**SSIS Interview Questions and Answers for Experienced and Freshers**

Here we are publishing series of posts on SSIS Interview questions with answers for experienced and freshers . Below is the series 1.

**Q. Define SSIS?**

**Ans:**

SQL Server Integration Services — commonly known as SSIS is the new platform that was introduced in SQL Server 2005, for data transformation and data integration solutions. This replaced the DTS in SQL Server 2000.

**Q. Name a few SSIS components?**

**Ans**:

* Integration Services Projects
* Integration Services Packages
* Control Flow Elements
* Data Flow Elements
* Integration Services Connections
* Integration Services Variables
* Integration Services Event Handlers
* Integration Services Log Providers

**Q. What is a project and Package in SSIS?**

**Ans:**

Project is a container for developing packages. Package is nothing but an object. It implements the functionality of ETL — Extract, Transform and Load — data.

**Q. What are the 4 elements (tabs) that you see on a default package designer in BIDS?**

**Ans:**

Control Flow, Data Flow, event Handler and package explorer. (Parameters – 2012 Data Tools)

**Q.** **What is a Control flow and Data Flow elements in SSIS?**

**Ans:**

**Control Flow:**

Control flow element is one that performs any function or provides structure or control the flow of the elements. There must be at least one control flow element in the SSIS package. In SSIS a workflow is called a control-flow. A control-flow links together our modular data-flows as a series of operations in order to achieve a desired result.

A control flow consists of one or more tasks and containers that execute when the package runs. To control order or define the conditions for running the next task or container in the package control flow

**Data Flow:**

All ETL tasks related to data are done by data flow elements. It is not necessary to have a data flow element in the SSIS package. A data flow consists of the sources and destinations that extract and load data, the transformations that modify and extend data, and the paths that link sources, transformations, and destinations. Before you can add a data flow to a package, the package control flow must include a Data Flow task. The Data Flow task is the executable within the SSIS package that creates, orders, and runs the data flow. A separate instance of the data flow engine is opened for each Data Flow task in a package.

**Q. What are the 3 different types of control flow elements in SSIS?**

**Ans:**

* Structures provided by Containers
* Functionality provided by Tasks
* Precedence constraints that connect the executables, containers, and tasks into an ordered control flow.

**Q. What are the 3 data flow components in SSIS?**

**Ans:**

* Source
* Transformation
* Destination

**Q. What are connections and connection managers in SSIS?**

**Ans:**

Connection as its name suggests is a component to connect to any source or destination from SSIS — like a sql server or flat file or lot of other options that SSIS provides. Connection manager is a logical representation of a connection.

**Q. What is the use of Check Points in SSIS?**

**Ans:**

SSIS provides a Checkpoint capability which allows a package to restart at the point of failure.

**Q. What are the command line tools to execute SQL Server Integration Services packages?**

**Ans:**

**DTSEXECUI –** When this command line tool is run a user interface is loaded in order to configure each of the applicable parameters to execute an SSIS package.

**DTEXEC –** This is a pure command line tool where all of the needed switches must be passed into the command for successful execution of the SSIS package.

**Q. Can you explain the SQL Server Integration Services functionality in Management Studio?**

**Ans:**

You have the ability to do the following:

* Login to the SQL Server Integration Services instance
* View the SSIS log
* View the packages that are currently running on that instance
* Browse the packages stored in MSDB or the file system
* Import or export packages
* Delete packages
* Run packages

**Q. Can you name some of the core SSIS components in the Business Intelligence Development Studio you work with on a regular** **basis** **when** **building an SSIS package?**

**Ans:**

* Connection Managers
* Control Flow
* Data Flow
* Event Handlers
* Variables window
* Toolbox window
* Output window
* Logging
* Package Configurations

**Q. Name Transformations available in SSIS?**

**Ans:**

**DATACONVERSION:** Converts columns data types from one to another type. It stands for Explicit Column Conversion.

**DATAMININGQUERY:** Used to perform data mining query against analysis services and manage Predictions Graphs and Controls.

**DERIVEDCOLUMN:** Create a new (computed) column from given expressions.

**EXPORTCOLUMN:** Used to export a Image specific column from the database to a flat file.

**FUZZYGROUPING:** Used for data cleansing by finding rows that are likely duplicates.

**FUZZYLOOKUP:** Used for Pattern Matching and Ranking based on fuzzy logic.

**AGGREGATE:** It applies aggregate functions to Record Sets to produce new output records from aggregated values.

**AUDIT:** Adds Package and Task level Metadata: such as Machine Name, Execution Instance, Package Name, Package ID, etc..

**CHARACTERMAP:** Performs SQL Server column level string operations such as changing data from lower case to upper case.

**MULTICAST:** Sends a copy of supplied Data Source onto multiple Destinations.

**CONDITIONALSPLIT:** Separates available input into separate output pipelines based on Boolean Expressions configured for each output.

**COPYCOLUMN:** Add a copy of column to the output we can later transform the copy keeping the original for auditing.

**IMPORTCOLUMN:** Reads image specific column from database onto a flat file.

**LOOKUP:** Performs the lookup (searching) of a given reference object set to a data source. It is used for exact matches only.

**MERGE:** Merges two sorted data sets into a single data set into a single data flow.

**MERGEJOIN:** Merges two data sets into a single dataset using a join junction.

**ROWCOUNT:** Stores the resulting row count from the data flow / transformation into a variable.

**ROWSAMPLING:** Captures sample data by using a row count of the total rows in dataflow specified by rows or percentage.

**UNIONALL:** Merge multiple data sets into a single dataset.

**PIVOT:** Used for Normalization of data sources to reduce anomalies by converting rows into columns

**UNPIVOT:** Used for de-normalizing the data structure by converts columns into rows in case of building Data Warehouses.

**SSIS Interview Questions and Answers for Experienced and Fresher’s**

**Here we are publishing series of posts on SSIS Interview questions and answers Part 2 for experienced and freshers** **. Below is the series 2.**

**Q. What is a breakpoint in SSIS?**

**Ans:**

A breakpoint is a stopping point in the code. The breakpoint can give the Developer\DBA an opportunity to review the status of the data, variables and the overall status of the SSIS package.  
Breakpoints are setup in BIDS. In BIDS, navigate to the control flow interface. Right click on the object where you want to set the breakpoint and select the ‘Edit Breakpoints…’ option.

**Q. Can you name 5 or more of the native SSIS connection managers?**

**Ans:**

* **OLEDB connection** – Used to connect to any data source requiring an OLEDB connection (i.e., SQL Server)
* **Flat file connection –** Used to make a connection to a single file in the File System. Required for reading information from a File System flat file
* **ADO.Net connection –** Uses the .Net Provider to make a connection to SQL Server 2005 or other connection exposed through managed code (like C#) in a custom task
* **Analysis Services connection –** Used to make a connection to an Analysis Services database or project. Required for the Analysis Services DDL Task and Analysis Services Processing Task
* **File connection –** Used to reference a file or folder. The options are to either use or create a file or folder
* **Excel**
* **FTP**
* **HTTP**
* **MSMQ**
* **SMO**
* **SMTP**
* **SQL Mobile**
* **WMI**

**Q. How do you eliminate quotes from being uploaded from a flat file to SQL Server?**   
**Ans:**

In the SSIS package on the Flat File Connection Manager Editor, enter quotes into the Text qualifier field then preview the data to ensure the quotes are not included.

**Q. Can you name 5 or more of the main SSIS tool box widgets and their functionality?**

**Ans:**

* ActiveX Script Task
* Analysis Service Processing Task
* Analysis Services Execute DDL Task
* Backup Database Task
* Bulk Insert Task
* CDC Control Task
* Check Database Integrity Task
* Data Flow Task
* Data Mining Query Task
* Data Profiling Task
* Execute DTS 2000 Package Task – Till 2008
* Execute Package Task
* Execute Process Task
* Execute SQL Server Agent Job Task
* Execute SQL Task
* Execute T-SQL Statement Task
* Expression Task
* File System Task
* For Loop Container
* Foreach Loop Container
* FTP Task
* History Cleanup Task
* Maintenance Cleanup Task
* Message Queue Task
* Notify operator Task
* Rebuild Index Task
* Reorganize Index Task
* Script Task
* Send Mail Task
* Sequence Container
* Shrink Datbase Task
* Transfer Database Task
* Transfer error message
* Transfer Jobs Task
* Transfer Logins Task
* Transfer Mastor Stored Procedures Task
* Transfer SQL Server Object Task
* Update Ststistics Task
* Web Service Task
* WMI Datareader Task
* WMI Event Watcher Task
* XML Task

**Q. Can you explain one approach to deploy an SSIS package?**

**Ans:**

* One option is to build a deployment manifest file in BIDS, then copy the directory to the applicable SQL Server then work through the steps of the package installation wizard
* A second option is using the dtutil utility to copy, paste, rename, delete an SSIS Package
* A third option is to login to SQL Server Integration Services via SQL Server Management Studio then navigate to the ‘Stored Packages’ folder then right click on the one of the children folders or an SSIS package to access the ‘Import Packages…’ or ‘Export Packages…’option.
* A fourth option in BIDS is to navigate to File | Save Copy of Package and complete the interface.

