Learning Life Cycle Management (LCM): Command Line Security Synchronization

This purpose of this article is to introduce the command line Life Cycle Management(LCM) utility in Oracle EPM. The LCM tool can be used to export and import objects that can be found within the Oracle EPM Environment. This includes Security, Essbase, Hyperion Planning, Financial Management … etc. As once gets more familiar with LCM, one comes to realize how powerful the tool is and how empty life without LCM was. Without LCM some of the more detailed artifacts within an application were difficult to move between environments. LCM provides a centralized mechanism for exporting and importing nearly all of the objects within an Oracle EPM application or module. The table below is listed to get an idea of all the facets of LCM.

Application Artifacts by Module

Module	Artifacts
Shared Services	User and Group Provisioning Projects/Application Metadata

Essbase	Files (.csc, .rpt, .otl, .rul) Data Filters Partitions Index and Page files (drive letters) Application and Database properties Security
EAS/Business Rules	Rules Locations Sequences Projects Security
Hyperion Planning	Forms Dimensions Application Properties Security
Hyperion Financial Management	Metadata Data Journals Forms/Grids Rules Lists Security
Financial Data Quality Management	Maps Security Data Metadata Scripts Security

Reporting and Analysis (Workspace)	Reports Files Database Connections Security

The LCM tool is integrated into the Shared Services Web Interface. If can be found under the Application Groups tab. Within the application groups there are three main areas of interest:

1. Foundation – includes Shared Services security such as Users/Groups and Provisioning.
2. File System – This is where the exported files will go by default. The default location is to be stored server side, on the Shared Services server in the location: E:\Hyperion\common\import_export
   Under this main folder, the contents are broken out by the user account that performed the export. Within the export folder, there is an "info" folder and a "resource" folder. The info folder provides an xml listing of the artifacts contained within the export. The resource folder contains the actual objects that were exported.

   The LCM Command line tool provides more flexibility because it can be installed on any machine and the results can be directed to output to any local folder. Sometimes this is very useful if the Shared Services node is a Unix machine, and the LCM users are unfamiliar with Unix. Simply install the LCM Command Line Utility on the Windows machine and redirect its output to a local Windows folder using the –local command line option.

3. Products and Applications – Each registered product will be listed and provide a mechanism to export and import

the respective objects for the associated applications, Essbase, Planning…etc.

