Life Cycle of SAS® Intelligence Platform Project

Gaurav K Agrawal, Tata Consultancy Services, USA

ABSTRACT

This paper provides an overview of the SAS® 9 Intelligence Platform and the steps to be followed from setting up of this platform to the implementation in the production. This paper will be a help for the project manager from the planning perspective and SAS architect in implementing this platform. This paper facilitates as an introduction to SAS® Data Integration (DI) and Business Intelligence (BI) server and its functionality.

1. INTRODUCTION

This paper covers or explores all the steps required to be followed for SAS Intelligence Platform project. This paper will cover the following topics:

1. SAS Tools, Hardware and Server Architecture Requirement
2. Prerequisite for Installation
3. Installation and Configuration
4. Different strategies to setup Development and Testing environments
5. Environment Setup after Installation and Configuration
6. Development components
7. Backup of Metadata
8. Promotion of Metadata

2. SAS TOOLS, HARDWARE AND SERVER ARCHITECTURE REQUIREMENT

2.1 SAS TOOLS

SAS Institute Inc provides many tools for SAS Intelligence Platform and the first step towards its implementation is to understand the required tools stack as per the business requirement. SAS Institute Inc also provides many industry solutions (based on Intelligence Platform architecture) which too are selected on the basis of requirements. As this paper is mainly focused on the SAS Intelligence Platform, only the tools related to the same will be discussed. SAS Institute Inc sales tools in two category as SAS Data Integration and SAS Business Intelligence Tools. Following is the list of some tools under both of these:

SAS Data Integration Tools:
- Metadata Server
- Data Integration Server
- Stored Process Server
- SPDS Server
- Data Flux® Server
- Management Console
- SAS® Data Integration Studio
- LSF Scheduler
- SAS/ACCESS® Tools to Databases
- Data Flux Studio
- SAS Editor
- External Data Source access Utility

SAS Business Intelligence Tools:
- Business Intelligence Server
- SAS® OLAP Server
- SAS® Stored Processes
- SAS® OLAP Cube Studio
- SAS® Information Map Studio
- Mid Tier
- Web Tools (SAS® Web Report Studio, OLAP Viewer, SAS® Information Delivery Portal)

- LSF Scheduler
- External Data Source access Utility

Third Party Tools required:
- JRE
- Application Server for Mid Tier
- Web Server for Mid Tier
- Content Server for Mid Tier

2.2 HARDWARE AND SERVER ARCHITECTURE

After understanding the requirement and work load, SAS architect team should decide upon the sever structure and configuration of the SAS Intelligence Platform Architecture. This is mainly decided on the basis of non functional (performance) requirements. Note: In this paper it is assumed that SAS platform will be installed on four servers as Metadata, Data Integration, Business Intelligence and Mid Tier.

3. INSTALLATION AND CONFIGURATION

3.1 PREREQUISITE FOR INSTALLATION

3.1.1 Required Installation Architecture
As mentioned above, SAS architect team will come up with an estimate that how many servers and what configuration of severs are required to install SAS Intelligence Platform. Mentioned estimation depends upon on the workload on application.

3.1.2 License File
SAS supplies a license file considering the number of available CPU(s) in server. Hence, the installation architecture and server configuration should be well placed.

3.1.3 Plan File
The Plan file is also one of the key files required for installation purpose. It determines the components that need to be installed on a particular server. SAS generates this file depending on the architecture.

3.2 Installation Steps
Perform the following steps for installation after Installation Media (CDs), License and Plan Files
have been received.
1. Create a dump of the CDs supplied.
2. Ensure this dump is accessible from each machine where installation has to be performed. If the common/shared location/machine is not allowed due to security reasons then copy the dump on all machines where installation needs to carry out.
3. Create required User/Groups on machine (if host authentication) and then add them in appropriate group (SAS User Group). In case of network authentication, only users have to be added in SAS User Group. Recommendation: Create SAS internal users as host authentication to avoid any complexity.
4. Users should have the permission as "Logon as batch job" and "Act as part of Operating System".
5. Follow the sequence as SAS® Metadata server, DI, BI and Mid Tier.
   - For Mid Tier, it is recommended to install WebDAV (Content server) first and then Application server (if required Web server also).
   - For mid Tier WAR files for WebDAV and BI web applications will be deployed in application server as per the process suggested by Application server vendor.
5. The installation wizard will ask for License and Plan file on each server. Choose the type of machine where you are doing installation and accordingly it will choose the required components from the Plan file.
6. Configuration wizard will run after the installation procedure is completed to configure the required components on the machine.
7. After completing the configuration, the wizard will create an HTML file, which will contain steps to be performed to setup the required servers.