**Q. Can you explain how to setup a checkpoint file in SSIS?**

**Ans:**   
The following items need to be configured on the properties tab for SSIS package:

**CheckpointFileName** – Specify the full path to the Checkpoint file that the package uses to save the value of package variables and log completed tasks. Rather than using a hard-coded path, it’s a good idea to use an expression that concatenates a path defined in a package variable and the package name.

**CheckpointUsage** – Determines if/how checkpoints are used. Choose from these options: Never (default), If Exists, or Always. Never indicates that you are not using Checkpoints. “If Exists” is the typical setting and implements the restart at the point of failure behavior. If a Checkpoint file is found it is used to restore package variable values and restart at the point of failure. If a Checkpoint file is not found the package starts execution with the first task. The Always choice raises an error if the Checkpoint file does not exist.

**SaveCheckpoints** – Choose from these options: True or False (default). You must select True to implement the Checkpoint behavior.

**Q. Would you recommend using “Check Points” in SSIS?**

**Ans:**

As per my experience I could say “NO” as there are compatibility issues with various options hence using checkpoints may give unpredictable results. Checkpoints doesn’t work properly when a SSIS package contains

* Complex logic
* Iterations/Loops
* Transactions Enabled
* “Object” type variables
* Parallel execution

Checkpoints works fine when the package is having straightforward control flow with a single thread.

**Q. Can you explain different options for dynamic configurations in SSIS?**

**Ans:**

* Use an XML file
* Use custom variables
* Use a database per environment with the variables
* Use a centralized database with all variables

**Q. How do you upgrade an SSIS Package?**

**Ans:**

Depending on the complexity of the package, one or two techniques are typically used:

* Recode the package based on the functionality in SQL Server DTS.
* Use the Migrate DTS 2000 Package wizard in BIDS and then recode any portion of the package that is not accurate

**Q. Can you name five of the Perfmon counters for SSIS and the value they provide?**

**Ans:**

**SQLServer: SSIS Service**

**SSIS Package Instances** – Total number of simultaneous SSIS Packages running

**SQLServer: SSIS Pipeline**

**BLOB bytes read** – Total bytes read from binary large objects during the monitoring period.

**BLOB bytes written** – Total bytes written to binary large objects during the monitoring period.

**BLOB files in use** – Number of binary large objects files used during the data flow task during the monitoring period.

**Buffer memory:** The amount of physical or virtual memory used by the data flow task during the monitoring period.

**Buffers in use** – The number of buffers in use during the data flow task during the monitoring period.

**Buffers spooled** – The number of buffers written to disk during the data flow task during the monitoring period.

**Flat buffer memory** – The total number of blocks of memory in use by the data flow task during the monitoring period.

**Flat buffers in use** – The number of blocks of memory in use by the data flow task at a point in time.

**Private buffer memory** – The total amount of physical or virtual memory used by data transformation tasks in the data flow engine during the monitoring period.

**Private buffers in use** – The number of blocks of memory in use by the transformations in the data flow task at a point in time.

**Rows read** – Total number of input rows in use by the data flow task at a point in time.

**Rows written** – Total number of output rows in use by the data flow task at a point in time.

**Q. How do you handle errors in ssis?**

**Ans:**

When a data flow component applies a transformation to column data, extracts data from sources, or loads data into destinations, errors can occur. Errors frequently occur because of unexpected data values.

Errors typically fall into one the following categories:

**Data conversion errors:** occurs if a conversion results in loss of significant digits, the loss of insignificant digits, and the truncation of strings. Data conversion errors also occur if the requested conversion is not supported.

**Expression evaluation errors**: occurs if expressions that are evaluated at run time perform invalid operations or become syntactically incorrect because of missing or incorrect data values.

**Lookup errors**: occurs if a lookup operation fails to locate a match in the lookup table.

Many data flow components support error outputs, which let you control how the component handles row-level errors in both incoming and outgoing data. You specify how the component behaves when truncation or an error occurs by setting options on individual columns in the input or output.

**Q. How do you do Logging in SSIS?**

**Ans:**

* SSIS includes logging features that write log entries when run-time events occur and can also write custom messages.
* The Integration Services log providers can write log entries to text files, SQL Server Profiler, SQL Server, Windows Event Log, or XML files.
* Logs are associated with packages and are configured at the package level. Each task or container in a package can log information to any package log. The tasks and containers in a package can be enabled for logging even if the package itself is not.

To enable logging in a package:

* In Business Intelligence Development Studio, open the Integration Services project that contains the package you want.
* On the SSIS menu, click Logging.
* Select a log provider in the Provider type list, and then click Add.

**Q. Demonstrate how you would suggest using configuration files in packages. Would you consider it a best practice to create a configuration file for each connection manager or one for the entire package?**

**Ans:**

There should be a single configuration file for each connection manager in your packages that stores their connection string information. So if you have 6 connection managers then you have 6 config files. You can use the same config file across all your packages that use the same connections.

If you have a single config file that stores all your connection managers then all your packages must have contain the connection managers that are stored in that config file. This means you may have to put connection managers in your package that you don’t even need.

**Q. Demonstrate how checkpoints work in a package.**

**Ans:**

When checkpoints are enabled on a package if the package fails it will save the point at which the package fails. This way you can correct the problem then rerun from the point that it failed instead of rerunning the entire package. The obvious benefit to this is if you load a million record file just before the package fails you don’t have to load it again.

**Q. Demonstrate how transactions work in a package.**

**Ans:**

If transactions are enabled on your package and tasks then when the package fails it will rollback everything that occurred during the package. First make sure MSDTC (Microsoft Distributed Transaction Coordinator) is enabled in the Control Panel -> Administrative Tools -> Component Services. Transactions must be enabled not only on the package level but also on each task you want included as part of the transaction. To have the entire package in a transaction set Transaction Option at the package level to Required and each task to Supported.

**Q. If you have a package that runs fine in Business Intelligence Development Studio (BIDS) but fails when running from a SQL Agent Job what would be your first guess on what the problem is?**

**Ans:**

The account that runs SQL Agent Jobs likely doesn’t have the needed permissions for one of the connections in your package. Either elevate the account permissions or create a proxy account.

To create a proxy account you need to first create new credentials with the appropriate permissions. Next assign those credentials to a proxy account. When you run the job now you will select Run As the newly created proxy account.

**Q. What techniques would you consider to add auditing to your packages? You’re required to log when a package fails and how many rows were extracted and loaded in your sources and destinations.**

**Ans:**

I like to create a database that is designated for package auditing. Track row counts coming from a source and which actually make it to a destination. Row counts and package execution should be all in one location and then optionally report off that database.

There are also third party tools that can accomplish this for you (Pragmatic Works BI xPress).

**Q. Demonstrate or whiteboard techniques you would use to for CDC (Change Data Capture)? Tell how you would write a package that loads data but first detects if the data already exists, exists but has changes, or is brand new data for a destination.**

**Ans:**

For small amounts of data I may use the Slowly Changing Dimension. More often than not the data is too large to use in such a slow transform.

I prefer to do a lookup on the key of the target table and rows that don’t match are obviously new rows that can be inserted. If they do match it’s possible they are updates or duplicates. Determine this by using a conditional split comparing rows from the target to incoming rows. Send updates to a staging table that can then be updated in an Execute SQL Task.

Explain that putting updates in a staging table instead of updating using the OLE DB Command is much better for performance because the Execute SQL Task performs a bulk operation.

