**ETL Environment Set Up**

This process needs to be set up on every server that is running SSIS packages.

Create File I/O directory structure

When you have to process import files or output files back to users, there is a directory structure for you to do that. The structure looks like:

[Drive letter]:\InterfaceAndExtractFiles\[Business Unit]\[ETL Process Name]\[In | Out]

You will need to replicate this structure by creating the root InterfaceAndExtractFiles directory.

ETL process names should not be cryptic. They should be as descriptive as possible. When you create a new project in SSDT your project name should match the name of the directory.

Inside of each process directory you create should be an In and Out folder. The In folder should be where you import files from other processes. The Out folder is the result of any processing done in ETL processes where the output is a flat file. You can create whatever file structure necessary to facilitate your process inside of the In/Out folders as long as the base structure mentioned above exist.

Create Global Environment in the Integration Services Catalog

Packages are deployed to the SSIS catalogue using the project deployment model. There are some common variables used across all packages. These variables are stored in an environment named Global. The name value pairs of those variables are as follows:

Variable name: FileExchangeRootDirectory

Value: \\AGS1217-APP01\FileExchange\

Variable name: RootFileDirectory

Value: [Drive Letter]:\InterfaceAndExtractFiles\

Create SQL Server Aliases

Create an alias with the following settings:

Name: AgspringDW

IP: [Server Name]

This setting makes the assumption that server names are relatively set in stone. If server names change, then it’s preferable to use actual IP addresses. Create both 32 and 64bit aliases and make sure TCP/IP is enabled.

Create Server Objects

Create a database called SSISManagement and run the following scripts in order:

1. Create ETL Framework Tables.sql
2. Create ETL Framework Stored Procs.sql
3. Create Configuration Table.sql
4. Create Proc ErrorLogByPackage.sql
5. Create Proc BatchRunTime.sql
6. Create Proc ErrorLogByBatch.sql
7. Create Proc PackageRunTime.sql
8. Create Proc GetVariablesValues.sql