Note: Cleansing activity can be performed in ETL job using the Data Flux tool. It depends on the strategy decided by the project, therefore, the scope of cleansing and design should be clear from very beginning. If required, Data Flux tool also has to be installed. Plan file will have an entry for Data Flux installation, whereas License file for this will be supplied separately. LSF scheduler will be installed on DI/BI server for scheduling the SAS Jobs/Reports. Other scheduling utilities also can be used for simple scheduling.

3.3 TESTING, DEVELOPMENT AND PRODUCTION ENVIRONMENT SET UP

In order to set up Test, Development and Production environments there can be two strategies, which are described below.

3.3.1 Separate Machine for Dev/Test/Pro

We will have separate machine/server for Development, Testing and Production environments. It is an easy and recommended solution to have these environments separately. This can be achieved by following the installation procedure separately on each environment server. In this scenario it is also recommended to have different port on Test/Dev/Pro environment in order to avoid any confusion.
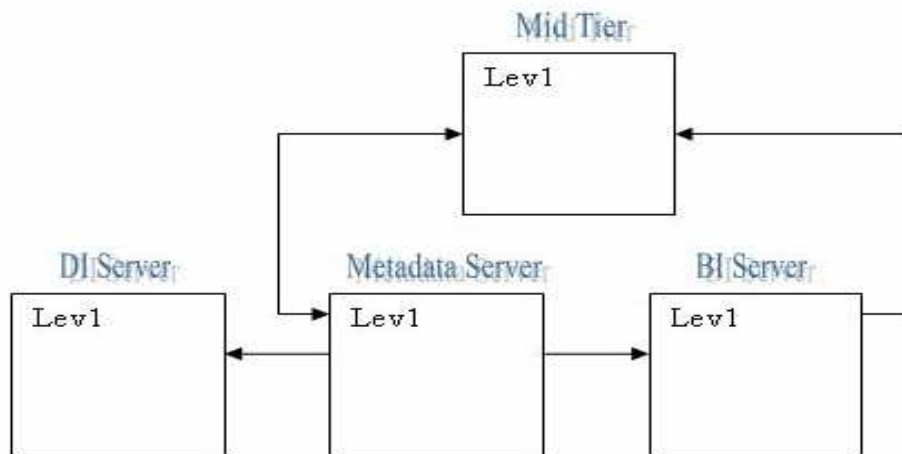
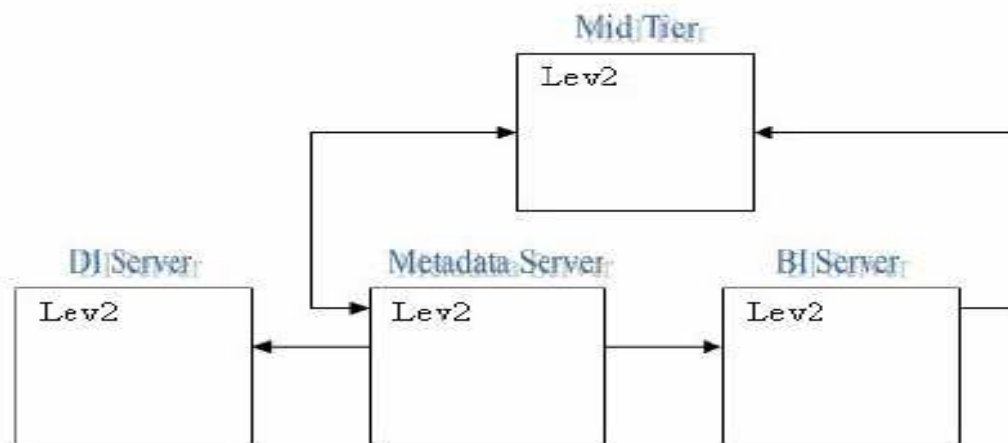**Figure: 1 – Different Boxes for Development Environment**

Figure: 2 – Different Boxes for Testing Environment

Note: Lev1 and Lev2 refer to Development and Testing environments. For production, structure will be the same. Consider this note for other such figures also.