**Q. Explain what breakpoints are and how you would use them.**

**Ans:**  
Breakpoints put pauses in your package. It’s a great tool for debugging a package because you can place a breakpoint on a task and it will pause the package based on execution events.

A reason in which I have used breakpoints is when I have a looping container and I want to see how my variables are changed by the loop. I would place a watch window on the package and type the variable name in. Set a break point on the container the stop after each iteration of the loop.

**Q. What are the main components involved in SSIS?**

**Ans:**

* SSIS is not improved version of DTS
* SSIS is completely redesigned and build from ground up using .NET code.
* SSIS is mainly divided into two parts.
* Data Transformation Pipeline (DTP) – Data Flow
* Data Transformation Runtime (DTR) – Control Flow
* In SQL SERVER 7 / 2000 the data flow is stronger than control flow but in SSIS both are in the same level

**Q. What is the work of DTP Engine?**

**Ans:**

* DTP consists of DTP Engine and DTP Object model
* DTP uses Data Adapters to connect source and destination
* DTP engine uses DTP Object Model which is nothing but an API.
* SSIS comes with adapters for SQL Server databases, XML, flat files, and other OLE DB–compliant data sources
* The job of the data adapters is to make connections to the data’s source and destination endpoints
* The job of the transformations is to move and optionally manipulate the data as it’s moved between the source and destination endpoints.

**Q. How the DTR works in SSIS?**

**Ans:**

* The DTR consists of the DTR engine and the DTR components.
* DTR components are objects that enable you to govern the execution of SSIS packages.
* The primary DTR components are packages, containers, and tasks.
* DTR engine stores package layout; runs packages; and provides debugging, logging, and event handling services.
* The DTR is accessed using the DTR object framework. The DTR run-time object framework is the API that supports the Integration Services Import/Export Wizard and the Integration Services Designer in addition to the command-line dtexec tool.

**Q. Can you explain the SSIS Architecture?**

**Ans:**

**Runtime engine**

The Integration Services runtime saves the layout of packages, runs packages, and provides support for logging, breakpoints, configuration, connections, and transactions.

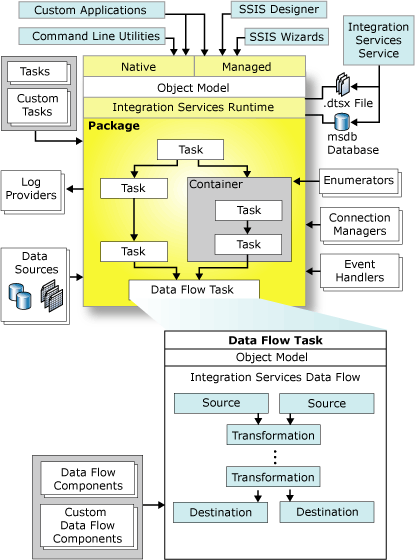
**API or object model**

The Integration Services object model includes managed application programming interfaces (API) for creating custom components for use in packages, or custom applications that create, load, run, and manage packages. Developer can write custom applications or custom tasks or transformations by using any common language runtime (CLR) compliant language.

**Integration Services service:** It is a Windows service, monitors running SSIS packages and manages the storage of packages.

**Data flow:** It contains a data flow engine that manages the data flow components. There are 3 types of

Data Flow components – Source components (which extracts the data from a system), Transformation components (performs transformations, modifications onto the extracted data) and Load components (which simply performs the data loading tasks into the destination systems). Besides the available data flow components, we can write our own custom data flow components to accomplish any custom requirements.



**SSIS – Performance Tuning**

**SSIS Interview Questions and Answers Part 3**

Here we are publishing series of posts on SSIS Interview questions and answers Part 3 for experienced and freshers. Below is the series 3.

**Q. How to quickly load data into sql server table?**

**Ans:**

Fast Load option: This option is not set by default so most developers know this answer as otherwise the load is very slow.

**Q. What are the fast load options available in SSIS?**

**Ans:**

The OLE DB Destination provides more than one way to load data in the destination (5 types of Data Access Mode). Use Fast Load option while loading data into the destination.

* **Data Access Mode** – It allows to define the method to upload data into the destination. The fast load option will use **BULK INSERT** statement instead of **INSERT** statement. If the fast load option is not selected then by default INSERT is used.
* **Keep Identity** – If selected, the identity values of source are preserved and the same are uploaded into the destination table. Else destination table will create its own identity values if there is any column of identity type.
* **Keep Nulls** – If selected, the null values of the source are preserved and are uploaded into the destination table. Else if any column has default constraint defined at destination table and NULL value is coming from the source for that column then in that case, default value will be inserted into the destination table.
* **Table Lock** – If selected, the TABLOCK is acquired on the table during data upload. It is the recommended option if table is not being used by any other application at the time of data upload as it removes the overhead of lock escalation.
* **Check Constraints** – Check constraints will always check for any constraint for the data that is coming through pipeline. It is preferable to uncheck this option if constraint checking is not required. This will reduce the overhead for the pipeline engine.
* **Rows per batch** – **RowsPerBatch** is the number of rows you would want in One Buffer. SSIS automatically sets this property based on the **RowSize** and **MaxBufferRows** property. The number of rows coming from the pipeline per batch can be defined by user. The default value is **-1** if it is kept blank. You can specify the no. of rows as a positive integer (**N**) so that the records will come as small segments or batches, each segment containing **N** no. of rows.
* **Maximum insert commit size** – You can specify the batch size that the OLE DB destination tries to commit during fast load operations; it actually splits up chunks of data as they are inserted into your destination. If you provide a value for this property, the destination commits rows in batches that are the smaller from either (a) the Maximum insert commit size, or (b) the remaining rows in the buffer that is currently being processed.
* **Network limitations:** You can transfer data as fast as your network supports. But use them efficiently; you can customize SSIS to use the maximum bandwidth of your network. You can set the **Packet Size** property of the connection manager to an integer value that suits you. The max value that you can insert is **32767.**

**Q. What are the lookup cache modes available and how to use them?**

**Ans:**

In 2008 we have three different cache modes for lookup transformations.

* Full Cache – Default
* Partial Cache
* No Cache

**Full Cache:**

The database is queried once during the pre-execute phase of the data flow. The entire reference set is pulled into memory. This approach uses the most memory, and adds additional startup time for your data flow. Lookup will not swap memory out to disk, so your data flow will fail if you run out of memory.

* **When to use Full cache mode**
* When you’re accessing a large portion of your reference set
* When you have a small reference table
* When your database is remote or under heavy load, and you want to reduce the number of queries sent to the server

**Partial Cache:**

In this mode, the lookup cache starts off empty at the beginning of the data flow. When a new row comes in, the lookup transform checks its cache for the matching values. If no match is found, it queries the database. If the match is found at the database, the values are cached so they can be used the next time a matching row comes in.

In 2008 there is a new **Miss Cache** feature that allows you to allocate a certain percentage of your cache to remembering rows that had no match in the database. This is useful in a lot of situations, as it prevents the transform from querying the database multiple times for values that don’t exist

* **When to use this cache mode**
* When you’re processing a small number of rows and it’s not worth the time to charge the full cache
* When you have a large reference table
* When your data flow is adding new rows to your reference table

**No Cache:**

As the name implies, in this mode the lookup transform doesn’t maintain a lookup cache (actually, not quite true – we keep the last match around, as the memory has already been allocated). In most situations, this means that you’ll be hitting the database for every row.

* **When to use this cache mode**
* When you’re processing a small number of rows
* When you have non-repeating lookup indexes
* When your reference table is changing (inserts, updates, deletes)
* When you have severe memory limitations

**Q. What are the different types of Transformations in SSIS?**

**Ans:**

**Non-Blocking –** No blocking

**Partial Blocking –** The downstream transformations wait for certain periods, it follows start then stop and start over technique

**Full Blocking:** The downstream has to be waiting till the data has been released from the upstream transformation.

**Non-blocking transformations**

* Audit
* Cache Transform
* Character Map
* Conditional Split
* Copy Column
* Data Conversion
* Derived Column
* Export Column
* Import Column
* Lookup
* Multicast
* OLE DB Command
* Percentage Sampling
* Script Component
* Slowly Changing Dimension

**Partial blocking transformations**

* Data Mining
* Merge
* Merge Join
* Pivot
* Unpivot
* Term Lookup

**Fully Blocking Transformations**

* Aggregate
* Fuzzy grouping
* Fuzzy lookup
* Row Sampling
* Sort
* Term Extraction

If you clearly observe Sort is a fully blocking transformation, so it’s better to sort your data using the SQL command in OLE DB Source instead of using sort transformation. Merge transform requires Sort but not Union All, so use Union All wherever possible.

