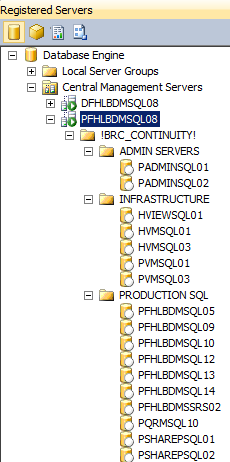
BRC Continuity Testing Documentation

Open SSMS (SQL Server Management Studio) and select the Central Management Server “PFHLBDMSQL08” and select the !BRC\_CONTINUITY! Group. Then Right Click and select new query.



Run the following Query against the entire !BRC\_CONTINUITY! Group:

SELECT

@@ServerName AS ServerName,

name AS DatabaseName,

state\_desc AS [Status]

FROM sys.databases

WHERE state\_desc <> 'ONLINE'

If you see any results listed, this means there are databases at the BRC that are in a state other than ONLINE and will need attention to fix the status of the database (i.e. restore a backup copy or put db ONLINE).

The next step would be to verify the status of the Tidal Enterprise Scheduler Jobs in the TESCache database located on PFHLBDMSQL05.

Run the following query to find job status:

SELECT

DTYPE,

ID,

NAME,

CASE

WHEN STATUS = 3 THEN 'HELD'

WHEN STATUS = 5 THEN 'TIMED OUT FOR DAY'

WHEN STATUS = 7 THEN 'AGENT UNAVAILABLE'

WHEN STATUS = 8 THEN 'AGENT DISABLED'

WHEN STATUS = 50 THEN 'LAUNCHED'

WHEN STATUS = 51 THEN 'ACTIVE'

END AS STATUSMESSAGE,

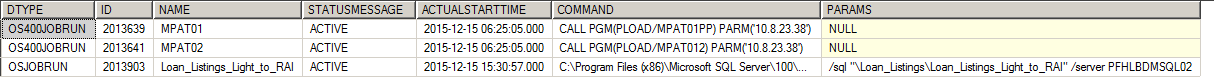
ACTUALSTARTTIME,

COMMAND,

PARAMS

FROM TESCache.TES.jobrun

WHERE Status IN (3, 5, 7, 8, 50, 51) and DTYPE<>'JOBRUNGROUP'



The script above is used to find any jobs that are in the following statuses: HELD, TIMED OUT FOR DAY, AGENT UNAVAILABLE, AGENT DISABLED, LAUNCHED, ACTIVE. You will then want to run an update command to place any found jobs into the TIMED OUT STATUS in order for the Tidal Application server to be brought back online to a clean state for jobs to be run.

Use the following update statement template and fill in the appropriate JOB IDs into the where clause to update the jobs to the status of TIMED OUT.

update TESCache.TES.jobrun set [Status]=108 where ID in ('','')

Any application servers still on with tidal agent may need to be rebooted or have the tidal agent service restarted in order to clear the agent job cache.

Run the following SQL Scripts on the Central Management Server for !BRC\_CONTINUITY! Group and review results to verify correct RunDates are configured for SSIS Jobs and that last modified dates are reflected correctly.

--Return Most Recent ETL Maintenance Process Log Entry Date

SELECT MAX(End\_DateTime) AS LastETLMaintProcessLogDateEntry FROM [ETL\_Maintenance].[dbo].[Processes\_Log]

--Return Business Date Settings from ETLMaintenance

SELECT \* FROM [ETL\_Maintenance].[dbo].[Process\_Business\_Dates]

Running the following SQL Script against the !BRC\_CONTINUITY! Group with the PFHLBDMSQL12 Server selected will return all of the SSIS Package ETL Maintenance configured RunDates for each SSIS Package including the ProcessID and Department responsible.

SELECT

CASE

WHEN b.Value = CAST(GETDATE() AS DATE) THEN 'JOB DATE MATCH'

ELSE 'DATE MISMATCH'