3.3.2 Single Installation but Dual Configuration

"Single Installation and Dual Configuration" is an interesting and cost effective solution although we are trading off with the resources of servers in this solution. In this solution we will execute installation only once but will run configuration wizard multiple time. During the configuration, the wizard will ask

for the port numbers, which should be assigned differently for each configuration. This will ensure that Test/Dev/Pro SAS services run on different ports and do not conflict each other. Above mentioned process will be followed for Metadata and DI server. Trick start when we go to Mid Tier, which indirectly makes BI server installation a trick as both are inter related.

Mid Tier First Approach:

Although dual configuration can be achieved through some manipulation in the property file in web installation directory, it is recommended to have separate machines for Mid Tier and Dual configuration of BI as mentioned above for DI. By this approach, one Mid Tier installation will be linked with first configuration of BI/Metadata server and the second Mid Tier installation will be linked with the second configuration of BI/Metadata server.
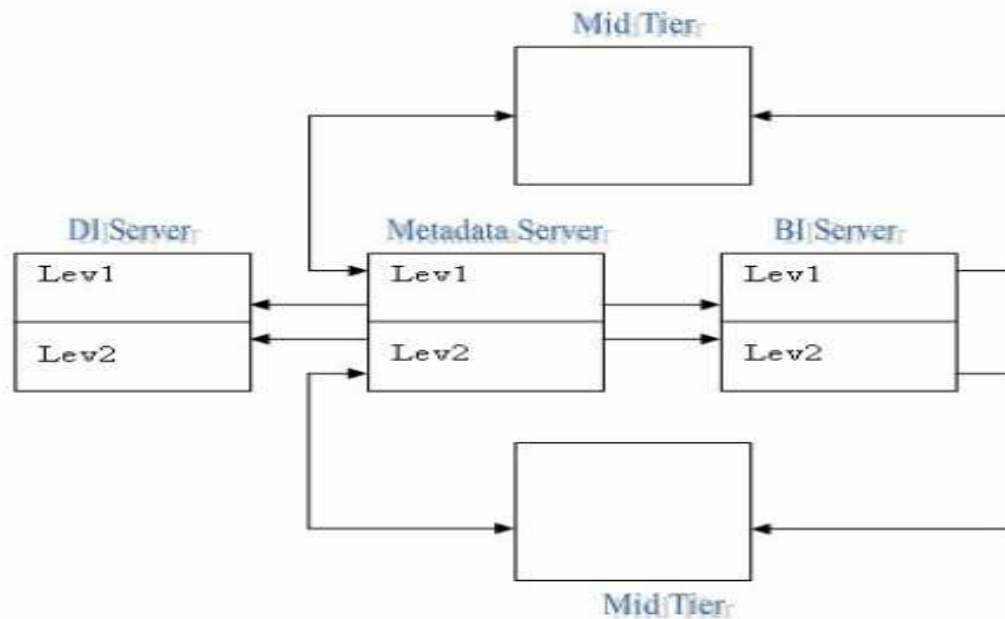

Figure: 3 – Same Boxes (Except Mid Tier) used for Development and Testing Environment

Mid Tier Second Approach:

The second approach is to have different Development and Testing folders in BIP Tree. When Development is done, export all components to Testing folder for the testing. To achieve this scenario, the following points need to be remembered:

- Required Testing libraries have been registered.
- During import process, change all references to Testing libraries from Development.
- This needs to be followed for OLAP cubes also.