**Q. Consider a scenario where I am using “Sort” transformation and my requirement is to after sort operation completed I have to remove all duplicate records. Duplicate records are defined based on sort values for example I am sorting result set based on three columns, when these 3 columns are having same values those rows are considered as duplicates. Now my question is which transformation we have to use to ignore all these duplicate records?**

**Ans:**

We need not use any specific transformation to remove duplicate records based on sort columns. There is a feature available at “Sort” transformation itself. We can find an option “Remove duplicate sort values” at the bottom left corner of SORT transformation editor. Just check that box.

**Q. How to avoid the sort transformation in SSIS?**

Ans:

Input datasets are to be in sorted order while dealing with the “Merge” or “Merge Join” transformations. To avoid the sort transformation we can directly use a “Query” with order by clause at data source. But remember we can do that when data source is OLEDB or Excel. When it comes to flat file we don’t have a choice but choose the best way to implement sort transformation.

For example if there is an aggregate required then apply aggregate before applying the sort transformation. If possible load flat file data into stage tables, apply sort at database level and load them at destination. So that we should have two data flows one is to load data from flat files to stage tables and other is to loading data into destination from stage tables hence we can use parallelism property.

**Q.** **How an SSIS package or a data flow knows that the input dataset / source dataset is in sorted order?**

**Ans:**

If the source dataset is in sorted order or we are using a query with order by at source we have to explicitly mention this information at “OLEDB” source.

There are two properties that we need to change at OLEDB source.

1. Open OLEDB source advanced editor

2. Goto tab “Input and Output Properties”

3. Select “OLEDB source OUTPUT”

4. In the properties select the value “True” for the property “IsSorted”

5. Expand Output Column list and select the column name and set “SortKeyPosition” value to one.

6. Repeat the step 5 for all columns in order by cluase by giving appropriate priority

**Q. What data providers supported for OLEDB connection manager for cache option when lookup transformation?**

**Ans:**

SQL Server

Oracle

DB2

**Q. From your customer side one of the architect asked you the below information. “I just wanted to know how many number of execution trees are being created for SSIS package which loads data on daily basis.”**

**How do we know this information?**

**Ans:**  
We can actually use custom log events to capture this information.

The log entry “PipelineExecutionTrees” helps us know about the execution trees created at run time. It includes lots of info for example number of rows stored in a buffer while executing a transformation etc.

For more info please have a look at below link

<http://msdn.microsoft.com/en-us/library/ms345174.aspx>

**Q. Do you know when an execution tree created and when it ends in a dataflow? Simply what is the scope of an execution tree?**

**Ans:**

The work to be done in the data flow task is divided into multiple chunks, which are called execution units, by the dataflow pipeline engine. Each represents a group of transformations. The individual execution unit is called an execution tree, which can be executed by separate thread along with other execution trees in a parallel manner. The memory structure is also called a data buffer, which gets created by the data flow pipeline engine and has the scope of each individual execution tree. An execution tree normally starts at either the source or an asynchronous transformation and ends at the first asynchronous transformation or a destination. During execution of the execution tree, the source reads the data, then stores the data to a buffer, executes the transformation in the buffer and passes the buffer to the next execution tree in the path by passing the pointers to the buffers.  
​​

**Q. While running SSIS package, after 15 min of execution it went to hung state. How you troubleshoot?**

**Ans:**

There are three common reasons that hold / hung the SSIS execution.

1. Resource Bottleneck: Memory / CPU / IO / Network
2. Blocking / Deadlock: Blocking happens at database level or In accessing a file or reading writing variables from script task.
3. Poor Performance query: If SSIS stops at Execute SQL Task look for query using inside the task and tune it.

Looking through above aspects one can identify the issue, based on that we can provide the resolution. If everything looks good but still SSIS is in hung state then check the latest service pack is applied if that’s also passed collect the hung dump file using ADPlus and contact Microsoft support center.

**Q. SSIS 2008 uses all available RAM, and after package completes Memory is not released?**

**Ans:**

This is not actually a problem. You have allowed SQL Server to use x amount of memory, so it does. SQL Server takes that memory as required, up to the limit set, but it does not release it. It can respond to request from OS, again read up on the fine details, but by default once it has got hold of some memory it will keep it even if it is not using it currently. The simple reason is that finding and taking hold of memory is quite expensive to do, so once it has it it keeps it and then any subsequent operations that need memory will have it available much faster. This makes perfect sense when you remember that SQL Server is a service application and more often than not runs on a dedicated machine.

**Q. What is the property “RunInOptimized”? How to set this property?**

**Ans:**

If this property is set to true then the SSIS engine ignore the unused/unmapped columns. Means it does not allocate memory to store data for those columns. At the compilation phase itself SSIS engine identifies what are the columns from source are using across the package, if it finds any columns are neither using nor mapping to destination, it simply ignores all those columns.

We can set this property at two levels “Project Level” and “Package Level”.

Project Level: From project properties → Debugging → RunIn\*\*\*\*\*. By default “FALSE”

Package Level: Can find in DataFlow properties. By default “TRUE”

**Q. Does using “RowCount” transformation affects the package performance?**

**Ans:**

Rowcount component is a synchronous component and it doesn’t actually do anything particularly resource intensive means the performance degradation of your package should be negligible.

We do use this component to capture the number of inserts, deletes and updates from each data-flow and then using “OnPost Execute” event this information would be written to a SQL Server table.

**Q. A SSIS 2008 package has been crashed due to low memory. How to resolve low memory issues with SSIS package?**

**Ans:**

1. Add more memory to the physical machine

2. Run SSIS package on a computer that is not running an instance of SQL Server

3. When SSIS and SQL instance on the same machine, balance the memory allocated to SQL Server instance using “MAX Server Memory” option.

4. Run SSIS package components in series instead of parallel

**Q. How to identify the SSIS processes?**

**Ans:**

SSIS run-time processes include the DTExec.exe process and the DTSHost.exe process.

**Q. How to enable containers continue to run even a task failed inside the container? Suppose you have an application, where we need to loop through some log table based on the IDs & load data into the destination. Now, in this scenario there might be the situation where some of the tasks in foreach loop container may fail. But your requirement is even though the inner tasks fail we should process the other sources which are available with us.**

Ans:

We can do this by updating the propagation property of a task / container to “False”. It means that the loop or sequence container ignores the failure of an internal task.

Assume we have designed a foreach loop container with a dataflow task. As per our requirement DFT is loading 100 files into database if DFT is failed to load 47th file it should skip the error and should continue to load from 48th file.

Steps to accomplish this are:

Select the Data Flow Task and goto eventhandler

Enable the OnError Event handler.

In the Event Handler tab, click on the “Show System Variables”.

Now select the “Propogate” property & change its value to “False”.

This will ensure that the parent control i.e. ForEach loop will not know about the error in the child task.

If incase the foreach loop container is having more than one task, instead of setting the property to all these tasks, add all these tasks to sequence container and change the “Propagate” property of sequence container.

**Note:** When this kind of situation comes to the single task instead of a loop we can actually use a property called “ForceExecutionValue” to “True” and give the value to “Force**d**ExecutionValue”“1”. This means that irrespective of execution result ssis engine forces the outcome to success.

**Q. What is ForceExecution property in SSIS component properties?**

**Ans:**

ForceExecution is a property of Controlflow elements in SSIS. If it is enabled to any of the element then ssis engine follows the execution result as per the given parameters. In other words to control the execution result of any control flow element we can use this property.

ForceExecutionValue: True or False

ForcedExecutionType: <Datatype>

ForcedExecutionValue: <Value>, we usually gives as 1 to make sute its true.

**Q. How to improve the performance of a SSIS package?**

**Ans:**

**1- Utilize parallelism:** It is easy to utilize parallelism in SSIS. All you need to do is to recognize which Data Flow Tasks (DFTs) could be started at the same time and set the control flow constraints of your package in the way that they all can run simultaneously.

