Solution Type** identifies whether the process solution package is for a Process Manager Product or a System Integration Module.
4. **Package Type** identifies whether the process solution package is a Base Install, Incremental Update, Fix, or Full Update package.
5. **Package Path** identifies the filepath of the process solution package Zipfile whose information is displayed.

*Note that a process solution package Zipfile does not need to reside in this directory in order to be processed by the Process Solution Command Line Interface action. Any process solution package Zipfile that is accessible from the Maximo Administrative Workstation may be identified as the parameter value for the -pkgpath parameter.*

**Preconditions:** None

**Sample Usage:** The following example will display information about all the Integration Module process solution package Zipfiles that are located under the `%CTG_CCMDB_HOME%\installableApps\solutions` directory.

```
solutionInstaller -action showavail -type integrationmodule
```

**install Action - Install a Package**

**Action:** install

**Purpose:** Perform a Base Install of a process solution package not already installed.

**Syntax:**
solutionInstaller
   -action install
   -pkgpath <path-to-base-install-package-file>
   [<middleware login information>]
   [-license <accept|prompt>]

Description: The install action is used to install a process solution package into the IBM Change and Configuration Management Database foundation. When installing a package, the filename of the process solution package Zipfile to be installed is specified using the -pkgpath parameter.

Preconditions: Before using this action, you will need to ensure that:

- The process solution package Zipfile you specify is a valid Base Install package.
- The package is not already be installed. You can check this by using the showinstalled action.
- All additional requirements associated with the package specified are satisfied.

Sample Usage: The following example will perform a Base Install of the Process Manager Product whose process solution package is located C:\Config_PMP_7.1.zip file. Access to both the Maximo Database and to the WebSphere Application server are performed using the specified userids and passwords. The license agreement associated with package is displayed for confirmation. Sample data associated with the Process Manager Product is loaded into the Maximo Database.

solutionInstaller
   -action install
   -pkgpath C:\Config_PMP_7.1.zip
   -wasuser <userid> -waspwd <password>
   -dbuser <userid> -dbpwd <password>
   -license prompt -loadsampdata

upgrade Action - Incrementally Update a Package

Action: upgrade

Purpose: Perform an incremental update of a currently installed process solution package.

Syntax:

solutionInstaller
   -action upgrade
   -pkgpath <path-to-update-package-file>
   [<middleware login information>]
   [-license <accept|prompt>]

Description: The upgrade action is used to upgrade a currently installed process solution package. During the upgrade process, the process solution installation service will check to ensure that a currently installed instance suitable to be upgraded has already been installed. When upgrading a package, you must specify the filename of the process solution package file containing the upgraded artifacts.

There are two types of upgrade packages supported:
• **Incremental update package** An incremental update package requires that an existing instance of the process solution package that is being upgraded is already installed.

• **Full update package** A full update package can be used to upgrade an existing instance, or it can be used to install a new instance using the -action install command.

**Preconditions:** Before using this action, you will need to ensure that:

- The process solution package Zipfile you specify is a valid Incremental Update or Full Update package.
- An instance of the process solution package that is eligible to be upgraded is already installed.
- All additional requirements associated with the package specified are satisfied.

**Sample Usage:** The following example will perform an incremental update of an already installed process solution package. The update package is located in the C:\Config_PMP_Upgrade_7.1.1.zip file. Access to both the Maximo Database and to the WebSphere Application server are performed using the specified userids and passwords. The license agreement associated with package is accepted without prompting for user confirmation.

```
solutionInstaller
  -action upgrade
  -pkgpath C:\Config_PMP_Upgrade_7.1.1.zip
  -wasuser <userid> -waspwd <password>
  -dbuser <userid> -dbpwd <password>
  -license accept
```

**undo Action - Undo an Update to a Package**

**Action:** undo

**Purpose:** Remove the changes made to process solution package during a previous incremental update.

**Syntax:**

```
solutionInstaller
  -action undo
  -pkguuid <unique-id-of-package> -pkgver <package-version>
  [[<middleware login information>]]
  [-force]
```

**Description:** The undo action is used to remove the changes that were made by a previously initiated incremental update. This action restores the process solution package to its previous state. During the undo process, the process solution installation service will check to ensure that the specified version of the package is an incremental update version and that the incremental update can be undone.