Using this approach, Mid Tier will be serving both Dev and Test content but the Developer will choose the content from Dev folder whereas for testing all scripts will point to Testing library only. Applying security, contents can be restricted to users easily. It is recommended that Production environment should be on separate box.
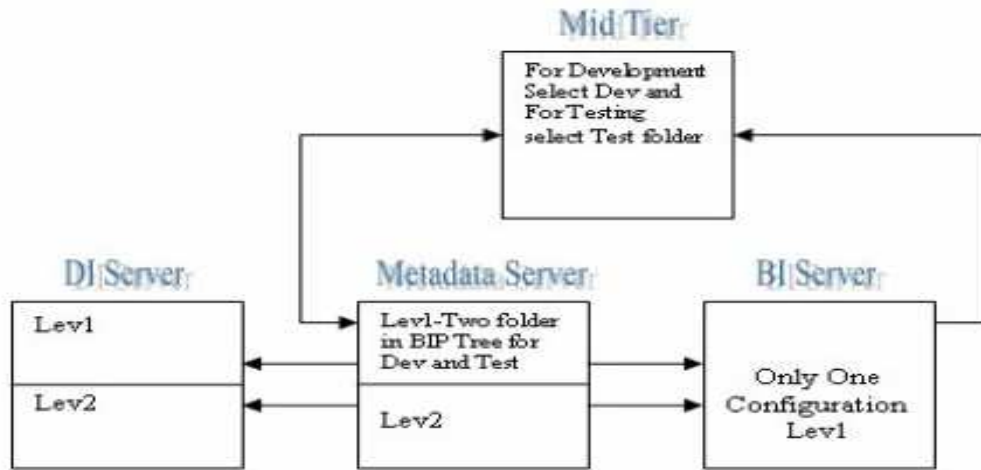
Figure: 4 Same Boxes used for Development and Testing Environment

4. SAS ENVIRONMENT SETUP

4.1 Repository Architecture

Repositories hierarchy/depth can be decided as per the project requirement. This means a Custom repository can have one or more Custom repositories as child. All libraries and tables should be registered in Foundation repositories, which will ensure central control on Libraries and Table metadata. Also it will ensure that the information map (created on tables and OLAP) can be served from Foundation (mandatory for information map) repository. Some rules to be followed at the time of repository setup are given below:

1   All source tables will be registered with Foundation repositories.
2   All target tables will be registered with Foundation repositories.
3   Custom repositories are created as per the strategy decided by architect/business.
4   All developers will be working in project repositories assigned by lead of construction activity.
5   As project repositories are attached with specific modules (Custom repository), this will ensure that jobs are checked in corresponding module repository after required development.
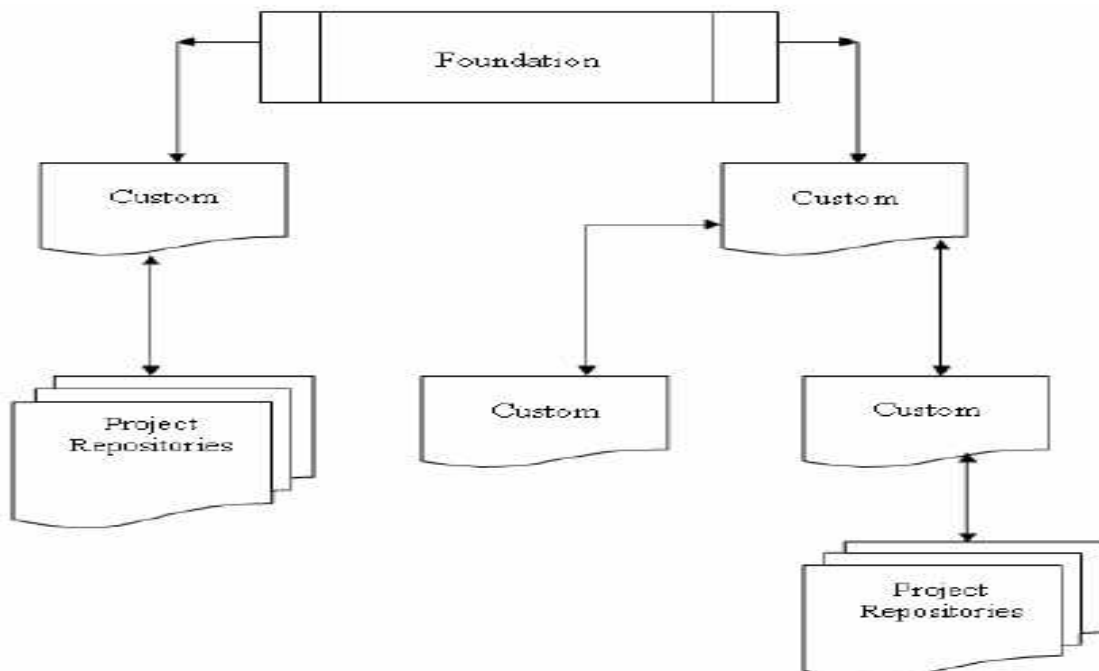6   Jobs should be checked out in dependent project repository to make required changes.

