Stuff

This blog Contains Useful information collected from notes, documents, forums or blogs from the Internet

Wednesday, April 17, 2013

Difference between sequential file stage and Data set stage?

1) When you use sequential file as Source, at the time of Compilation it will convert to native format from ASCII.where as, when you go for using datasets conversion is not required. Also, by default sequential files we be Processed in sequence only. Sequential files can accommodate up to 2GB only. Sequential files does not support NULL values.All the above can me overcome using dataset Stage,but selection is depends on the Requirement.suppose if you want to capture rejected data in that case you need to use sequential file or file set stage.

2)Sequential file is used to Extract the data from flat files and load the data into flat files and limit is 2GB.Dataset is a intermediate stage and it has parallelism when load data into dataset and it improve the performance.

3)Data set mainly consists of two files.

a)Descriptor file which consists of Metada,data location but not actual data itself
b)Data file contains the data in multiple files and one file file per partition.

4)orchadmin command is used to delete the datasets where as rm unix command is used to remove the flat files.

Complete information about orchadmin can be found in the below link-orchadmin

at 7:34 AM No comments: Posted by Vamsi Krishna

Labels: Datastage-Stages

Tuesday, April 16, 2013

COLUMN GENERATOR STAGE

1)The Column Generator Stage is a development/debug stage.

2)It can have a single input link and a single output link.

About Me

Vamsi Krishna

I am Vamsi Krishna, working for a software company in chennai, India. I love to work on Unix,Datastage and Database.I constantly keep trying things on Datastage.

View my complete profile

3)The Column Generator adds columns to incoming data and generates mock data for these columns for each data row processed.This is useful for testing a job when no real test data is available.

at 10:14 AM 1 comment: Posted by Vamsi Krishna

Labels: Datastage-Stages

ROW GENERATOR STAGE

1)The Row Generator Stage is a development/debug stage. The stage has no input links and only one output link.

2)The row generator stage generates a set of mock data fitting the specified metadata. This is useful for testing a job when no real test data is available.

3)Metadata can be specified using a schema file.

at 10:10 AM No comments: Posted by Vamsi Krishna

Labels: Datastage-Stages

Sort stage

1)Sort stage is a processing stage used to perform sorting operations on input data.

2)Need for Sorting:
Some stages require sorted input
ex- Join, merge stages

3)Sort stage requires a 'key' to be specified by which the sort is performed.Multiple sort keys can be specified

4)Sort operation is **performed partition wise**.To sort a complete set of data, you should change the Sort Stage execution mode to sequential.

5)There are two ways Sort can be performed in Datastage :

a)Within stages On input link Partitioning tab, set partitioning to anything other than Auto

b)In a **separate Sort stage which has more options** like Allow duplicates,case sensitive,sort order(ascending / descending) etc.

By default, both methods use the tsort operator which can be identified in Job score

6)Partitioning keys **are often different** than Sorting keys

Keyed partitioning (e.g.Hash) is used to group related records into the same partition where as Sort keys are used to establish order within each partition

at 10:07 AM No comments: Posted by Vamsi Krishna

Labels: Datastage-Stages

Modify stage

1)Modify stage is a processing stage that alters the record schema of the input data

2)Modify stage can have a single input and a single output link.

3)Modify stage can also be used to handle NULL values, string, date,

- dsxfiles (3)
- Job modified date (1)
- Job type (1)
- Lookup (1)
- Orchadmin (1)
- Other (3)
- sequence (1)
- Server Job (1)
- Server Routine (2)
- Transformer (2)
- Unix (2)

time and timestamp manipulation functions.

4)Modify stage is a native parallel stage and has performance benefits over the Transformer stage

at 10:04 AM No comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages

Sunday, April 14, 2013

Funnel stage

1)The Funnel stage is a processing stage that combines multiple inputs into a single output. It can have any number of inputs and a single output link.

2)The metadata for all input data sets must be identical.

3)Funnel can be operated in **three Modes and default Mode** is Continuous

a)Continuous:

1)Combines the records of the input link in no guaranteed order.

2)It takes one record from each input link in turn. If data is not available on an input link, the stage skips to the next link rather than waiting.

b)Sort Funnel:
Combines the input records in the order defined by the value(s) of one or more key columns and the order of the output records is determined by these sorting keys.

c)Sequence: Copies all records from the first input link to the output link, then all the records from the second input link and so on.

at 11:19 AM No comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages

Merge stage

1)Merge stage is a processing stage, which can have any number of input links and one output link, with same number of reject links as there are update links.

