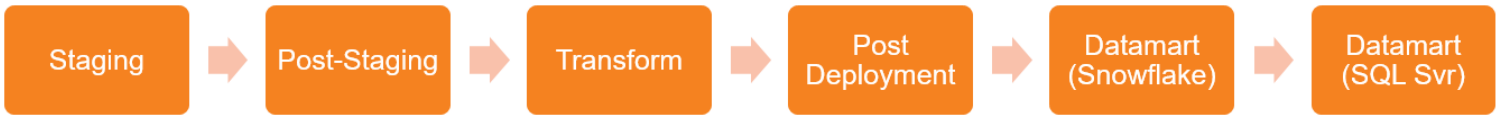
Talend Orchestration Overview

Last updated by | Archana Balachandran | Jun 3, 2020 at 8:33 AM EDT

Contents

Talend Orchestration

Orchestration Process Overview



The orchestration process consists of 6 steps:

1. **Staging**: Data from original sources such as SQL and Azure are staged into Snowflake tables in Staging database
2. **Post-Staging**: SnowSQL scripts stored in F:/ are executed to populate all the Helper tables in Snowflake Stagnig database.
3. **Transform**: SnowSQL scripts are executed to transform and load data from Snowflake Staging Database to tables in EDW Database.
4. **Post-deployment**: Merge scripts are executed to populate default values in EDW tables.
5. **Datamart (Snowflake)**: Scripts are executed to load data into Datamart_PA and Datamart_RM schemas in Snowflake EDW database
6. **Datamart (SQL server)**: Talend job is executed to load Datamart tables in SQL Server from Snowflake Datamart schemas.

Orchestration Setup

The orchestration process is carried out through some essential supporting processes.

Talend Job Designs:

All jobs that carry out the orchestration process is designed in Talend Studio, and then published to Talend Cloud for QA and execution.

Snowflake control tables:

- control.package table contains information on all the jobs that need to be executed by Talend
- control.packageexecution table records information on execution of each package (more details in next slide)

Snowflake Stored procedures:

- **recordpackageexecution** stored procedure creates an entry for the package being executed in the control.packageexecution table. Note that the stored procedure works only if the correct package name is specified in the control.package table and its set to Active=1. It's also important to note that at any given time, only one master process should be indicated as actively running in the packageexecution table. This stored procedure returns the packageexecutionID for the supplied package.

- **recordpackagesuccess** stored procedure closes the open entry for each package execution in control.packageexecution table, by entering the ExecutionEndTime and PackageExecutionStatus. This stored procedure takes the packageexecutionID and ExecutionStatusID as input.

Orchestration Driver Table: Control.Package

↓ PACKAGEID	PACKAGENAME	PACKAGEDESCI	PACKAGEFILENAME	PARENTPACKAGEID	PACKAGETYPEI	EXECUTIONORDER	TARGETTABLEI	CANRUNINPAR	ACTIVE
1	CoverysEDW_Extraction_Master	ETL of all da...	CoverysEDW_Extra...	NULL	1	1	NULL	1	0
2	CoverysEDW_Staging_Master	ETL of all da...	CoverysEDW_Stagi...	NULL	1	2	NULL	1	0
3	CoverysEDW_Transformation_Master	ETL of all da...	CoverysEDW_Trans...	NULL	1	3	NULL	1	1
4	CoverysEDW_CubeProcessing_Master	Process Cube	CoverysEDW_Cube...	NULL	1	4	NULL	1	0
5	CoverysEDW_CubeProcessingETLFramewo...	Process Cub...	CoverysEDW_Cube...	NULL	1	5	NULL	1	0
6	CoverysEDW_Staging_Process_IS4East_Vali...	IS4East row ...	CoverysEDW_Stagi...	24	3	127	NULL	1	0
7	CoverysEDW_Staging_Process_IS4West_Va...	IS4West row...	CoverysEDW_Stagi...	25	3	127	NULL	1	0
8	CoverysEDW_Transformation_Process_Xref...	XrefMapping...	F:\SnowSQL\Transf...	34	3	0	444	1	1
9	CoverysEDW_Transformation_Process_Rem...	Package tha...	CoverysEDW_Trans...	34	3	0	NULL	1	0
10	CoverysEDW_Transformation_Process_Upd...	RowStartDat...	CoverysEDW_Trans...	34	3	99	NULL	1	0
11	CoverysEDW_Transformation_Process_Upd...	RowStartDat...	CoverysEDW_Trans...	36	3	127	NULL	1	0
12	CoverysEDW_Extraction_Control_IS4East	IS4East Extr...	CoverysEDW_Extra...	1	2	7	NULL	1	0
13	CoverysEDW_Extraction_Control_IS4West	IS4West Extr...	CoverysEDW_Extra...	1	2	8	NULL	1	0
14	CoverysEDW_Extraction_Control_SharePoint	SharePoint E...	CoverysEDW_Extra...	1	2	9	NULL	1	0

