Using the SRP Mobile Repository

Introduction

There are occasions when it is convenient or necessary for business procedures to store critical application components outside of the local OpenInsight repository. While OpenInsight does provide ways to extract components (aka entities) into separate folders (e.g. Runtime Deployment Kit, Check-out), these cannot be utilized by OpenInsight without first importing them back into the local repository. Of course, this means those components which are already in the local repository will need to be backed up or they will be over-written. The SRP Mobile Repository allows components to be stored in an external (or mobile) repository. Once a mobile repository is opened its components can be accessed in design mode or executed using the OpenInsight toolset without compromising the original components in the local repository.

While the SRP Mobile Repository is designed to work without any special understanding by the end user, it is often helpful to know how some of the core functionality works. Mobile repositories consist of several database tables that mimic the native repository tables. Currently the following tables are created in a mobile repository location:

MD	MOBREPOS
MOBREPOSAPPNOTES	MOBREPOSMENUCONTEXT
MOBREPOSMESSAGES	MOBREPOSEVENTS
MOBREPOSEVENTEXES	MOBREPOSREPORTS
MOBREPOSWINS	MOBREPOSWINEXES
MOBREPOSPOPUPS	MOBREPOSSYSPROCS
MOBREPOSSYSOBJ	

Accordingly, the only supported entity types at this time are:

|	
Stored Procedures	Forms
Stored Procedure Executables	Form Executables
Stored Procedure Debug Tables	Messages
Events	Popups
Event Executables	QuickHelp
Event Debug Tables	Context Menus
Reports	

If a mission critical entity type is not supported please contact us at support@srpcs.com and we'll see if we can add that one in.

Known Limitations

The SRP Mobile Repository relies on OpenInsight's built-in toolset for accessing, executing, and compiling application entities. If a tool was designed to use alternative methods (e.g. if an entity is read directly from a system table rather than through the Repository function) then the SRP Mobile

Graphical User Interface

The SRP Mobile Repository provides the end user with a full featured user-friendly interface (the GUI) that provides point-and-click as well as drag-and-drop convenience:
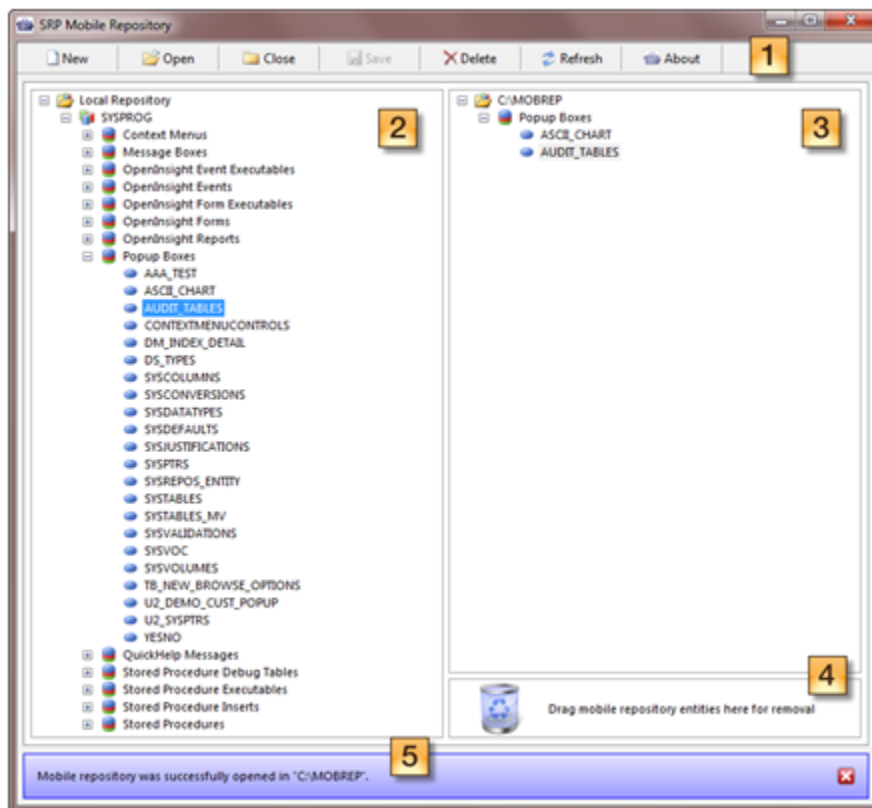
Legend

1. This is the main toolbar of the SRP Mobile Repository. The buttons provide the following features:
   - New – Creates a new mobile repository. User will be prompted with a dialog box to choose a path.
   - Open – Opens and attaches an existing mobile repository. User will be prompted with a dialog box to choose a path.
   - Close – Closes the currently open mobile repository.
   - Save – Updates the currently open mobile repository with the pending add and remove requests.
   - Delete – Deletes the currently open mobile repository. (NOTE: This will remove all files in the mobile repository folder other than sub-folders.)
   - Refresh – Updates the local repository display. This is helpful is entities have been added or removed while the interface is still running.
   - About – Displays the SRP Mobile Repository About box with the current version information.
2. This is a hierarchical view of the local repository. The current and inherited applications will be listed here. Users can select entire entity types (e.g. Popup Boxes, Stored Procedures, etc.) or individual repository entities (one or multiple entities at a time) and drag them onto the right side of the GUI. This will flag these entities to be added when the mobile repository is saved.
3. This is a hierarchical view of the mobile repository. A new mobile repository won't have any entities displayed. Users can drag entities back onto the left side of the GUI to flag these entities for restoration (which happens when the mobile repository is saved.)
4. A drop zone to flag mobile repository entities for removal when the mobile repository is saved.
5. This is the status bar for the GUI. Notifications, wait requests, and error messages will be displayed down here. Each type of message is color coded (e.g. error messages have a red background) for easy identification. The status bar can be closed by clicking on the button or it will automatically close itself when another action is taken.

Launching the GUI

Even though the GUI is an OpenInsight form (SRP_MOBILE_REPOSITORY) it is not meant to be launched directly using Start_Window or any other means of displaying an OpenInsight form. Instead, the API Init method should be used. See the documentation below for more information.

Command Line API

Not all situations can rely on an end user's interaction with the GUI. For instance, if the SRP Mobile Repository is being used for version control then there are probably built-in processes that already know how to package related entities and store them off-line for future reference. For these situations the SRP Mobile Repository can be operated independent of the GUI.

The core routine is the SRP_Mobile_Repository function. It is designed to accept various messages (aka methods or properties) to perform a very specific task. For instance, there is a **CreateMobileRepository** method which only has the responsibility of creating the necessary tables in an empty folder so that it can be used as a mobile repository. These messages are designed to make it easy for a developer to incorporate within existing applications.

Below is a list of all the supported messages with syntax, comments, and sample code. In order to help the developer understand how to use the command line API we have provided the Mobile_Repos stored procedure. This is a fully functional utility in itself which can be called from other routines or it can be used as an example for other programs.