Going Command Line

The Shared Services LCM GUI is a great way to become familiar with the LCM tool. However, when it is time to start automating LCM tasks and debugging issues, the Command Line LCM utility is very helpful. To get started, the LCM Command Line tool requires a single command line argument, an xml file that contains the migration definition. The quickest way to obtain the xml file is to use the Shared Services LCM Web interface to select the objects you wish, select Define Migration to pull up the LCM Migration Wizard, and follow the prompts until the last step. Two options are presented, "Execute Migration" or "Save Migration Definition". Choose "Save Migration Definition" to save the migration definition to a local file.
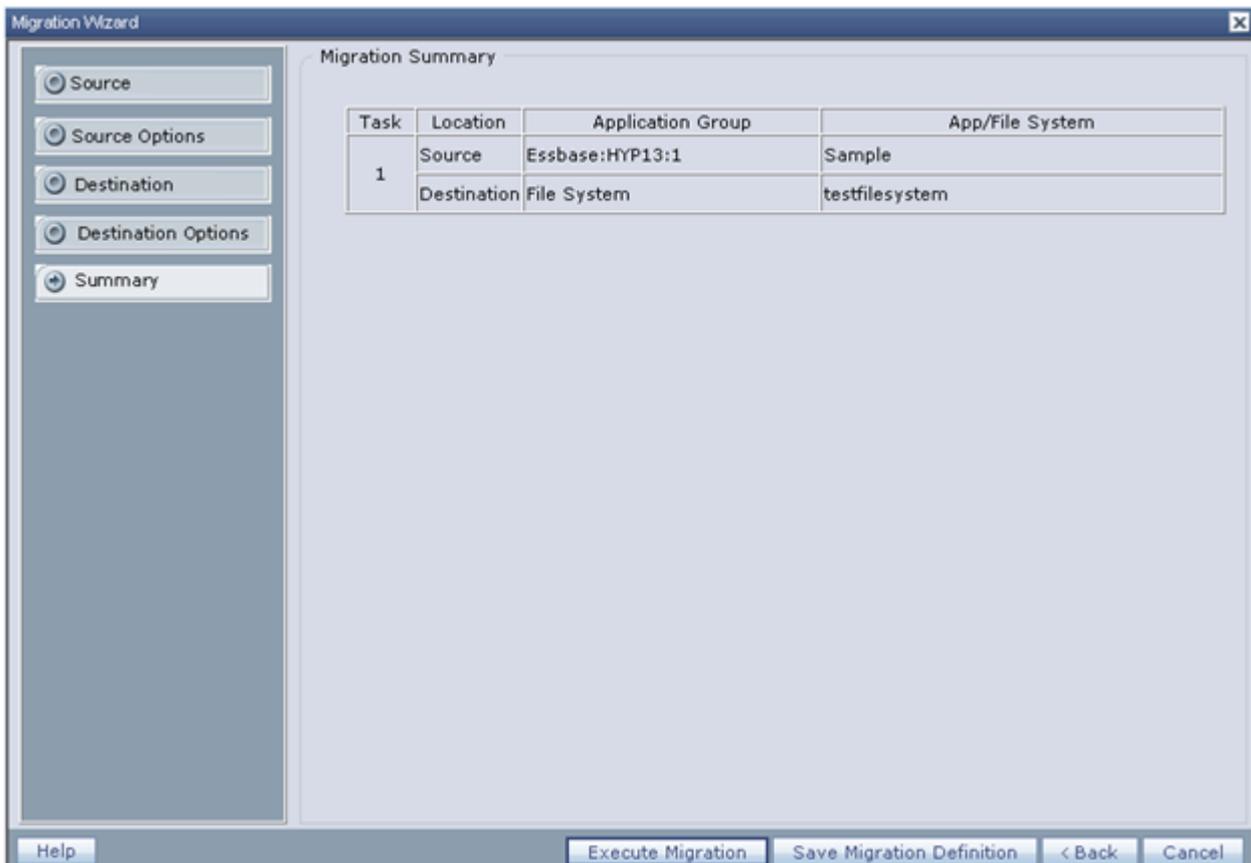
That is pretty much all there is to it… move the xml migration definition file to the location you have installed LCM. For instance, \Hyperion\common\utilities\LCM\9.5.0.0\bin, open a command line and run Utility.bat as indicated:

```
E:\Hyperion\common\utilities\LCM\9.5.0.0\bin>Utility.bat
SampleExport.xml
Attempting to load Log Config File:../conf/log.xml
2011-03-20 11:50:49,015 INFO  - Executing package file -
E:\Hyperion\common\util
ities\LCM\9.5.0.0\bin\SampleExport.xml
>>> Enter username - admin
>>> Enter Password---------
--2011-03-20 11:50:57,968 INFO  - Audit Client has been
created for the server h
ttp://hyp13:58080/interop/Audit
2011-03-20 11:50:58,421 WARN  - Going to buffer response body
of large or unknow
n size. Using getResponseBodyAsStream instead is recommended.
2011-03-20 11:51:03,421 INFO  - Audit Client has been created
for the server htt
p://hyp13:58080/interop/Audit
2011-03-20 11:51:03,437 INFO  - MIGRATING ARTIFACTS FROM
"Foundation/Shared Serv
ices" TO "/SampleExport"
2011-03-20 11:51:32,281 INFO  - Message after RemoteMigration
execution - Succes
s.       HSS      log      file      is      in      -
E:\Hyperion\common\utilities\LCM\9.5.0.0\logs\LCM_2011_0
3_20_11_50_48_0.log
2011-03-20 11:51:32,687 INFO  - Migration Status - Success

E:\Hyperion\common\utilities\LCM\9.5.0.0\bin>
```

LCM Example: Synchronizing Shared

Services Security between Environments

LCM often requires moving objects and security between environments, such as from a development environment to a production environment. While LCM makes it easy, it is not as straightforward as simply running an export from one environment and importing into another environment. The reason is that LCM imports work in a "create/update" mode. In other words, the operations performed in LCM are typically additive in nature. While the typical LCM method would capture new users and new application provisioning, it will not handle removing user provisioning, removing or changing groups, or essentially removing users from the system. This can be an easy oversight, but it will ensure that the security becomes out of sync over time and can cause issues as well as security implications. At a high level, the steps to sync provisioning using LCM would be:

1. Export Users/Groups/Provisioning from Source Environment
2. Export Users/Groups from Target Environment
3. Delete Using Step 2 Results the Users/Groups in Target Environment
4. Import Users/Groups/Provisioning into Target Environment

Essentially, Step 1 and 4 are the typical import/export operations — where security is exported from one environment and imported into another environment. However, two additional steps are necessary. In Step 3, the users and groups in the target environment are deleted, removing provisioning too. This leaves an empty, clean environment to then import security, ensuring no residual artifacts remain in the environment. To use the LCM delete operation, a list of items to be deleted must be supplied. This is where Step 2 comes in, a simple export of the Users and Groups in the Target environment will provide the necessary information to provide to Step 3 — deleting the respective users and groups.

Below are some sample XML migration definitions for each step:

Step 1 – Export Users/Groups/Provisioning from Source Environment

Note: By default the results will be sent to the source Shared Services server in the "import_export" directory. You can use LCM to redirect the output to keep the results all in the same environment (the target system) by using the command line option [-local/-l] (run utility.bat without any command line options to see help for your version of LCM). Simply redirect the results into the local folder, \Hyperion\common\import_export, in the Target system.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Package name="web-migration" description="Migrating Shared Services to File System ">
    <LOCALE>en_US</LOCALE>
    <Connections>
        <ConnectionInfo name="MyHSS-Connection1" type="HSS" description="Hyperion Shared Service connection" url="http://sourceSvr:58080/interop" user="" password=""/>
        <ConnectionInfo name="FileSystem-Connection1" type="FileSystem" description="File system connection" HSSConnection="MyHSS-Connection1" filePath="/Step1ExportFromSource"/>
        <ConnectionInfo name="AppConnection2" type="Application" product="HUB" project="Foundation" application="Shared Services" HSSConnection="MyHSS-Connection1" description="Source Application"/>
    </Connections>
    <Tasks>
        <Task seqID="1">
            <Source connection="AppConnection2">
                <Options>
                    <optionInfo name="userFilter" value="*"/>
                    <optionInfo name="groupFilter" value="*"/>
                    <optionInfo name="roleFilter" value="*"/>
                </Options>
                    <Artifact recursive="false" parentPath="/Native Directory" pattern="Users"/>
```

```xml
                    <Artifact recursive="true" parentPath="/Native
Directory/Assigned Roles" pattern="*"/>
                            <Artifact recursive="false"
parentPath="/Native Directory" pattern="Groups"/>
            </Source>
            <Target connection="FileSystem-Connection1">
                <Options/>
            </Target>
        </Task>
    </Tasks>
</Package>
```

Step 2 – Export Users / Groups from Target Environment

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Package name="web-migration" description="Migrating Shared
Services to File System ">
    <LOCALE>en_US</LOCALE>
    <Connections>
        <ConnectionInfo name="MyHSS-Connection1" type="HSS"
description="Hyperion Shared Service connection"
url="http://targetSvr:58080/interop" user="" password=""/>
          <ConnectionInfo name="FileSystem-Connection1"
type="FileSystem" description="File system connection"
HSSConnection="MyHSS-Connection1"
filePath="/Step2UsersGroupsTarget"/>
                <ConnectionInfo name="AppConnection2"
type="Application" product="HUB" project="Foundation"
application="Shared Services" HSSConnection="MyHSS-
Connection1" description="Source Application"/>
    </Connections>
    <Tasks>
        <Task seqID="1">
            <Source connection="AppConnection2">
                <Options>
                    <optionInfo name="userFilter" value="*"/>
                    <optionInfo name="groupFilter" value="*"/>
                </Options>
                            <Artifact recursive="false"
parentPath="/Native Directory" pattern="Users"/>
```

```xml
                                        <Artifact recursive="false"
parentPath="/Native Directory" pattern="Groups"/>
            </Source>
            <Target connection="FileSystem-Connection1">
                <Options/>
            </Target>
        </Task>
    </Tasks>
</Package>
```