END AS DATEREVIEW,

a.[ID],

a.[Name],

a.[Description],

a.[Department],

b.Value AS ProcessRunDate

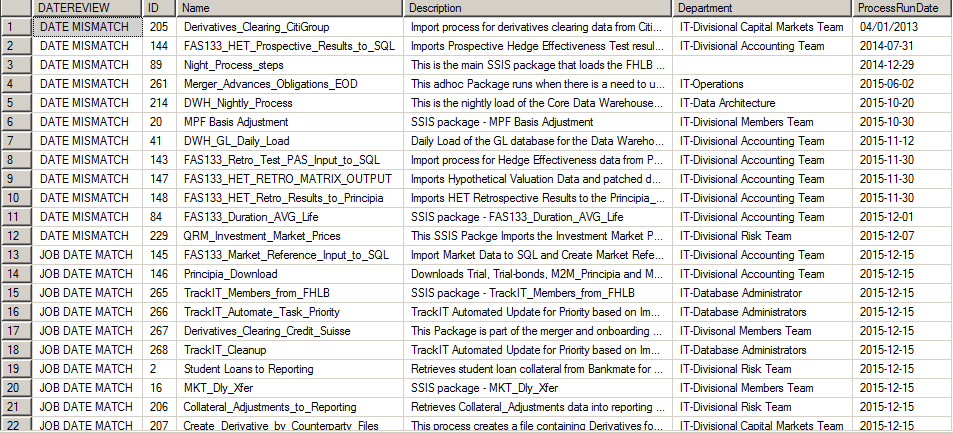
FROM [ETL\_Maintenance].[dbo].[Processes] a,

[ETL\_Maintenance].[dbo].[Process\_Settings] b

WHERE a.Id = b.Process\_ID

AND b.Name = 'RunDate'

ORDER BY b.Value ASC



Running the following SQL Script against the !BRC\_CONTINUITY! Group will give you a result set to review all databases on each server and their last modified date.

DECLARE @sqlString NVARCHAR(max)

DECLARE @union NVARCHAR(max)

SET @sqlString = ''

SET @union = ''

DECLARE @name nvarchar(50);

DECLARE crs CURSOR FOR

SELECT Name FROM sys.databases WHERE state = 0

OPEN crs

FETCH NEXT FROM crs INTO @name

WHILE @@FETCH\_STATUS = 0

BEGIN

SET @sqlString = @sqlString + @union

SET @sqlString = @sqlString + '

SELECT

TOP 1

@@servername as ServerName,''' + @name + ''' as DBName, modify\_date

FROM

[' + @name + '].sys.tables'

SET @union = ' UNION '

FETCH NEXT FROM crs INTO @name

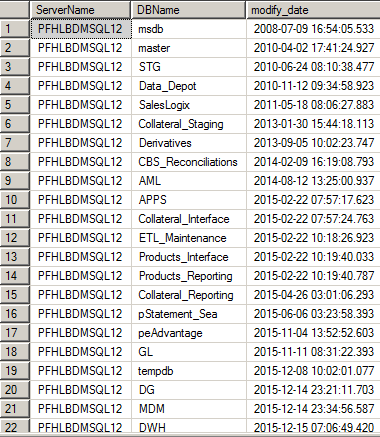
END

SET @sqlString = @sqlString + ' ORDER BY modify\_date ASC'

CLOSE crs;

DEALLOCATE crs;

EXEC(@sqlString)



Running the following Script against the !BRC\_CONTINUITY! Group with the PFHLBDMSQL14 Server selected, will give you the last ReportCreatedOn date for a spot check on several tables in the FHLB and QRM\_Source Databases on PFHLBDMSQL14.

--QRM\_SOURCE EXTRACT SCHEMA

select max(ReportCreatedOn) as QRM\_Source\_Extract\_Call\_Sched\_Date from QRM\_Source.extract.Call\_Sched

select max(ReportCreatedOn) as QRM\_Source\_Extract\_Call\_Rate\_Date from QRM\_Source.extract.Rate\_Sched

select max(ReportCreatedOn) as QRM\_Source\_Extract\_Amort\_Sched\_Date from QRM\_Source.extract.Amort\_Sched

select max(ReportCreatedOn) as QRM\_Source\_Extract\_QRM\_Data\_Date from QRM\_Source.extract.QRM\_Data

--QRM\_SOURCE

select max(ReportCreatedOn) as QRM\_Source\_QRM\_Advance\_Pricing\_Date from QRM\_Source.dbo.QRM\_Advance\_Pricing

--FHLB

select max(ReportCreatedOn) as FHLB\_CDS\_Fact\_EOD\_Date from FHLB.dbo.CDS\_Fact\_EOD

select max(ReportCreatedOn) as FHLB\_Advances\_Fact\_EOD\_Date from FHLB.dbo.Advances\_Fact\_EOD

select max(ReportCreatedOn) as FHLB\_Commitments\_FACT\_EOD\_Date from FHLB.dbo.Commitments\_FACT\_EOD

select max(ReportCreatedOn) as FHLB\_TBA\_Trades\_Date from FHLB.dbo.TBA\_Trades

select max(ReportCreatedOn) as FHLB\_MPFBAL\_FACT\_EOD\_Date from FHLB.dbo.MPFBAL\_FACT\_EOD

select max(ReportCreatedOn) as FHLB\_Accounts\_Receivable\_Date from FHLB.dbo.Accounts\_Receivable

select max(ReportCreatedOn) as FHLB\_MPF\_Loan\_DM\_Date from FHLB.dbo.MPF\_Loan\_DM