**Introducción**

El objetivo de este documento es plasmar los resultados de las pruebas en las ejecuciones de la nueva versión de HIve (3.0.1) a través de Airflow (v.2.0.1) que fueron instalados en el cluster de AWS del que dispondremos para la nueva plataforma cloud.

**Descripción del trabajo realizado**

Se realizaron reiteradas ejecuciones en Airflow del DAG que ejecuta un período de la BDR (<https://zonda-airflow-dev.aws.ar.bsch/tree?dag_id=BDR_LOAD_Tables-Monthly>) considerándolo apropiado para testear ejecuciones por la variedad de desarrollos, volúmen de datos, cantidad y complejidad de querys que contiene.

Dado el upgrade entre la versión en Cloudera (Prod) y la existente actualmente en AWS (Desa) que incluye el cambio del motor de Map Reduce a Tez, algunas consultas no se ejecutaban correctamente a pesar de estar corriendo en Prod sin errores. Los mismos incluían mensajes que mostraban fallas a muy bajo nivel los cuales necesitaron de un análisis en profundidad para detectar qué cambios en el script eran requeridos.

Una vez detectado el error y el plan de acción para mitigarlo, se editaba el correspondiente archivo .hql alojado en el cluster Desa de forma local, es decir, sin pushear los cambios al repositorio para prevenir fallas no deseadas en el proceso productivo.

**Conclusiones y errores encontrados**

En reiteradas ocasiones, se detectaron errores que se resuelven teniendo en cuenta las siguientes consideraciones:

* A la hora de hacer llamados a la función NVL, la cual reemplaza los NULL por un valor indicado, **si el valor a reemplazar es de tipo numérico**, debe especificarse la parte decimal. Por ej:

Si tenemos:

**create** **temporary** **table** test\_temporal **as**

**select**

nvl('40067761','0') **as** cod\_nvl;

**insert** **into** bi\_corp\_staging.test\_hive\_aws

**partition** ( partition\_date= '2021-05-26' )

**select** cod\_nvl **from** test\_temporal;

Reemplazar por:

**create** **temporary** **table** test\_temporal **as**

**select**

nvl('40067761','0.0') **as** cod\_nvl;

**insert** **into** bi\_corp\_staging.test\_hive\_aws

**partition** ( partition\_date= '2021-05-26' )

**select** cod\_nvl **from** test\_temporal;

Cabe aclarar que los casos en que se produce el error son únicamente al momento de intentar hacer un insert en una tabla cuya columna destino es de alguno de los tipos de datos numéricos (Decimal, Double, Float).

* En el caso de tener campos que por default llevan NULL, se debe castear dicho NULL al tipo de dato String en los casos en que esté involucrada una tabla temporal. De lo contrario, el valor tomará el tipo de dato VOID y al no coincidir con el tipo declarado en el DDL de la tabla, fallará a la hora de hacer el Insert. Por ejemplo:

Si tenemos:

**create** **temporary** **table** test\_temporal **as**

**select**

**null** **AS** cod\_nulo;

**insert** **into** bi\_corp\_staging.test\_hive\_aws

**partition** ( partition\_date= '2021-05-26' )

**select** cod\_nulo **from** test\_temporal;

Reemplazar por:

**create** **temporary** **table** test\_temporal **as**

**select**

**cast(null as string)** **AS** cod\_nulo;

**insert** **into** bi\_corp\_staging.test\_hive\_aws

**partition** ( partition\_date= '2021-05-26' )

**select** cod\_nulo **from** test\_temporal;

* En caso de realizar consultas que involucren el uso de “UNION” o “UNION ALL” tener muy presente el tipado de todos los campos que intervienen en la union. En la versión anterior, Hive “dejaba pasar” algunos casos a pesar de haber diferencias en los tipos de datos que se querían unir. Actualmente dicha casuística es controlada y debe respetarse para evitar el error. Por ejemplo:

**insert** **into** tabla1

**select** nup **from** tabla2

**union** **all**

**select** nup **from** tabla3

Asumiendo que el campo “nup” de la tabla 2, no es String como el de la tabla1, debería resolverse haciendo el casteo correspondiente:

**insert** **into** tabla1

**select** nup **from** tabla2

**union** **all**

**select** **cast**(nup **as** **string**) **from** tabla3

**Anexo**

Adicionalmente, se recomienda ampliamente consultar la siguiente [documentación](https://docs.cloudera.com/HDPDocuments/HDP3/HDP-3.1.5/hive-overview/content/hive-semantics.html), en la cual se detallan las diferencias de sintaxis y semántica que tendremos que tener en cuenta dado el upgrade de versión del motor Hive (Estábamos en versión 1 en Cloudera y en AWS estaremos el 3.0.1)