Figure: 5 Repository Hierarchy

4.2 User and Group Access

It is advisable to follow following rules at the time of privileges assignments.

1. All required users will be created in domain.
2. Groups will be created for all modules.
3. Users will be attached to groups.
4. These groups will be provided required permission on Foundation and Custom repository.
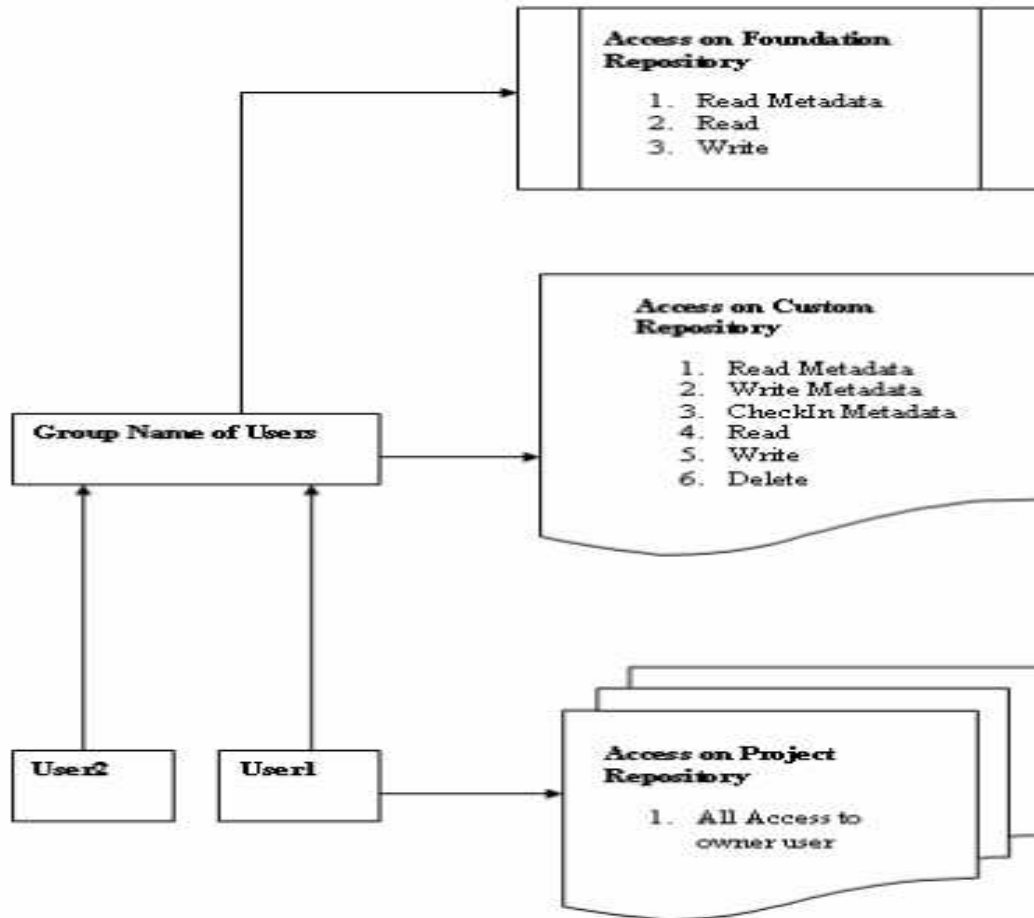5. Users will be given full authority on Project Repository.

Figure: 6 – Security Permission

5. DEVELOPMENT

5.1 SAS ETL Jobs

SAS **E**xtract **T**ransform **L**oad Jobs can be developed using the SAS ETL Studio desktop client. ETL jobs can extract the data from multiple sources, which are defined or registered at the time of environment setup. After extracting the data from different sources, these jobs do manipulation as per the business requirement and load the data in the target system. Note: Cleansing activity can be performed in ETL job using the Data Flux tool. It depends on the strategy decided by the project, therefore, the scope of cleansing and design should be clear from very beginning.