Etlconfig.Control.package table: Acts as a driver table who's records determine how all six orchestration processes are run. This table maintains the hierarchy of master>control>process jobs using the packageID and parentpackageID columns.

- PackageID: Unique ID assigned to each job/package.
- PackageName: Unique name assigned to each job/package.

- PackageFileName: determines where the SnowSQL scripts are stored for SQL jobs.
- ParentPackageID: Indicates the parent job for that package.
- PackagetypeID: Indicates the type of package. 1= Master, 2 = Control, 3 = Process
- ExecutionOrder: Indicates the sequence in which jobs need to be run, and if jobs have the same sequence number, they can be run in parallel.
- Active: If this value is set to 1 for a package, it means that the package is included in the execution.

Orchestration Execution Table: Control.PackageExecution

PACKAGEEXECUTIONID	PARENTPACKAGEEXECUTIONID	PACKAGEID	PACKAGEEXECUTIONSTATUSID	EXECUTIONSTART	EXECUTIONEND
200258	NULL	2	2	2019-10-15 15:38:23.364	2019-10-15 15:53:12.535
200259	200258	17	2	2019-10-15 15:38:29.143	2019-10-15 15:53:09.566
200260	200259	175	2	2019-10-15 15:38:37.054	2019-10-15 15:39:04.544
200261	200259	174	2	2019-10-15 15:38:37.062	2019-10-15 15:39:25.398
200262	200259	176	2	2019-10-15 15:39:11.954	2019-10-15 15:39:40.072
200263	200259	177	2	2019-10-15 15:39:32.432	2019-10-15 15:39:58.024
200264	200259	178	2	2019-10-15 15:39:46.613	2019-10-15 15:40:08.038
200265	200259	179	2	2019-10-15 15:40:04.708	2019-10-15 15:40:26.529
200266	200259	180	2	2019-10-15 15:40:16.635	2019-10-15 15:40:59.329
200267	200259	181	2	2019-10-15 15:40:33.218	2019-10-15 15:41:26.231
200268	200259	182	2	2019-10-15 15:41:06.310	2019-10-15 15:41:26.713
200269	200259	183	2	2019-10-15 15:41:32.732	2019-10-15 15:41:51.860
200270	200259	184	2	2019-10-15 15:41:33.654	2019-10-15 15:41:47.687
200271	200259	185	2	2019-10-15 15:41:54.851	2019-10-15 15:42:38.970
200272	200259	186	2	2019-10-15 15:41:58.684	2019-10-15 15:42:22.144
200273	200259	187	2	2019-10-15 15:42:28.975	2019-10-15 15:42:40.641

Control.packageexecution table: Records the execution details of each package:

- PackageExecutionID: unique identifier assigned for that specific execution of a package.
- ParentPackageExecutionID: records the packageexecutionID of the parent package
- PackageExecutionStatusID: Indicates the final status of package execution. 1=Running, 2 = Completed successfully, 3=Failed.
- ExecutionStart and ExecutionEnd: Records the start and end time for each package execution.

Getting Package Details for Execution

```
SELECT PACKAGENAME
FROM package
where PARENTPACKAGEID =17
and Active=1
ORDER BY ExecutionOrder asc;
```