**2- Synchronous vs. Asynchronous components:** A synchronous transformation of SSIS takes a buffer, processes the buffer, and passes the result through without waiting for the next buffer to come in. On the other hand, an asynchronous transformation needs to process all its input data to be able to give out any output. This can cause serious performance issues when the size of the input data to the asynchronies transformation is too big to fit into memory and needs to be transferred to HDD at multiple stages.

**3- Execution tree:** An execution tree starts where a buffer starts and ends where the same buffer ends. These execution trees specify how buffers and threads are allocated in the package. Each tree creates a new buffer and may execute on a different thread. When a new buffer is created such as when a partially blocking or blocking transformation is added to the pipeline, additional memory is required to handle the data transformation; however, it is important to note that each new tree may also give you an additional worker thread.

**4-OLE DB Command transformation:** OLE DB Command is a row-by-row transformation, meaning that it runs the command in it on each one of its input rows. This make sit to be damn too slow when the number of the rows goes up. The solution for boosting performance is to stage data into a temporary table and use Execute SQL Task outside that DFT.

**5-SQL Server Destination vs. OLE DB Destination:** There is multiple reason why to use OLE DB Destination and not use SQL Server Destination:

* OLE DB Destination is mostly faster,
* OLE DB Destination is a lot clearer when it fails (The error message is more helpful),
* SQL Server Destination works only when SSIS is installed on the destination server.

**6- Change Data Capture (CDC):** Try to reduce the amount of data to be transferred to the maximum level you can, and do it as close to the source as you can. A Modified On column on the source table(s) helps a lot in this case.

**7- Slowly Changing Dimension (SCD) transformation:** There is only one advice about SSIS’s Slowly Changing Dimension transformation, and that is get rid of it! The reasons are:

* It doesn’t use any cached data, and goes to the data source every single time it is called,
* It uses many OLE DB Command transformations,
* Fast Data Load is off by default on its OLE DB Destination.

**8. Choose the best way in designing Data flow between SQL and SSIS:** Remember SSIS is good at Row by Row operations where AS SQL is not. So depends on the situation design data flow using DFT components instead of executing a query using “Execute SQL Task”.

**9.** Use queries for selecting data rather than selecting a table and checking off the columns you want. This will reduce the initial record set before SSIS gets it rather than ignoring the fields

**10.** Carefully deal with your connections. By default, your connection manager will connect to the database as many times as it wants to. You can set the **RetainSameConnection** property so it will only connect once. This can allow you to manage transactions using an ExecuteSQL task and BEGIN TRAN / COMMIT TRAN statements avoiding the overhead of DTC.

**11.** While running the package with in BIDS ensure you set the package to run in optimized mode.

**12.** While loading data into destination tables it’s helpful to use the “Fast Load option”.

**13.** Wherever possible Consider aggregating and (un)pivotting in SQL Server instead doing it in SSIS package – SQL Server outperforms Integration Services in these tasks;

**14**. **Avoid manipulating large datasets using T-SQL** statements. All T-SQL statements cause changed data to write out to the transaction log *even if you use Simple Recovery Model*.

**15**. For large datasets, **do data sorts at the source** if possible.

**16**. **Use the SQL Server Destination** if you know your package is going to run on the destination server, since it offers roughly 15% performance increase over OLE DB because it shares memory with SQL Server.

**17**. **Increase the network packet size to 32767 on your database connection managers**. This allows large volumes of data to move faster from the source servers.

**18**. If using Lookup transforms, **experiment with cache sizes** – between using a Cache connection or Full Cache mode for smaller lookup datasets, and Partial / No Cache for larger datasets. This can free up much needed RAM.

**19.** Make sure “Lock Options” is using while loading very large datasets as bulk insert happens when it satisfies the below conditions.

a) Destination table is empty

b) Destination database recovery model is either simple or bulk insert

c) When table lock option specified

**20.** **Experiment with the DefaultBufferSize and DefaulBufferMaxRows** properties. You’ll need to monitor your package’s “Buffers Spooled” performance counter using Perfmon.exe, and adjust the buffer sizes *upwards* until you see buffers being spooled (paged to disk), then back off a little.

**21.** Do all setbased, aggregations and sort operations at source or destination using T-SQL.

**22.** If possible always use “NOLOCK” at source and “LOCK” at destination.

**23.** While loading to data warehouses try to disable the indexes while loading.

**Q. Can you explain the settings “Rows Per Batch” and “Maximum Insert Commit Size”?**

**Ans:**

These options are available at “OLEDB destination” in DFT.

**Rows per batch** – The default value for this setting is -1 which specifies all incoming rows will be treated as a single batch. You can change this default behaviour and break all incoming rows into multiple batches. The allowed value is only positive integer which specifies the maximum number of rows in a batch.

**Maximum insert commit size** – The default value for this setting is ‘2147483647’ (largest value for 4 byte integer type) which specifies all incoming rows will be committed once on successful completion. You can specify a positive value for this setting to indicate that commit will be done for those number of records. You might be wondering, changing the default value for this setting will put overhead on the dataflow engine to commit several times. Yes that is true, but at the same time it will release the pressure on the transaction log and tempdb to grow tremendously specifically during high volume data transfers.

**Q. Can you explain the DFT properties “DefaultBufferMaxRows” and “DefaultBufferMaxSize”?**

**Ans:**

The data flow task in SSIS (SQL Server Integration Services) sends data in series of buffers. How much data does one buffer hold? This is bounded by DefaultBufferMaxRows and DefaultBufferMaxSize, two Data Flow properties. They have default values of 10,000 and 10,485,760 (10 MB), respectively. That means, one buffer will contain either 10,000 rows or 10 MB of data, whichever is less.

You can adjust these two properties based on your scenario. Setting them to a higher value can boost performance, but only as long as all buffers fit in memory. In other words, no swapping please!

**Q. How can we connect to Oracle, DB2 and MySQL from SSIS?**

**Ans:**

**Oracle:**

Native OLEDB\Microsoft OLEDB Provider for Oracle

Native .Net providers\ or

.Net providers for OLEDB\

**MySQL:**

.Net Providers \ MySQL Data Provider Or

.Net Providers \ ODBC

**DB2:**

Native OLEDB\Microsoft OLEDB Provider for DB2

Native .Net providers\ ,

.Net providers\ ODBC OR

.Net providers for OLEDB\

**Q. Can’t we do FastLoad using “ADODotNet Destination”?**

**Ans:**

Yes, there is an option called “Use Bulk insert when possible” that needs to be tick at the time of mapping.

**Q. How to check whether SSIS transformations are using memory or spilling to Disk due to huge loads and asynchronous transformations?**

**Ans:**

A great way to check if your packages are staying within memory is to review the SSIS performance counter Buffers spooled, which has an initial value of 0; above 0 is an indication that the engine has started swapping to disk.

**Q. How to find how much of total memory allocated to SSIS and SQL Server?**

**Ans:**

Below are the performance counters which can help us in finding memory details.

**Process / Private Bytes (DTEXEC.exe)**: The amount of memory currently in use by Integration Services.

**Process / Working Set (DTEXEC.exe)**: The total amount of allocated memory by Integration Services.

**SQL Server: Memory Manager / Total Server Memory**: The total amount of memory allocated by SQL Server. Because SQL Server has another way to allocate memory using the AWE API, this counter is the best indicator of total memory used by SQL Server..

**Memory / Page Reads / sec**: Represents to total memory pressure on the system. If this consistently goes above 500, the system is under memory pressure.

**Q. See there is a scenario: We have a package which has to be open using BIDS / SSDT and has to be modified different elements. But from the location where the SSIS has to be open and modified is not having permissions to access the databases hence all connection managers and other location constraints will fail in validation phase and it takes lot of time to validate all of these connections. Do you have any idea how to control this validation phase?**

**Ans:**

Below are the different methods to switch off the package validation.

**Work OffLine:** There is a option called Work Offline. It doesn’t try to locate/validate packages. Once the package is ready then we have to uncheck the option Work Offline from SSI menu.

**Delay Validation:** Set the values to “True” to skip the validation while opening the package. It only applies for executables / control flow elements including package.

**ValidateExternalMetadata:** Property is set to be “True” for disabling the validation for dataflow components.

**SSIS Interview Questions and Answers for Experienced and Freshers**

Here we are publishing series of posts on SSIS Interview questions and answers Part 4 for experienced and freshers. Below is the series 4.