5.2 SAS OLAP

SAS OLAP Cube Studio enables users to build multidimensional cubes through a user-friendly front end. SAS® Metadata server will contain complete architecture of the OLAP cube. SAS OLAP cube studio interacts with Metadata server, Workspace server and OLAP server in order to execute query from cube.

5.3 Information Map

SAS® Information Map Studio is used to define content in a manner that helps business users in designing the report. Information maps are actually same as tables but the names of fields are changed to more business friendly term. An information map presents raw table and column information in business friendly terms that make it easy for anyone to quickly produce analyses of operational data. SAS

Information Map Studio stores these maps in metadata, and uses Workspace and OLAP servers to execute SAS code that supports the associated filters and queries.
Note: Information maps should exist in Foundation repository as Web Report Studio can access the information from Foundation repository only. This concept is likely to change from SAS® 9.2.

5.4 Reports
Reports will be designed using Web Report Studio (WRS). WRS will use information maps and Stored Process as input to design the report.

5.4.1 Web Report Studio
WRS is a webbased tool for providing reporting capabilities on the Web. This tool helps to create reports easily as per the business requirements thus enabling fast decision making. The business users can create reports without having any knowledge of the complex queries. WRS also provides the functionality of creating folders and segregating reports as per different categories decided by the business users. Various types of graphs can also be generated by this tool and can be exported to Excel.

5.4.2 Information Delivery Portal
Information Delivery Portal (IDP) is a single access point for aggregated information through an easyto-use Webbased interface. IDP provides a common frontend for SAS intelligence, Webbased SAS solutions and other digital content in a role based, secure, customizable and extensible environment. Users can easily personalize their portal environments, reducing both information overload and IT workload.

5.5 Jobs Scheduling
To schedule SAS jobs, as used by SAS applications, the jobs must be deployed to a SAS batch server in a schedulingparticipating SAS application. A batch submission flow using the batch submission (bsub) command is shown in Figure 7
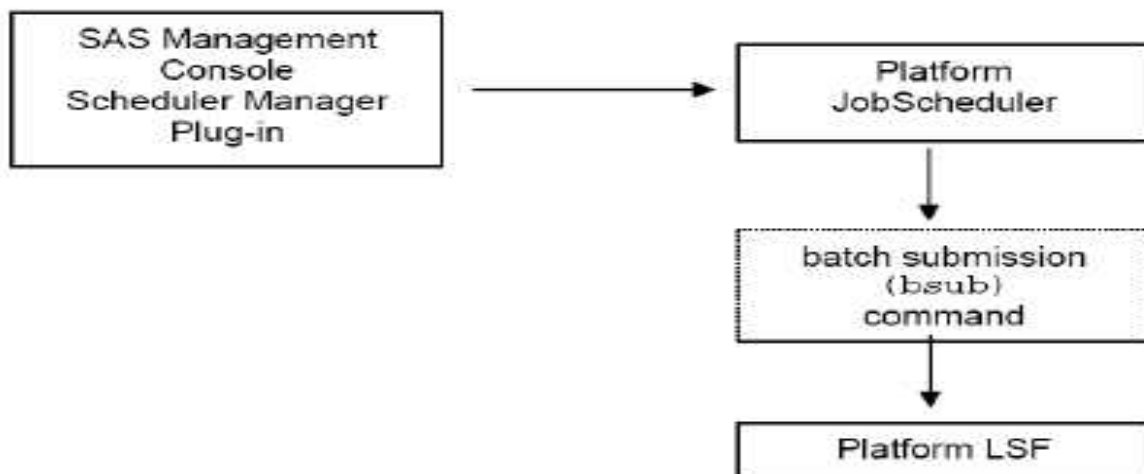

Figure 7

The SAS® Management Console Schedule Manager plugin gets metadata about the flow from the Metadata server, converts that metadata to metadata that the underlying scheduler (Platform Job Scheduler) understands, and submits the information to the scheduling server. Each job can have combinations of time, other jobs, or file dependencies (dependencies are the criteria that must be met so that the job will run). After dependencies are defined for each job, the jobs in the flow can be executed. Platform Job Scheduler executes the jobs in a flow by issuing a "bsub" command to Platform LSF. The Platform LSF configuration contains a cluster of machines that represents a virtual machine. Within a cluster there can be multiple queues, and each queue has its own associated resources and properties. Before you can validate, modify, or reject job submissions, you need to develop the jobscheduling criteria that will be used at your site. Developing the criteria involves understanding the current Platform LSF configuration, knowing what needs to be accomplished, and identifying the available resources.