When undoing an incremental update, the user must specify the unique identifier and version of the incremental update that is to be undone. Note that a particular incremental update package may not support the undo action. If an undo action is initiated for such a package, the process solution installation client will display a message.

**Preconditions:** Before using this action, you will need to ensure that:
The process solution package identified by the unique-id and version that you specify is currently installed and represents an Incremental Update Package.

The Incremental Update that you are attempting to undo supports the Undo action.

All additional requirements associated with the package specified are satisfied.

**Sample Usage:** The following example will undo a previously made incremental update to the process solution package with the specified unique identifier at version 1.1.1. Access to both the Maximo Database and to the WebSphere Application server are performed using the specified userids and passwords.

```
solutionInstaller
  -action undo
  -pkguuid DC2894C667CE48ABAC25214A7FED16D5 -pkgver 1.1.1
  -wasuser <userid>b -waspwd <password>
  -dbuser <userid> -dbpwd <password>
```

**uninstall Action - Uninstall a Package**

**Action:** uninstall

**Purpose:** Uninstall a currently installed process solution package.

**Syntax:**

```
solutionInstaller
  -action uninstall
  -pkguuid <unique-id-of-package>
  [<middleware login information>]
  [-force]
```

**Description:** The uninstall action is used to uninstall a currently installed process solution package. When uninstalling a package, the user will specify the unique identifier of the process solution package to uninstall. The unique identifier is displayed when the showinstalled action is run.

*Not all process solution packages support the ability to be uninstalled. If the package does not support full uninstall, messages are displayed during the uninstall action.*

**Preconditions:** Before using this action, you will need to ensure that:

- The process solution package identified by the unique-id that you specify is currently installed.
- The process solution package supports partial or full uninstallation.
- All additional requirements associated with the package specified are satisfied.

**Sample Usage:** The following example will uninstall the process solution package with the specified unique identifier. Access to both the Maximo Database and to the WebSphere Application server are performed using the specified userids and passwords.

```
solutionInstaller
  -action undo
  -pkguuid DC2894C667CE48ABAC25214A7FED16D5
  -wasuser <userid>b -waspwd <password>
  -dbuser <userid> -dbpwd <password>
```
applyfix Action - Apply Interim Fix to a Package

Action: applyfix

Purpose: Apply an Interim Fix package to a currently installed process solution package.

Syntax:

```
solutionInstaller
   -action applyfix
   -pkgpath <path-to-fix-package-file>
   [ <middleware login information>]  
   [-license <accept|prompt>]  
```

Description: The applyfix action is used to apply patches or interix fixes to a currently installed process solution package. During the upgrade process, the process solution installation service will check to ensure that a currently installed instance suitable to be patched has already been installed. When applying a fix to a package, you must specify the filename of the process solution package file containing the artifacts associated with the fix. This package file is referred to as an Interim Fix Package.

Preconditions: Before using this action, you will need to ensure that:

- The process solution package Zipfile you specify is a valid Interim Fix package.
- An instance of the process solution package that is eligible to be patched by the Interim Fix package is already installed.
- All additional requirements associated with the package specified are satisfied.

Sample Usage: The following example will apply an Interim Fix package to a currently installed process solution package. The Interim Fix package is located in the C:\Config_PMP_Fix_PTF0010.zip file. Access to both the Maximo Database and to the WebSphere Application server are performed using the specified user ids and passwords. The license agreement associated with package is accepted without prompting for user confirmation.

```
solutionInstaller
   -action applyfix
   -pkgpath C:\Config_PMP_Fix_PTF0010.zip
   -wasuser <userid>b -waspwd <password>
   -dbuser <userid> -dbpwd <password>
   -license accept
```

undofix Action - Undo an Interim Fix from a Package

Action: undo

Purpose: Remove a previously applied Interim Fix from a process solution package.