2)The input datasets to the Merge stage must be key partitioned and sorted.This ensures that rows with the same key column values are located in the same partition and will be processed by the same node.

3)Merge stage combines master data with one or more updates link data where the keys match.

4)Master and update links must have duplicate free data to ensure proper results.If the input data is not duplicate-free, the output generated will be improper.

5)Check link ordering to make sure the master and update links are proper otherwise the output generated will be improper

at 11:14 AM No comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages

Lookup Stage

1)The 'Lookup' stage is a processing stage that can have more than one input links and one output link, as well as one reject link.

2)Lookup Failure options
Continue, Drop, Fail, Reject

3)If the lookup fails to find a matching key column, **one of these actions can be taken:**

Fail: the lookup Stage reports an error and the job fails immediately. This is the default.

Drop:The input row with the failed lookup(s) is dropped

Continue:The input row is transferred to the output, together with the successful table entries.The failed table entry(s) are not transferred, resulting in either default output values or null output values depends on datatype.

Reject:The input row with the failed lookup(s) is transferred to a second output link, the reject link.

4)Sparse **lookup can be used if the input data** is smaller than the reference data.

5)Joins **have better performance when the reference data is huge**. Avoid lookups in such cases.

6)Lookup Stage **does not need sorted input data** where as for Join stage and Merge stage input data should be sorted.

7)Please find the below link to find the difference between Normal Lookup and sparse Lookup-NormalvsSparse

at 11:10 AM No comments: Posted by Vamsi Krishna

Labels: Datastage-Stages

Join Stage

1)The 'Join' stage is a processing stage that performs a join operation on two or more input data sets and then provides output in the form of one resultant data set.

2)The following four types of joins can be performed:
- Inner join
- Left outer join
- Right outer join
- Full outer join

3)The join stage supports 2 or more sorted input links and 1 output link

4)The join stage editor allows you to specify the keys on which join is performed. More than one key can be specified. Specified keys should have same name on all links.

5)No fail/reject option for missed matches

6)Link ordering is very important while using left or right outer join and also the input data on all links to join stage should be sorted.

Capturing unmatched records from a Join:
a)The Join stage does not provide reject handling for unmatched records.If unmatched rows must be captured,an OUTER join operation must be performed,so that when a match does not occur, the Join stage inserts Null value into the unmatched non-key columns provided non-key column is defined as nullable on the Join input links.

b)After Join Stage Use Transformer to filter Null records with the help of IsNull Built function

at 11:04 AM No comments: Posted by Vamsi Krishna

Labels: Datastage-Stages

Filter stage

1)Filter stage is used to transform (not modify) records from the input link based on a specific condition

2)Different criteria/requirements can be specified to direct data to different output links from the Filter stage.

3)This processing stage can have a single input link and any number of output links. It can also, optionally, have a reject link.

4)Switch stage **supports only 128 output links** whereas Filter stage can have any number of output links

Friday, April 12, 2013

When to use which stage in Datastage?

Copy STAGE-To drop a Particular column

Sort STAGE-sorting,generating Key change and similar to order by clause in oracle

Filter STAGE-Similar to where clause in oracle but we can not perform Join operation

Lookup,Join,Merge-To perform Join operation

Pivot Enterprise STAGE-Rows to columns and columns to Rows

External Filter STAGE-Filter the records by using Unix filter commands like Grep etc

MODIFY STAGE-Metadata conversion,Null Handling and similar to conversion functions in oracle

FUNNEL STAGE -Combining the multiple input data into a single output.Metadata should be same for all the inputs

REMOVE DUPLICATES STAGE-To remove duplicate values from a single sorted input.

ENCODE / DECODE STAGES:To encode/compress a data set using UNIX encoding commands like gzip etc

TRANSFORMER STAGE:
a)Filtering the Data(constraints)
b)Metadata conversion(Using Functions)
c)Rows to columns and columns to Rows(Using Stage variables)
d)Looping
e)Creating a counter(Using macros)-Counter using Transformer

SURROGATE KEY GENERATOR STAGE-To generate SURROGATE KEYs similar to oracle Database sequence

Aggregator Stage:To perform Group by Operations like max,min etc similar to Group by clause in oracle

ROW GENERATOR STAGE:To generate a set of mock data fitting the specified metadata when no real data is available

XML OUTPUT STAGE -To convert tabular data such as tables and sequential files to XML hierarchical structures.