Results Data Preview

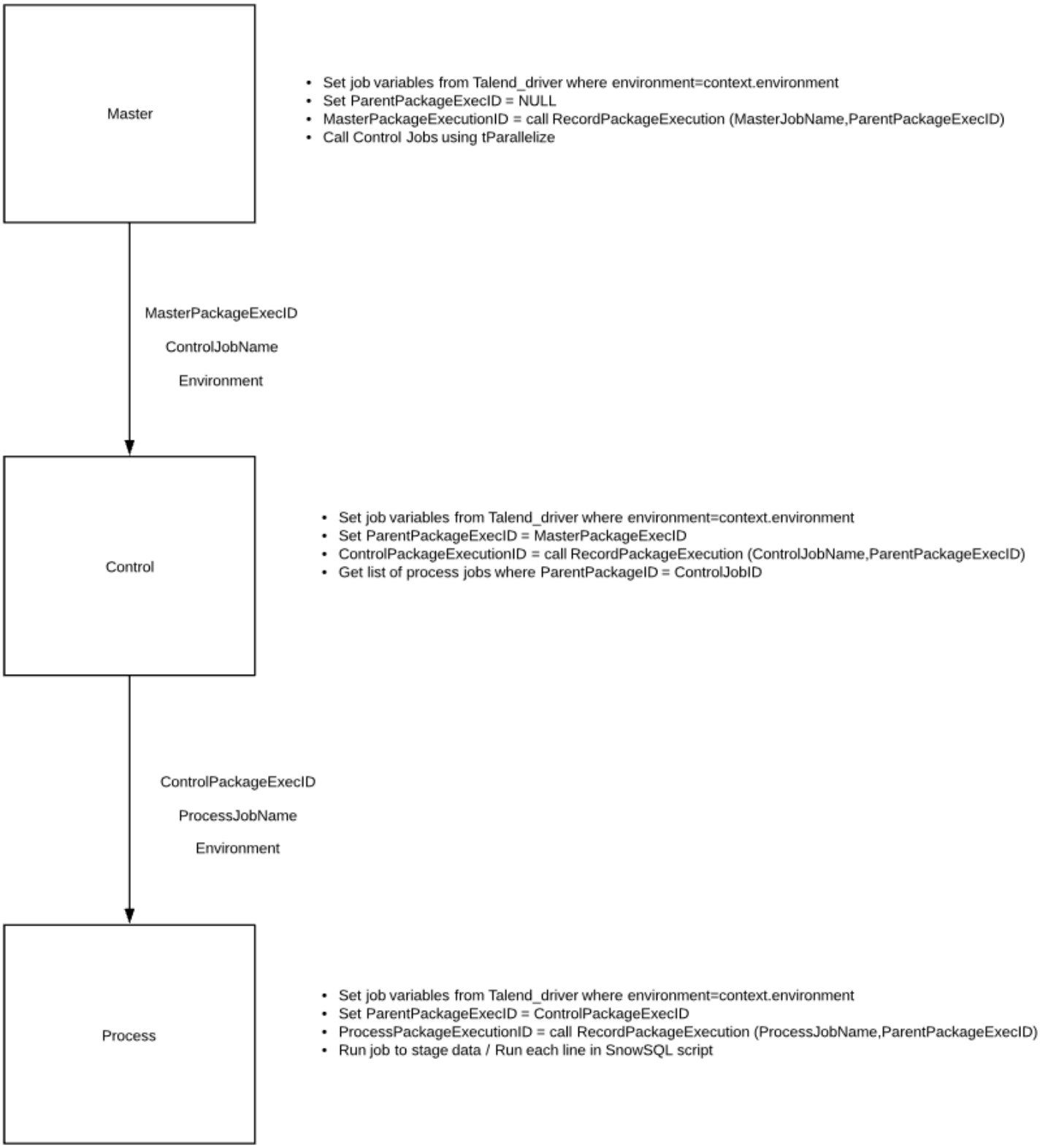
Query ID SQL 801ms 17 rows

Filter result... Copy

Row	PACKAGENAME
1	STG_ASW_Party_dbo_ADDRESS
2	STG_ASW_Party_dbo_PARTY
3	STG_ASW_Party_dbo_PARTY_ADDR
4	STG_ASW_Party_dbo_PARTY_BUS
5	STG_ASW_Party_dbo_PARTY_BUS_NM

- When the staging process begins, Talend issues the above query to Snowflake control.package table,
- And the list of packages to run and their executionorder is returned to talend.
- Talend then runs each job from the list, and makes an entry in control.packageexecution table.

Flow of control variables



- Control variables are: job names and job IDs as specified in the control.package table.
- For example, when the staging process starts, the staging master job name, along with 'NULL' is passed as arguments to the recordpackageexecution stored procedure. This will create an entry for

the master package execution in control.packageexecutiontable return the PackageExecutionID as MasterPackageExecutionID.

- The master package execution ID is then passed to the control job. When each control job is being executed, the control job name, along with the parent package's execution ID (MasterPckageExecutionID) is passed to the stored procedure.
- The control Package Execution ID returned by the last step is passed to each process package job that's executed.