Syntax:

```
solutionInstaller
   -action undo
   -pkguuid <unique-id-of-package> -pkgver <package-version>
   [ <middleware login information>]  
```
Description: The undofix action is used to remove a previously applied fix package. Each Interim Fix package has a fix identifier that uniquely identifies the Interim Fix within a process solution package. You specify this fix identifier (as well as the unique identifier of the base package) when you invoke the undofix action. During the undofix process, the process solution installation service will check to ensure that the specified fix is already applied.

Note that a particular Interim Fix package may not support the undofix action. If an undofix action is initiated for such a package, the process solution installation client will display a message.

You can determine the set of Interim Fixes already applied to a process solution package using the showfixes action of the Process Solution Command Line Interface.

Preconditions: Before using this action, you will need to ensure that:

- The process solution package identified by the unique-id is currently installed.
- The Interim Fix identified by the fix identifier is already applied to the process solution package.
- The Interim Fix that you are attempting to undo supports the undofix action.
- All additional requirements associated with the package specified are satisfied.

Sample Usage: The following example will undo a previously applied Interim Fix with fix identifier PTF00001 from the process solution package with the specified unique identifier. Access to both the Maximo Database and to the WebSphere Application server are performed using the specified userids and passwords.

```
solutionInstaller
   -action undofix
   -pkguuid DC2894C667CE48ABAC25214A7FED16D5
   -fixid PTF00001
   -wasuser <userid>b -waspwd <password>
   -dbuser <userid> -dpwd <password>
```

showfixes Action - List Installed Fixes for a Package

Action: showfixes

Purpose: List information about the Interim Fixes that have been applied to a currently installed process solution package.

Syntax:

```
solutionInstaller
   -action showfixes
   -pkguuid <unique-id-of-package>
   -pkgver <package-version>
```

Description: The showfixes action displays a list of the currently installed Interim Fixes that have been previously applied to a currently installed process solution package. Information about each installed fix is written to the command prompt. When you use this action, you specify the unique identifier and version of the currently installed process solution package for which the collection of Interim Fixes is to be displayed.
The following columns of information are displayed for each applied fix:

1. *Name* identifies the unique identifier for the Interim Fix.
2. *Fix Type* identifies the type of the Interim Fix.

**Preconditions:** Before using this action, you will need to ensure that:

- The process solution package identified by the unique-id and version that you specify is currently installed.

**Sample Usage:** The following example will list the currently applied Interim Fixes for the process solution package having the specified unique identifier and version.

```bash
solutionInstaller
   -action showfixes
   -pkguuid 1480586C22E24D6A8754265DC38AEBDD
   -pkgver 1.1.0
```

**refreshlangs Action - Refresh Language Support Files for a Package**

**Action:** refreshlangs

**Purpose:** Install or refresh the language support files for a package that is already installed.

**Syntax:**

```bash
solutionInstaller
   -action refreshlangs
   -pkgpath <path-to-base-install-package-file>
   [middleware login information]
   [-license <accept|prompt>]
   [-skipj2eeCfg] [-skipDbCfg] [-force]
```

**Description:** The refreshlangs action is used to either install or refresh the language support files associated with a package that is already installed. The action will unpack the language support files contained within the package, write those language support files to the Maximo Administrative Workstation, and then load the language support files into the Maximo Database.

This action can be used whether or not language support files were originally installed using the -loadlanguages flag when the Base Install for the package was performed using the -action install command.

**Preconditions:** Before using this action, you will need to ensure that:

- The process solution package Zipfile you specify is a valid Base Install package.
- The package was previously installed. You can check this by using the showinstalled action.
- The package supports user selectable deployment of language support files.
- All additional requirements associated with the package specified are satisfied.
**Sample Usage:** The following example will refresh the language support files contained within the Process Manager Product whose process solution package is located `C:\Config_PMP_7.1.zip` file. Access to the Maximo Database is performed using the specified userid and password.

```
solutionInstaller
   -action refreshlangs
   -pkgpath C:\Config_PMP_7.1.zip
   -dbuser <userid> -dbpwd <password>
   -license accept
```