Step 3 – Delete Users/Groups in Target Environment

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Package name="web-migration" description="Migrating File
System to Shared Services">
    <LOCALE>en_US</LOCALE>
    <Connections>
        <ConnectionInfo name="MyHSS-Connection1" type="HSS"
description="Hyperion    Shared    Service    connection"
url="http://targetSvr:58080/interop" user="" password=""/>
            <ConnectionInfo    name="AppConnection1"
type="Application"  product="HUB"  description="Destination
Application"       HSSConnection="MyHSS-Connection1"
project="Foundation" application="Shared Services"/>
        <ConnectionInfo name="FileSystem-Connection2"
type="FileSystem"    HSSConnection="MyHSS-Connection1"
filePath="/Step2UsersGroupsTarget"    description="Source
Application"/>
    </Connections>
    <Tasks>
        <Task seqID="1">
            <Source connection="FileSystem-Connection2">
                <Options/>
                                <Artifact  recursive="false"
parentPath="/Native Directory" pattern="Users"/>
                                <Artifact  recursive="false"
parentPath="/Native Directory" pattern="Groups"/>
            </Source>
            <Target connection="AppConnection1">
                <Options>
                                <optionInfo  name="operation"
```

```
value="delete"/>
                    <optionInfo name="maxerrors" value="100"/>
                </Options>
            </Target>
        </Task>
    </Tasks>
</Package>
```

Step 4 — Import Users and Groups into Clean Target Environment

This step assumes that Step 1 was redirected onto the target environment within the import_export directory. The respective folder, Step1UsersGroupsSource, can also be manually copied from the source to the target environment without using the redirection to a local folder technique.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package name="web-migration" description="Migrating File System to Shared Services">
    <LOCALE>en_US</LOCALE>
    <Connections>
        <ConnectionInfo name="MyHSS-Connection1" type="HSS" description="Hyperion Shared Service connection" url="http://targetSvr:58080/interop" user="" password=""/>
        <ConnectionInfo name="AppConnection1" type="Application" product="HUB" description="Destination Application" HSSConnection="MyHSS-Connection1" project="Foundation" application="Shared Services"/>
        <ConnectionInfo name="FileSystem-Connection2" type="FileSystem" HSSConnection="MyHSS-Connection1" filePath="/Step1UsersGroupsSource" description="Source Application"/>
    </Connections>
    <Tasks>
        <Task seqID="1">
            <Source connection="FileSystem-Connection2">
                <Options/>
                <Artifact recursive="true" parentPath="/Native Directory" pattern="*"/>
            </Source>
```

```
            <Target connection="AppConnection1">
                <Options>
                        <optionInfo name="operation"
value="create/update"/>
                    <optionInfo name="maxerrors" value="100"/>
                </Options>
            </Target>
        </Task>
    </Tasks>
</Package>
```

Troubleshooting with Command Line LCM

LCM can be a great tool when it works flawlessly. However, it can quickly become part of mission critical activities like promoting artifacts from development to production. Consequently, it is necessary to learn some troubleshooting skills to maintain business continuity using LCM.

1. Review the output of the LCM operation. Usually it will provide some detail about the error that was received.
2. Review the server side Shared_services_LCM.log in ORACLE_HOME\logs\SharedServices\SharedServices_LCM.log
3. Turn on debugging for the command line LCM tool. Change the line "info" to "debug" in the files E:\Hyperion\common\utilities\LCM\9.5.0.0\conf in log.xml and hss-log.xml
   <param name="Threshold" value="info" />
4. Use Google, the Oracle Knowledgebase to search for more information.
5. Try only a subset of the initial objects. For instance, Essbase can export a number of objects, Outline, Calc Scripts, Rule Files, Report Scripts, Substation Variables, Location Aliases, and Security. Try one at a time to determine which part of the whole is failing.
6. Restart the environment. LCM is an emerging technology and can sometimes just be in a bad state. I've seen countless LCM issues where bouncing the environment

clears the issue up.

7. Look for special characters that might be present in your data. LCM is a java tool and uses xml and text files to transmit data. There are instances where special characters can mess up the parsing.

8. Look for patches — as mentioned previously, LCM is an emerging technology and is still somewhat buggy (especially older versions). Check release notes in patches for enhancements/bug fixes in LCM.