**Q.** **Difference between Union–all and Merge Join?**

**Ans:**

* Merge transformation can accept only two inputs whereas Union all can take more than two inputs
* Data has to be sorted before Merge Transformation whereas Union all doesn’t have any condition like that.

**Q. What is difference between Multicast and Conditional Split?**

**Ans:**

The Multicast transformation distributes its input to one or more outputs. This transformation is similar to the Conditional Split transformation. Both transformations direct an input to multiple outputs. The difference between the two is that the Multicast transformation directs every row to every output, and the Conditional Split directs a row to a single output

**Q. What is the difference between DTS and SSIS?**

**Ans:**

Well, nothing except both the Microsoft SQL Server Products.

Even though both are the ETL tools, we can differentiate if you are asked observations.

|  |  |  |
| --- | --- | --- |
| **S.no** | **DTS** | **SSIS** |
| 1 | Data Transformation Services | Sql Server Integration Services |
| 2 | Using Activex Script | Using Scripting Language |
| 3 | No Deployment wizard | Deployment wizard |
| 4 | Limited Set of Transformation available | Huge of Transformations available |
| 5 | Not Supporting BI Functionality | Completely supporting end to end process of BI |
| 6 | Single Tasks at a time | Multi Tasks run parallel |
| 7 | It is Un managed script | Managed by CLR |
| 8 | DTS can develop thru Enterprise manager | SSIS can thru Business Intelligence Development Studio (BIDS, nothing but new version of VS IDE) |
| 9 | We can deploy only at local server | It can be deployed using multiple server using BIDS |
| 10 | Designer contains Single Pane | SSIS designer contains 4 design panes: |
|  |  | a) Control Flow |
|  |  | b) Data Flow |
|  |  | c) Event Handlers & |
|  |  | d) Package Explorer. |
| 11 | No Event Hander | Event Handler Available |
| 12 | No Solution Explorer | Solution Explorer is available, with packages, connections and Data Source Views (DSV) |
| 13 | Connection and other values are static, not controlled at runtime. | It can be controlled dynamically using configuration |

**Q. What is the difference between Fuzzy Lookup and Fuzzy Grouping?**

**Ans:**

The Fuzzy Grouping task performs the same operations as the Fuzzy Lookup task but instead of evaluating input records against an outside reference table, the input set becomes the reference. Input records are therefore evaluated against other records in the input set and evaluated for similarity and assigned to a group.

**Q. What’s the difference between Control Flow and Data Flow?**

**Ans:**

**Control Flow:**

* Process Oriented
* Doesn’t manage or pass data between components.
* It functions as a task coordinator
* In control flow tasks requires completion (Success.,failure or completion)
* Synchronous in nature, this means, task requires completion before moving to next task. If the tasks are not connected with each other but still they are synchronous in nature.
* Tasks can be executed both parallel and serially
* Three types of control flow elements in SSIS 2005
* **Containers:**Provides structures in the packages
* **Tasks**: Provides functionality in the packages
* **Precedence Constraints**: Connects containers, executables and tasks into an ordered control flow.
* It is possible to include nested containers as SSIS Architecture supports nesting of the containers. Control flow can include multiple levels of nested containers.

**Data Flow**

* Streaming in nature
* Information oriented
* Passes data between other components
* Transformations work together to manage and process data. This means first set of data from the source may be in the final destination step while at the same time other set of data is still flowing. All the transformations are doing work at the same time.
* Three types of Data Flow components
* Sources: Extracts data from the various sources (Database, Text Files etc)
* Transformations: Cleans, modify, merge and summarizes the data
* Destination: Loads data into destinations like database, files or in memory datasets

**Q. What is difference between For Loop and For Each Loop?**

**Ans:**

A for loop will execute the tasks a specified number of times, in other words 10 times, or 25 times, and the number of times is specified in the definition of the container. You can use a variable to specify what that count is.

A for each loop will execute once for each item in the collection of items that it is looking at. A good example would be if users are putting an Excel file into a directory for import into the DB. You cannot tell ahead of time how many will be in the directory, because a user might be late, or there might be more than one file from a given user. When you define the ForEach container, you would tell it to execute for each \*.xls in the directory and it will then loop through, importing each one individually, regardless of how many files are actually there.

**Q. What is the difference between “OLEDB command” transformation and “OLEDB” destination in dataflow?**

**Ans:**

The OLE DB Command is a pretty simple transformation that’s available within a Data Flow that can run a SQL statement that can insert, update, or delete records to, in, or from a desired table. It’s good to keep in mind that this transformation initiates a row-by-row operation, so you may experience some performance limitations when dealing with large amounts of data.

OLEDB destination can use Fast Load options hence perform bulk uploads.

**Q. What is the Difference between merge and Merge Join Transformation?**

**Ans:**

* **Merge Transformation:**
* The data from 2 input paths are merged into one
* Work as UNION ALL
* Metadata for all columns needs to be same
* Use when merging of data from 2 data source
* **Merge Join Transformation:**
* The data from 2 inputs are merged based on some common key.
* Work as JOIN (LEFT, RIGHT OR FULL)
* Key columns metadata needs to be same.
* Use when data from 2 tables having foreign key relationship needs to present based on common key

**Q. What is the difference between “ActiveX Script” and “Script Task”?**

**Ans:**

* We could say “Script Task” is the latest version for the deprecated feature “ActiveX Script”. Both are used to implement extended functionality in SSIS.
* ActiveX script supports VBScript and JScript where as “Script Task supports “VB.Net and C#.Net”.
* “Script Task” is preferable as “ActiveX Script” has been removed in MSSQL 2012.
* Script Task is supported with integrated help, IntelliSense, debugging and can reference external Dotnet assembles.

**Q. What is the difference between “Script Task” and “Script Component”?**

**Ans:**

* Both are used to extend the native functionality of SSIS.
* “Script Task” is to enhance the functionality for control flow where as “Script Component” is to enhance the functionality for Data flow.
* “Script Task” can handle the execution of parts of the package where as “Script Component” can handle the data flow and transformations by processing row by row.

**Q. What is the difference between “Execute SQL Task” and “Execute T-SQL statement” Task?**

**Ans:**

* The Execute T-SQL Statement task takes less memory, parse time, and CPU time than the Execute SQL task, but is not as flexible.
* If you need to run parameterized queries, save the query results to variables, or use property expressions, you should use the Execute SQL task instead of the Execute T-SQL Statement task
* Execute T-SQL Statement task supports only the Transact-SQL version of the SQL language
* Execute SQL task supports many connection types but the Execute T-SQL Statement task supports only ADO.NET

**Q. What is the difference between “Data Conversion” and “Derived Column” transformations?**

**Ans:**

Data Conversion transformation is used o convert the datatype of a column. Same operation can be done using “Derived Column “transformation using typecast but derived column can also be used to add / create a new column by manipulating the existing column based on expressions.

We have to choose “Data Conversion” when the requirement is only intended to change the datatype. In other words “Data Conversion” is introduced just for developer convenience as it’s a direct method where as in “Derived Column” we have to use an expression to change the datatype of a column.

From 2008 in “Derived Column” transformation, datatype and length information is read only, when we create a new column or created from existing , data type would be assigned based on the expression outcome and the datatype is a read-only column.

To change the datatype we have to use “Data Conversion” transformation.

**Q. What is the difference between “Copy Column” and “Derived Column”?**

**Ans:**

Both transformations can add new columns.

Copy column can add new columns only through existing columns but coming to Derived column it can add new columns without any help from existing columns.

Derived Column can assign different transformations and data types to the new columns whereas Copy Column cannot.

Derived Column supports error output whereas Copy Column cannot.

**Q. What is the difference between UNIONALL and MERGE transformations?**

**Ans:**

The Merge transformation combines two sorted datasets into a single dataset. The rows from each dataset are inserted into the output based on values in their key columns.

The Merge transformation is similar to the Union All transformations. Use the Union All transformation instead of the Merge transformation in the following situations:

* The transformation inputs are not sorted.
* The combined output does not need to be sorted.
* The transformation has more than two inputs.

**Q. What is the difference between for loop and for each loop container?**

**Ans:**

The “For Loop Container” executes specified number of times like 10 times, 20 times until the specified condition is met.

The “Foreach Loop Container” runs over an iterator. This iterator can be files from a folder, records from ADO, data from a variable etc.