6. BACKUP OF REPOSITORIES

In a development/production environment it's always recommended that we should have code or metadata backup after a certain period of time to avoid any problems in case of server crash. Generally all product vendors provide facility to backup of the data or metadata. In SAS development environment, it is necessary to maintain regular backup of metadata. To achieve this, SAS provides functionalities like Promotion/Replication, Export/Import and OMABAKUP. Although Promotion/Replication (Data Integration backup only) and export/import are more about transferring the metadata from one machine to other environment, they can be used to maintain regular backup. SAS has created a macro %OMABAKUP specially to maintain the regular backup of the metadata and the same will be discussed in this paper.

6.1 OMABAKUP Macro Functionality

SAS suite provided the functionality of metadata backup through macro called %OMABAKUP. Internally this macro calls XCOPY to copy all metadata existing in the source directory to the target directory. In addition to repository backup, this macro also creates the backup for the repository manager. Repository manager backup may be required in some exceptional cases, for example if a repository is accidentally deleted from actual environment or the actual development/production server crashes. In both the scenarios repository manager also has to be restored. During the backup process, OMABAKUP macro locks the repositories to avoid any other transaction to take place in the meanwhile. OMABAKUP macro also can be used to restore the metadata in case any repository has been corrupted.

6.2 OMABAKUP Macro Syntax and Parameters Description

The syntax of OMABAKUP macro is given below. In this macro the first three parameters are mandatory and remaining parameters are optional.

```
%omabakup(
      DestinationPath ="DestinationPathname ",
      ServerStartPath ="ServerStartPathName ",
      RposmgrPath ="ReposMGRpath",
      <Reorg ="Yes|No">,
      <RepositoryList ="repositoryname<….repositorynamen">>,
      <Restore ="Yes|No">,
      <RunAnalysis ="Yes|No">)
```

Table1: %OMABAKUP Macro Parameter Description

Parameter	Description
DestinationPathname	Absolute path of the target location where backup are to be stored. Specify the absolute pathname of an existing directory. Ensure that the mentioned location exists. OMABAKUP does not create the directory although all folders for the repository backup will be created by OMABAKUP in the mentioned location. For Restore=YES, this directory will be the location where backup is stored.
ServerStartPathName	Full path to the directory in which the Metadata server process was started.
ReposMGRpath	Specify the relative location of the repository manager as defined in the omaconfig.xml file.
Reorg	"YES|NO" Default value is NO. This parameter helps to reclaim unused disk space left from previously deleted metadata objects from SAS metadata repositories. YES (or Y) specifies to reclaim disk space. NO (or N) specifies to copy. This option is not valid when %OMABAKUP is executed in restore mode.
RepositoryList	REPOSITORYLIST="repositoryname<,...repositorynamen>" List all repositories (required to be backed up) separated by commas. If all repositories have to be backed up then just specify this parameter as "ALL". Be careful while listing the repositories as names are case sensitive. Backup and restore both can be selective based on the repositories specified for this parameter.

Restore	YES (or Y) turns on restore mode. NO (or N) specifies that %OMABAKUP run in backup mode. The default value is NO. When RESTORE="YES", %OMABAKUP restores all repositories registered in the SAS Metadata server's repository manager and mentioned in RepositoryList option.
RUNANALYSIS	YES (or Y) specifies to Analyze the backup repositories. NO (or N) specifies not to analyze the backup repositories. The default value is NO. This parameter specifies whether to analyze backup copies of repository data sets for information that can be used to optimize their memory footprint on the Metadata server. The analysis is run immediately after client activity is resumed on the SAS Metadata server. This option is not valid when %OMABAKUP is executed in restore mode.

6.3 General Code
In order to execute the OMABAKUP macro, the program should first connect with the Metadata server and then the macro can be executed. The following code will clarify how to call OMABAKUP macro.