SWITCH STAGE- It performs an operation similar to the switch statement in C and to filter the data

CHANGE CAPTURE STAGE-To identify Delta changes(inserts,updates,deletes etc) between two sources

oracle connector-To connect to the oracle Database.

at 10:23 AM 4 comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages, Other

Aggregator stage

1)Aggregator stage is a processing stage that can have one input and one output link. It classifies records from the input link into groups and computes the totals or performs specified aggregator functions for each group

2)Records can be grouped on one or more keys

3)In parallel environment, we need to be careful when partitioning. It can affect the result of the aggregator. If the records that fall in the same group are in different partitions, then the generated output will be wrong.Therefore, it is better to do Hash partition on grouping keys before the aggregator stage so that records with same keys will go to same partition.

4)In Aggregator two grouping methods(Hash and sort) are present.Please find the following link-Grouping Methods for more information about grouping methods in aggregator stage

at 10:18 AM No comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages

Copy Stage

1)Copy stage is a processing stage that can have a single input and any number of output links

2)Copy stage is used to copy a single input data set to a number of output data sets

3)This stage is generally used for following things

a)Columns can be dropped.
b)The order of the columns can be altered.

at 10:14 AM No comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages

Saturday, February 9, 2013

Difference between Normal lookup and Sparse lookup?

1)The first input link to lookup stage is called the 'Primary' link. Other links are called 'Lookup' links. When lookup links are from a stage that is other than a database stage, **all data from the lookup link is read into memory**. Then, for each row from the primary link, the lookup is performed. If the source of lookups is a database, there can be two types of lookups:

Normal lookup:
All the data from the database is read into memory, and then lookup is performed.

Sparse lookup: For each incoming row from the primary link, the SQL is fired on database at run time.

2)Sparse lookups can be used if the input data is smaller than the reference data.

at 3:33 AM 3 comments: Posted by Vamsi Krishna

+1 Recommend this on Google

Labels: Datastage-Stages, Lookup

Tuesday, February 5, 2013

Difference between Hash and sort grouping methods in Aggregator stage

Grouping Methods
Hash (default)
1)Calculations are made for all groups and stored in memory
2)Results are written out after all input has been processed so large memory is required when volume of input is high
3)Input does not need to be sorted
4)Useful when the number of unique groups is small
Sort
1)Requires the input data to be sorted by grouping keys
2)Only a single aggregation group is kept in memory so less memory is required
3)When a new group is seen, the current group is written out
4)Can handle unlimited numbers of groups

Conclusion-When the volume of input is high and is not predictable it is better to use Sort Method

at 8:39 AM No comments: Posted by Vamsi Krishna

Labels: aggregator, Datastage-Stages

Thursday, December 13, 2012

What is External Filter Stage?

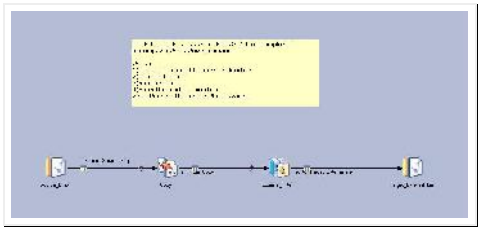
1)External filter stage is a processing stage.
2)In Datastage User can filter the Data using Unix Commands(sed,cut, cat, grep, head etc) with the help of External filter stage.
External filter stage allows us to run these commands during processing the data in the job
Job:
Now I want to use grep command to filter input file(employee file-emp.txt) with DEPTNO=10

Command : grep "10$"

Click the links below for input file(employee file-emp.txt) and dsx file for the Job-J_ExternalFilter

The Source File-emp.txt and dsx file for the Job-J_ExternalFilter

Job:



at 8:34 AM No comments: Posted by Vamsi Krishna

Labels: Datastage-Stages, dsxfiles, Unix

Tuesday, December 4, 2012

How to abort the Job based on Condition?

I/p

col
100
200
300
100

100

If any records come other than 100 in col column, I need to abort the job.

We can use Transfomer to abort the Job based upon certain condition

If Inlink.col<>"100" Then 'Y' Else 'N'=StgVar

Constraint:

StageVar='Y'

Abort After rows=1

at 8:35 AM No comments: Posted by Vamsi Krishna

Labels: Abort, Datastage-Stages, Transformer

Home Older Posts

Subscribe to: Posts (Atom)