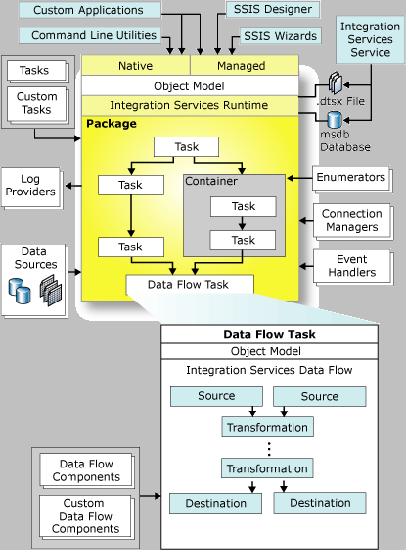
# SSIS Architecture



# Development Interface

* 

## Control Flow

## Data Flow

## Parameters

## Event Handlers

## Package Exploration

## Execution Result

# Log

## SSIS Log Provider

### Text Files

* Useful – anyone can understand – we can send the text file to the concerned team / people
* Folder access needed

### SQL Server

* It creates SYS.SSISLog table
* Read permission needed on the sys.ssislog table. Getting permission in production is most probably difficult.

### SQL Server Profiler

* It will create Trace file in shared directory.
* Folder access needed

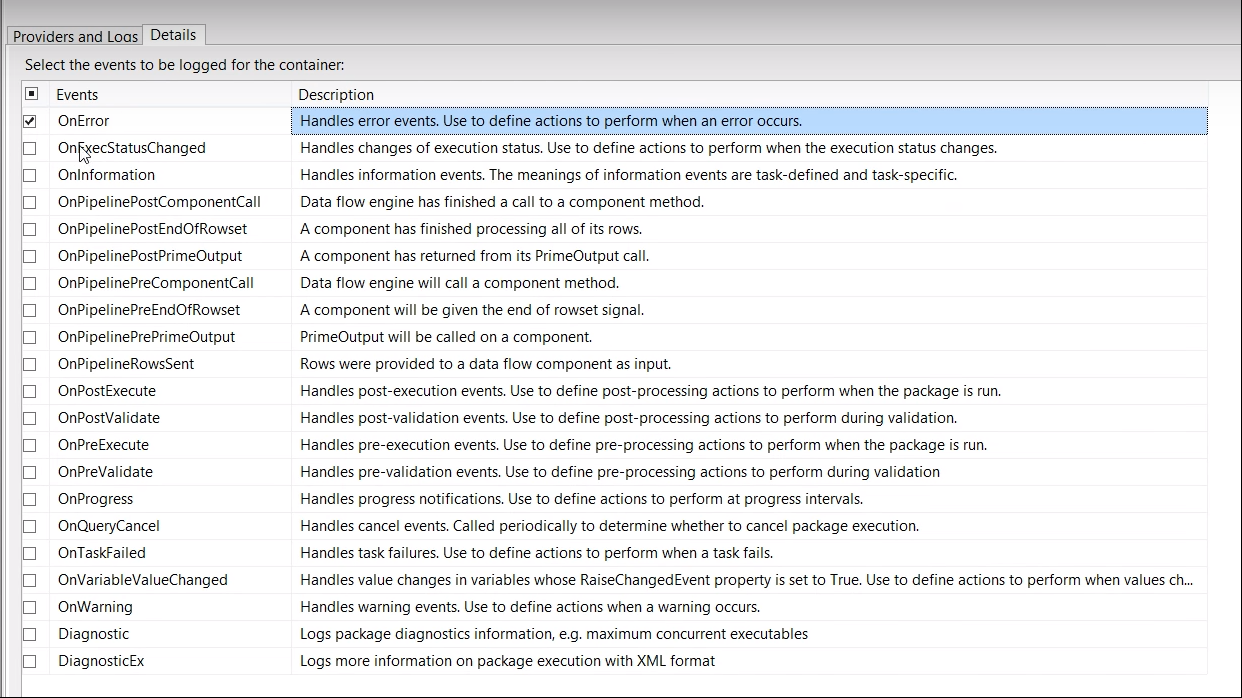
### Windows Event Log

* We need read permission on windows event viewer , If we don’t have access on that server, we cannot see – so it is not good choice

### XML file

* Folder access needed

## Logging Events



# Terminology:

## SSISDB Database

## SSISDB catalog

* The SSISDB catalog is the central point for working with Integration Services (SSIS) projects that you’ve deployed to the Integration Services server. For example, you set project and package parameters, configure environments to specify runtime values for packages, execute and troubleshoot packages, and manage Integration Services server operations.
* The objects that are stored in the SSISDB catalog include projects, packages, parameters, environments, and operational history.
* You inspect objects, settings, and operational data that are stored in the SSISDB catalog, by querying the views in the SSISDB database. You manage the objects by calling stored procedures in the SSISDB database or by using the UI of the SSISDB catalog. In many cases, the same task can be performed in the UI or by calling a stored procedure.

## Deployment:

* SQL Server Integration Services includes tools and wizards that make it simple to deploy packages from the development computer to the production server or to other computers.
* There are four steps in the package deployment process:
  + The first optional step is optional and involves creating package configurations that update properties of package elements at run time. The configurations are automatically included when you deploy the packages.
  + The second step is to build the Integration Services project to create a package deployment utility. The deployment utility for the project contains the packages that you want to deploy
  + The third step is to copy the deployment folder that was created when you built the Integration Services project to the target computer.
  + The fourth step is to run, on the target computer, the Package Installation Wizard to install the packages to the file system or to an instance of SQL Server.