**Q. How to pass property value at Run time? How do you implement Package Configuration?**

**Ans:**

A property value like connection string for a Connection Manager can be passed to the pkg using package configurations. Package Configuration provides different options like XML File, Environment Variables, SQL Server Table, Registry Value or Parent package variable.

**Q.** **How would you deploy a SSIS Package on production?**

**Ans:**

* Using Deployment Manifest
* Create deployment utility by setting its property as true.
* It will be created in the bin folder of the solution as soon as package is build.
* Copy all the files in the utility and use manifest file to deploy it on the Prod.
* Using import/Export and scheduling a job

**Q. What are the new features added in SQL Server 2008 SSIS?**

**Ans:**

* **Improved Parallelism of Execution Trees**
* **.NET language for Scripting**
* **New ADO.NET Source and Destination Component**
* **Improved Lookup Transformation**
* **New Data Profiling Task**
* **New Connections Project Wizard**
* **DT\_DBTIME2, DT\_DBTIMESTAMP2, and DT\_DBTIMESTAMPOFFSET data types**

**Improved Parallelism of Execution Trees**: The biggest performance improvement in the SSIS 2008 is incorporation of parallelism in the processing of execution tree. In SSIS 2005, each execution tree used a single thread whereas in SSIS 2008, the Data flow engine is redesigned to utilize multiple threads and take advantage of dynamic scheduling to execute multiple components in parallel, including components within the same execution tree

**.NET language for Scripting:** SSIS 2008 is incorporated with new Visual Studio Tool for Application(VSTA) scripting engine. Advantage of VSTA is it enables user to use any .NET language for scripting.

**New ADO.NET Source and Destination Component**: SSIS 2008 gets a new Source and Destination Component for ADO.NET Record sets.

**Improved Lookup Transformation:** In SSIS 2008, the Lookup Transformation has faster cache loading and lookup operations. It has new caching options, including the ability for the reference dataset to use a cache file (.caw) accessed by the Cache Connection Manager. In addition same cache can be shared between multiple Lookup Transformations.

**New Data Profiling Task**: SSIS 2008 has a new debugging aid Data Profiling Task that can help user analyze the data flows occurring in the package. The Data Profiling Task can help users to discover the coerce of these errors by giving better visibility into the data flow.

**New Connections Project Wizard:** One of the main usability enhancements to SSIS 2008 is the new Connections Project Wizard. The Connections Project Wizard guides user through the steps required to create source and destinations.

**DT\_DBTIME2, DT\_DBTIMESTAMP2, and DT\_DBTIMESTAMPOFFSET data types** – facilitate data type mapping to equivalent T-SQL date/time data types introduced in SQL Server 2008. Their primary purpose is to provide support for more accurate time measurements.

**Q. What are Synchronies and Asynchronous transformations in SSIS?**

**Ans:**

**Synchronizes Transformations:**

A synchronous transformation processes incoming rows and passes them on in the data flow one row at a time. Output is synchronous with input, it occurs at the same time. Therefore, to process a given row, the transformation does not need information about other rows in the data set. When a transform can modify the row in place so as to not change the physical layout of the result set, it is said to be a synchronous transformation. The output of a synchronous component uses the same buffer as the input and does not require data to be copied to a new buffer to complete the transformation. Reuse of the input buffer is possible because the output of a synchronous component usually contains the same number of records as the input;

An example of a synchronous transformation is the Data Conversion transformation. For each incoming row, it converts the value in the specified column and sends the row on its way. Each discrete conversion operation is independent of all the other rows in the data set.

**Asynchronous Transformations:**

The output buffer or output rows are not in sync with the input buffer; output rows use a new buffer. In these situations it’s not possible to reuse the input buffer because an asynchronous component can have more, the same or less output records than input records.

* The component has to acquire multiple buffers of data before it can perform its processing. An example is the Sort transformation, where the component has to process the complete set of rows in a single operation.
* The component has to combine rows from multiple inputs. An example is the Merge transformation, where the component has to examine multiple rows from each input and then merge them in sorted order.
* There is no one-to-one correspondence between input rows and output rows. An example is the Aggregate transformation, where the component has to add a row to the output to hold the computed aggregate values.

Asynchronous components can further be divided into the two types described below:

* **Partially Blocking Transformation** – the output set may differ in terms of quantity from the input set. Thus new buffers need to be created to accommodate the newly created set.
* **Blocking Transformation** – a transformation that must hold one or more buffers while it waits on one or more buffers, before it can pass that buffer down the pipeline. All input records must read and processed before creating any output records. For example, a sort transformation must see all rows before sorting and block any data buffers from being passed down the pipeline until the output is generated.

**Note:**

Synchronous components reuse buffers and therefore are generally faster than asynchronous components

**Q.** **Any Idea About execution tree?**

**Ans:**

At run time, the data flow engine breaks down Data Flow task operations into execution trees. These execution trees specify how buffers and threads are allocated in the package. Each tree creates a new buffer and may execute on a different thread.

Execution trees are enormously valuable in understanding buffer usage. They can be displayed for packages by turning on package logging for the Data Flow task

**Q.** **Where are SSIS package stored in the SQL Server?**

**Ans:**

* SQL Server 2000: MSDB..sysdtspackages
* SQL Server 2005: MSDB..sysdtspackages90
* SQL Server 2008: MSDB..sysssispackages

Stores the actual content and the following tables do the supporting roles.

* Sysdtscategories
* sysdtslog90
* sysdtspackagefolders90
* sysdtspackagelog
* sysdtssteplog
* sysdtstasklog

2008:

* sysssispackagefolders
* sysssislog

**Q. How to achieve parallelism in SSIS?**

**Ans:**

Parallelism is achieved using MaxConcurrentExecutable property of the package. Its default is -1 and is calculated as number of processors + 2.

**Q. Differences between dtexec.exe and dtexecui.exe**

**Ans:**

Both dtexec.exe and dtexecui.exe execute SSIS packages in the same manner. The difference is that dtexecui provided a graphical user interface to construct the command line arguments for dtexec. The command string that is generated with dtexecui can be used as command line arguments to dtexec.

**Q. Demonstrate or whiteboard how you would suggest using configuration files in packages. Would you consider it a best practice to create a configuration file for each connection manager or one for the entire package?**

**Ans:**

There should be a single configuration file for each connection manager in your packages that stores their connection string information. So if you have 6 connection managers then you have 6 config files. You can use the same config file across all your packages that use the same connections.

If you have a single config file that stores all your connection managers then all your packages must have contain the connection managers that are stored in that config file. This means you may have to put connection managers in your package that you don’t even need.

**Q. Demonstrate or whiteboard using a loop in a package so each file in a directory with the .txt extension is loaded into a table. Before demonstrating this tell which task/container accomplishes this and which enumerator will be used.**

**Ans:**

This would require a Foreach Loop using the Foreach File Enumerator. Inside the Foreach Loop Editor you need to set a variable to store the directory of the files that will be looped through. Next select the connection manager used to load the files and add an expression to the connection string property that uses the variable created in the Foreach Loop.

**Q. What techniques would you consider to add notification to your packages? You’re required to send emails to essential staff members immediately after a package fails.**

**Ans:**

This could either be set in the SQL Agent when the package runs or actually inside the package you could add a Send Mail Task in the Event Handlers to notify when a package fails.

There are also third party tools that can accomplish this for you (Pragmatic Works BI xPress).

**Q. Have you used SSIS Framework?**

**Ans:**

This is common term in SSIS world which just means that you have templates that are set up to perform routine tasks like logging, error handling etc. Yes answer would usually indicate experienced person, no answer is still fine if your project is not very mission critical.

**Q. How many difference source and destinations have you used?**

**Ans:**

It is very common to get all kinds of sources so the more the person worked with the better for you. Common ones are SQL Server, CSV/TXT, Flat Files, Excel, Access, Oracle, MySQL but also Salesforce, web data scrapping.

**Q. What configuration options have you used?**

**Ans:**

This is an important one. Configuration should always be dynamic and usually is done using XML and/or Environment Variable and SQL Table with all configurations.  
**Q. How do you apply business rules in SSIS (Transformations….Specific calculations but also cleansing)?**

**Ans:**

Some people use SSIS only to extract data and then go with stored procedures only….they are usually missing the point of the power of SSIS. Which allow creating “a flow” and on each step applies certain rules this greatly simplifies the ETL process.