First Get the Encoded Password by below code.
Step 1:
```
proc pwencode in='GauravAgrawalb0962537';
run;
```

```
{sas001}R2F1cmF2QWdyYXdhbA==
NOTE: PROCEDURE PWENCODE used (Total process time):
real time 0.00 seconds
cpu time 0.00 seconds
```

Step 2:
```
/*
This program will help to backup/restore repositories. Before running the program, ensure that all
parameters are set with required value.


Program Name : BackupRepositories.sas
Author : Gaurav Agrawal
*/
%let DestinationPath = %str(C:\SAS\9.1\Lev1\SASBackup);
%let ServerStartPath = %str(C:\SAS\BIArchitecture\Lev1\SASMain);
%let RposmgrPath = %str(/MetadataServer/rposmgr);
%let Repository = %str(ALL);
%let Restore = %str(NO);


/* Connect to Metadata server using below code */
options metaserver = '01xw132923.India.TCS.COM'
metaport = 8561
metaprotocol = bridge
metauser = 'sasadm'
metapass = '{sas001}R2F1cmF2QWdyYXdhbA=='
metarepository = 'Foundation';


%omabakup
(
DestinationPath = "&DestinationPath",
ServerStartPath = "&ServerStartPath",
RposmgrPath = "&RposmgrPath",
```

```
RepositoryList = "&Repository",
Restore = "&Restore"
);
```

This code will create the backup of all repositories registered with the mentioned repository manager. OMABAKUP will create folders for all repositories in the target location at 'C:\SAS\9.1\Lev1\SASBackup' with the repository manager backup. The same program can be used to restore the metadata just by changing the parameter Restore to YES.

7. METADATA PROMOTION

After completing the development work Metadata have to be promoted to Testing and then Production environment. For this SAS provides several methods like Promotion/Replication, Export/Import etc.

7.1 Promotion/Replication

For the promotion replication we need three SAS configurations as Source (To be Migrated), Target (Target server) and Administrator (server that will create the promotion job). This means that the Administrator machine should be able to access Source and Target machines. There should be a network link between Source and Target. If the network fails during the process, it may cause problems. This utility executes the promotion automatically and will not require much manual intervention during the migration process.

7.2 Export/Import Utility

Export/Import utility provides a very easy method to migrate the Metadata from one server to other. This method is independent of network as migration will happen only on a single server at a time. However, all steps have to be performed manually. Execute the following steps to use export import utility:

1. Open Data Integration Studio.
2. Right click the repository that needs to be exported.
3. Click Export.
4. The next window will display all metadata related to DI and BI components under that repository. Select the metadata that needs to be exported.
5. By click on the metadata component, you can see all components attached with selected components in the lower part of the screen. If these components are not already registered within the target, select these metadata components also to migrate.
6. This process will create .spk file. Copy this file to target Metadata server.
7. Right click the repository and select Import.
8. Select the .spk file, which will show all components that are being imported.
9. Change the references if required. These steps can be followed for promotion from Development to Testing and then to Production environment.

8. CONCLUSION

SAS 9 Intelligence Platform provides end to end capabilities with Data Integration to Business Intelligence. This platform provides lots of flexibility in installation and configuration to achieve and maintain cost effective solution for customer. It is very easy to maintain/promote metadata in SAS 9 Intelligence Platform.

9. REFERENCES

1) SAS Institute, Inc. (2006), *SAS® Intelligence Platform Administration Guide v9.1.3, Fifth Edition.* Cary, NC: SAS Institute Inc.

2) SAS Institute, Inc. (2006), *SAS® Intelligence Platform Installation Guide v9.1.3, Fifth Edition.* Cary, NC: SAS Institute Inc.

3) SAS Institute, Inc. (2006), *SAS® Intelligence platform Overview v9.1.3, Second Edition.* Cary, NC: SAS Institute Inc.

4) SAS Institute, Inc. (2006), SAS Institute Inc. 2004. SAS® Web Infrastructure Kit 1.0: Administrator's Guide. Cary, NC: SAS Institute Inc.

5) Work Experience on SAS® 9 Intelligence Platform

10. ACKNOWLEDGMENTS

11. RECOMMENDED READING

http://support.sas.com/documentation/

12. CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:
Name : Gaurav Kumar Agrawal
Address : 1717 R T Dunn Dr, Suite# 2011
City, State ZIP : Bloomington, IL 61701
Work Phone : 7064053450
Fax : 7062436415
Email : gaurav_agrawal@yahoo.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.