Financial Systems Integration
SQL Server Integration Services (Internal Practices)

		Step-by-Step instructions	Caveat	Example
Getting Started		Start Visual Studio		
		File-->New-->Project		
		Set .Net Framework to desired level		.Net Framework 3.5
		Integration Service Project		
		Set location	Always use network location	\\NSHDSFile\Files\FSI\DWH\XXX\IntegrationScript\XXX
		Give the File a meaningful name		
		Ok to save		
		In the Solution Explorer right click on the SSIS Package to rename the SSIS Package from Package.dtsx to a meaningful name Galad.dtsx	"Do you want to rename the package object" --> yes	Must include the .dtsx at the end of package name
Security		Right click in the main package space (center of screen) and select properties		
		Change ProtectionLevel to "EncryptAllWithPassword"		
		Set PackagePassword with standard FSI password. See Simon if you need password. Note: you must open the property editor by clicking the grey box on the right edge of the box to set password.		
Logging		Right click in the main package space (center of screen) and select Logging	select SSIS from menu-->Logging	
		Check file name to left		
		Under Add a new log, change Provider type to 'SSIS log provider for SQL Server'		
		click Add		
		Click checkbox for Name 'SSIS log provider for SQL Server'		
		Under Configuration (last column in grid) select --> 'New connection'		
		New		
		Add SQL Server		(NSMHBIDB)
		Use Windows Authentication		
		Select starting database name		DWH
		Test Connection		
		Click Ok		
		Click Ok		
		Select Details tab		
		Select OnError, OnTaskFailed, OnWarning		
		OK		
Adding connections	OLE dB Connection	Right click on Connection manager select desired connection type		
		New Ole dB Connection		
		Enter Server Name		10.34.220.11
		Use windows authentication		Use SQL Server Authentication
		Select starting database name	ns-datamart, Nsh#gal22	Add User name/Password, check Save my password
		Test Connection	analyzer_ai6	
		OK	OK	
		Ok	ok	
	e Connection	New Flat File Connection		
		Give it a Connection Manager Name		
		Add Description		
		Click 'Browse' and navigation to desired txt or csv file --> Click Open		
		Under the general options add Format, Text Qualifier, header row delimiter, any header rows to skip and check column names in the first header row(If this is unchecked you will not get the column names imported and you will have to manually add them).		Delimited, '', {CR},{LF}. 0
		Under Columns option preview the data		

Financial Systems Integration
SQL Server Integration Services (Internal Practices)

	Flat fil	Under Advanced option change the data element settings (type, length, etc…) By default all data elements are string 50 character. By Clicking Suggest types automatic data types can be generated (Adjust number of rows to greater than 200 for better results.)		
		Optional - Under Advanced use Suggest Types to allow SSIS to set data types and sizes		
		Click Ok		
	Excel conn	New Excel File Connection		
		Browse to Excel file --> Click open		
Variables		View --> other windows --> Variables		
		Add Variable , Name		GroupID
		Set Data Type		String
		Set default value		"AI_ENT"
misc		toolbox --> double click on Sequence container to add to the control flow toolbox --> drag a dataflow to Sequence container		
Data Flow Tasks		Right Click on Dataflow and rename as what table will be brought in		adjcode_t
		Click on Dataflow tab to enter into dataflow		
		toolbox --> select Data Flow Source --> double click on OLE DB Source		
		double click on OLE DB SOURCE to open OR right click on OLE DB SOURCE and select edit		
		select OLE dB connection manager from dropdown		
		change data access mode to SQL command		
		Add SQL select statement		Select [columns] from [database].[schema].[table]
		OK		
		toolbox --> add derived column		
		connect OLE dB source to Derived column		
		double click derived column to open		
		Add Derived column name		dcGroupID
		Add Expression	you may need to change data type, if so use cast	(DT_STR,10,1252)@[User::GroupID]
		OK		
		toolbox --> add OLE db Destination		
		Connect Derived column to OLE dB Destination		
		select OLEdb Source from the toolbox and add to dataflow		
		select desired connection from dropdown box		
		select Table or view - fast load		
		Select desired table		
		click on mapping (verify mapping)		
		To test data flow right click on		
SSIS configuration		Go to SSIS --> Package Configurations		
		check enable Package configurations		
		Click Add		
		Change configuration type to SQL Server		
		Select Connection from drop down or add new		
		New - to add database schema		
		New		
		Type server name		
		Use windows authentication		
		Select SSISConfiguration Database name		
		OK		
		OK		
		Select SSISConfiguration table		

	Add Configuration filter (same name as the project schema)		
	Next		
	In the Objects tree select the desired items (Connection Managers, paths, variables…)		
	Next		
	Give it a configuration Name "<project>_Configuration"		
	Finish		
	Close		
Redirecting errors	in file source go to error output		
	Change desired row errors/truncations from Fail Components to Redirect row		
	OK		
	toolbox --> Data Flow Destinations --> Add new flat file Destination		
	connect red error to file source		
	Add destination		
	New		
	Delimited		
	Set Connection manager name <SourceName>Error		
	Browse to file location		
	Set file name to <ProjectName>Error.txt		
	Save		
	Set text qualifier to "		
	Check columns to first row		
	Set deliminator		
	OK		
	Go to mapping.		
	on Input Column choose to Ignore "ErrorCode" and "ErrorColumn"		
	OK		
ch Loop example	Create variable with a Scope of the entire project, set Type as object (Will be used outside Foreach loop container to hold data from table source)	rdGalenTables	
	Create variable with a Scope of the entire project, set Type as string (	strProcessName	
	set type as string		
	Create variable with a Scope of the entire project, set Type as string (Will be used to change query information in loop)	strSourceSystem	
	Create (14) variables with a scope of the entire project, set type as string (Will be used as the 14 queries being run for each practice.) Give each variable a default value of a valid select statement	strAdjcode_t	
	toolbox --> Add ForEach Loop Container		
	Drag Existing Sequence Container into Loop		
	Toolbox --> add new dataflow outside of ForEach loop		
	connect dataflow to ForEach Loop		
	Go into dataflow task		
	add new data flow source	Oledb	
	configure data flow source for lookup/reference table (This is the table that houses the names of the various medical practices)	[DWH].[dbo].[DWHProcesses_LU]	
	Use SQL Command see code to side	SELECT [Process] ,[SourceSystem] FROM [DWH].[dbo].[DWHProcesses_LU] WHERE DataSource = 'Galen' AND [Process] NOT IN ('AGA_PATH', 'AGA_PSA')	
	Select Process and SourceSystem columns		
	toolbox --> from destination add recordset dataset		
	assign global variable name to record set		
	on Input columns tab check both columns		

ForEac	Go into ForEach Loop		
	Collection --> change Enumerator to Foreach ADO Enumerator		
	ADO Object source variable set to User::GalenTalbes		
	Variable Mapping --> Chose ProcessName set index as 0		
	For Sourcesystem set index as 1		
	Add Script task to the ForEach loop container outside of the sequence container		
	Add all read only variables source system		
	add all readwrite variables ->all tables variables		
	Edit script		
	Go to Main		
	Add comment on task being done.		
	For each of the 14 variables created earlier replace portion of table name that needs to change for loop to work. See (Dts.Variables("User::strAdjcode_t").Value.ToString.Replace("ai6", Dts.Variables("User::strSourceSystem").Value))		
	save		
	close		
	OK on script task editor		
	go to data flow for table		
	go to oledb source change data access mode to SQL command from variable		
	set variable to variable created		
	OK		
misc	All network folders referenced in SSIS package must be give security permissions to the SQL Server		