**Q. Give example of handling data quality issues?**

**Ans:**

Data Quality is almost always a problem and SSIS handles it very well. Examples include importing customers from different sources where customer name can be duplicates. For instance you can have as company name: SQL Server Business Intelligence but also SQL Server BI or SQL Server BI LTD or SQL Server BI Limited or inteligence (with one l). There are different ways to handle it. Robust and time consuming is to create a table with or possible scenarios and update it after each update. You can also use fuzzy grouping which is usually easy to implement and will make usually very good decisions but it is not 100% accurate so this approach has to be justified.

Other typical quality issues are nulls (missing values), outliers (dates like 2999 or types like 50000 instead of 5000 especially important if someone is adjusting the value to get bigger bonus), incorrect addresses and these are either corrected during ETL, ignored, re-directed for further manual updates or it fails the packages which for big processes is usually not practiced.

**Q. When to use Stored Procedures?**

**Ans:**

This one is very important but also tricky. ALL SSIS developers have SQL Server background and that is sometime not very good if they use SQL not SSIS approach.

Let’s start with when you typically use SPs. This is for preparing tables (truncate), audit tasks (usually part of SSIS framework), getting configuration values for loops and a few other general tasks.  
During ETL extract you usually type simple SQL because it comes from other sources and usually over complication is not a good choice (make it dynamic) because any changes usually affect the package which has to be updated as well.

During Transformation phase (business rules, cleaning, core work) you should use Transformation tasks not Stored procedures! There are loads of tasks that make the package much easier to develop but also a very important reason is readability which is very important for other people who need to change the package and obviously it reduces risks of making errors. Performance is usually very good with SSIS as it is memory/flow based approach. So when to use Stored Procedures for transformations? If you don’t have strong SSIS developers or you have performance reasons to do it. In some cases SPs can be much faster (usually it only applies to very very large datasets). Most important is have reasons which approach is better for the situation.

**Q. What is your approach for ETL with data warehouses (how many packages you developer during typical load etc)?**

**Ans:**

This is rather generic question. A typical approach (for me) when building ETL is to. Have a package to extract data per source with extract specific transformations (lookups, business rules, cleaning) and loads data into staging table. Then a package do a simple merge from staging to data warehouse (Stored Procedure) or a package that takes data from staging and performs extra work before loading to data warehouse. I prefer the first one and due to this approach I occasionally consider having extract stage (as well as stage phase) which gives me more flexibility with transformation (per source) and makes it simpler to follow (not everything in one go). So to summarize you usually have package per source and one package per data warehouse table destination. There are might be other approach valid as well so ask for reasons.

**Q. What is XMLify component?**

**Ans:**

It is 3rd party free component used rather frequently to output errors into XML field which saves development time.  
**Q. What command line tools do you use with SSIS?**

**Ans:**

dtutil (deployment), dtexec (execution), dtexecui (generation of execution code)

**Q. What is data cleansing?**

**Ans:**

Used mainly in databases, the term refers to identifying incomplete, incorrect, inaccurate, irrelevant, etc. parts of the data and then replacing, modifying, or deleting this dirty data.

**Q. Any Idea what is ETI?**

**Ans:**

Yes! ETI (Error Tolerant Index) is a technique used in Fuzzy Lookup / Fuzzy Grouping for data cleansing operation. The ETI is a decomposition of the field values contained within a reference table of values into smaller tokens is nothing but a match index.

For example, instead of searching for a street address that contains the value “112 Sunny Vail Ln.”, smaller components of the reference value might be used, such as “sunn”, “nyva”, and “112”.

These individual words are called Tokens, and all tokens in a index are divided using some special character and search with the reference table.

http://udayarumilli.com/wp-content/uploads/2015/05/word-image1.png

**Q. What is Fuzzy Lookup? Can you demonstrate it?**

**Ans:**

Fuzzy lookup transformation is data cleaning task that helps to clean the incoming data with the reference table with the actual value. This transformation tries to find the exact or similar value as a result. The result data set is also depends on the fuzzy matching configuration in the fuzzy lookup transformation task. Fuzzy lookup task will be more helpful when you have data typo issues in the source data.

Fuzzy Lookup transformation creates temporary objects, such as tables and indexes in the SQL Server TempDB. So, make sure that the SSIS user account has sufficient access to the database engine to create and maintain this temporary table. Fuzzy lookup transformation has 3 features.

* Defining maximum number of matches to return to output – It starts with 1 and that is the recommended.
* Token delimiters – It has a set of predefined delimiters and we can also add our’s
* Similarity score – It is the fuzzy algorithm input to match the score with the input row and reference row. This value is between 0 and 1. higher the value is the accurate the result. It is usually 0.60 is the best value for similarity score.

**Q. What shape would you use to concatenate two input fields into a single output field?**

**Ans:**

Pivot transformation

**Q. What is the Multicast Shape used for?**

**Ans:**

The Multicast transformation distributes its input to one or more outputs. This transformation is similar to the Conditional Split transformation. Both transformations direct an input to multiple outputs. The difference between the two is that the Multicast transformation directs every row to every output, and the Conditional Split directs a row to a single output

**Q. What types of things can I pass between packages in SSIS?**

**Ans:**

We can pass Variables primarily between packages. Within a variable we can pass them as any type that is available. So if you were to create an object variable, although memory consuming, we could potentially pass a table that is in memory. Granted, in SQL Server 2012 (Denali) this is much, much easier now with parameters. Actually, this was almost a relief in a way. Configuring packages to consume parent variables was a time consuming and in some cases, confusing situation when many variables were in the process.

**Q. How to accomplish incremental loads? (Load the destination table with new records and update the existing records from source (if any updated records are available)**

**Ans:**

There are few methods available:

* You can use **Lookup Transformation** where you compare source and destination data based on some id/code and get the new and updated records, and then use **Conditoional Split** to select the new and updated rows before loading the table. However, I don’t recommend this approach, especially when destination table is very huge and volume of delta is very high.
* Use Execute SQL Task and with Staging table
* Find the Maximum ID & Last ModifiedDate from destination and store in package variables. (Control Flow)
* Pull the new and updated records from source and load to a staging table (A dataload table created in destination database) using above variables.(Data Flow)
* Insert and Update the records using Execute SQL Task (Control Flow)
* Use the feature CDC (Change Data Capture) from SQL Server 2008
* Use Conditional split to split data for Inserts. Updates and Deletes
* For inserts redirect to a OLEDB Destination
* For Updates and Deletes redirect using a OLEDB Command transformation

**Q. How can you enable the CDC for a table?**

**Ans:**

To enable CDC to a table first the feature should be enabled to the corresponding database. Both can be done using the below procs.

exec sys.sp\_cdc\_enable\_db\_change\_data\_capture

sys.sp\_cdc\_enable\_table\_change\_data\_capture

**Q. How can you debug Dataflow?**

**Ans:**

Microsoft Integration Services and the SSIS Designer include features and tools that you can use to troubleshoot the data flows in an Integration Services package.

* SSIS Designer provides data viewers.
* SSIS Designer and Integration Services transformations provide row counts.
* SSIS Designer provides progress reporting at run time.
* Redirect to specified points using error output

**Q. How to debug control flow?**

**Ans:**

* Integration Services supports breakpoints on containers and tasks.
* SSIS Designer provides progress reporting at run time.
* Business Intelligence Development Studio provides debug windows.

**Q. What can you tell me about Ralph Kimball?**

**Ans:**

Ralph Kimball is an author on the topic of data warehousing and BI. He has been regarded as one of the original architects of data warehousing. Kimball has always had the firm belief that data warehouses should fast and understandable. Oh, and he developed this whole methodology of dimensional modeling. There is that. (It’s also probably a good idea to know the basic idea and structure of dimensional modeling)

**Q.** **Are you familiar with Package Configurations?**

**Ans:**  
Yes. Recently I was working on a project where we used the SQL Server Table package configuration to store values for the package parameters. That allowed me to build a GUI for the users to update the package variables each month with new values.  
**Q. Have you ever used the XML package configuration?**

**Ans:**

Yes. In fact, that is the method we use for storing the connection string used by the sql server table package configuration for the project I just mentioned. We have a dev/production environment, so using an xml file with the connection string (and pointing to that XML file from an environment variable) makes it easy to switch between the two servers.