# DWYT-18 testing cycle

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Notes |
| 1.00 | 02/01/2018 | Tim | New document |
| 1.01 | 11/01/2018 | Tim | Minor revisions |
| 1.1 | 19/01/2018 | Tim | Revised to reflect increased process automation |
| 1.2 | 05/02/2018 | Tim | Revised again because jobs will not run successfully on V-AWD-SQL-11; also G: drive working folders have changed |

Contents

[DWYT-18 testing cycle 1](#_Toc505587519)

[Revision History 1](#_Toc505587520)

[Introduction 2](#_Toc505587521)

[TritonCelerity SSIS package 2](#_Toc505587522)

[TritonCelerity\_ENews SSIS package 2](#_Toc505587523)

[NeptuneDataExtract SSIS package 3](#_Toc505587524)

[NeptuneDataLoad SSIS package 3](#_Toc505587525)

[NeptuneCRM SSIS package 3](#_Toc505587526)

[HUBSPOT 4](#_Toc505587527)

## Introduction

This document details the steps required to perform a complete test cycle starting with the raw data feed files received from the Triton and Neptune systems through to uploading processed and staged data to the HubSpot API. The steps detailed in the following sections should be completed sequentially in a remote desktop session on server V-AWD-SQL-11. It is assumed that all the associated database revisions have already been deployed, and that the most up to date copies of the SSIS packages are available. The various SSIS packages and associated SQL scripts are run manually for testing purposes but are / will be bundled as MS SQL Server V-AWD-SQL-09 scheduled jobs for running live.

## TritonCelerity SSIS package

Copy the Triton test data feed files TST\_TRT\_\*.csv from folder \\V-JSH-FTP-02\Triton\TritonCelerity to folder G:\Marine\Triton\data\FullLoad. Copy the Triton trigger file TST\_TRT\_yyyymmdd\_TritonCelerity.csv from folder \\V-JSH-FTP-02\Triton to folder G:\Marine\Triton\data\FullLoad. [47 files in total]

Edit the DOS batch file G:\Marine\Triton\Batch\_files\Rename\_files so that the today variable has the correct datestamp value (line 5). Save the revised batch file

Open the TritonCelerity SSIS package in SQL Server Data Tools for Visual Studio 2013 (Connection Managers AMI\_Galaxy configured as V-AWD-SQL-11 and NEPTUNE\_Stage configured as V-AWD-SQL-11)

Run the ‘LOG OPEN - TRITON\_CELERITY’ component

Run the ‘Archive Files’ component

Run the ‘Rename Files’ component

Run the ‘Truncate ‘TRT\_Table\_Row\_Counts’ component

Run the ‘Triton’ sequence container

Run the ‘row counts’ sequence container

Run the ‘LOG CLOSE- TRITON\_CELERITY’ component

In MS SQL Server Management Studio run the following query:

USE CelerityMarine\_Stage

GO

EXEC dbo.TRUNCATE\_CELERITY\_TABLES

EXEC dbo.BUILD\_CELERITY\_TABLES

## TritonCelerity\_ENews SSIS package

Open the TritonCelerity\_ENews SSIS package in SQL Server Data Tools for Visual Studio 2013 and run as is (Connection Managers ‘CelerityMarine\_Stage’ configured as V-AWD-SQL-11 and Live.TRITON\_Stage configured as V-AWD-SQL-09, yes, live)

## NeptuneDataExtract SSIS package

Do not run this

## NeptuneDataLoad SSIS package

Copy the Neptune test data feed files CRMIII\_\*.txt from FTP folder 10.84.244.10 /data/reports/shg\_reports to folder G:\Marine\Neptune\data\FullLoad [41 files in total]

Open the NeptuneDataLoad SSIS package in SQL Server Data Tools for Visual Studio 2013 and run as is (Connection Managers AMI\_Galaxy configured as V-AWD-SQL-11 and NEPTUNE\_Stage configured as V-AWD-SQL-11)

In MS SQL Server Management Studio run the following query:

USE NEPTUNE\_Stage

GO

EXEC dbo.BUILD\_ST\_MARINE\_FBKG

EXEC dbo.BUILD\_ST\_MARINE\_FITN

EXEC dbo.BUILD\_ST\_MARINE\_PAYMENT

EXEC dbo.BUILD\_ST\_MARINE\_PAX

EXEC dbo.BUILD\_ST\_MARINE\_PAXACC

EXEC dbo.BUILD\_ST\_MARINE\_PAXBOAT

EXEC dbo.BUILD\_ST\_MARINE\_PAXFLIGHT

EXEC dbo.BUILD\_ST\_MARINE\_PAXFLIGHTDURATION

EXEC dbo.UPDATE\_ST\_MARINE\_FBKG\_CAA

EXEC dbo.BUILD\_ST\_MARINE\_FLIGHTINVENTORY

EXEC dbo.BUILD\_ST\_MARINE\_BOOKINGITEMS

EXEC dbo.BUILD\_NEPTUNE\_SFT\_ALLOCATION

EXEC dbo.BUILD\_ST\_MARINE\_INSURANCE

EXEC dbo.BUILD\_ST\_MARINE\_BOATAVAILABILITY

EXEC dbo.BUILD\_ST\_MARINE\_AVAILABLEPORTS

EXEC dbo.BUILD\_ST\_MARINE\_SOLDPERWEEK

EXEC dbo.BUILD\_ST\_MARINE\_AVAILPERWEEK

EXEC dbo.BUILD\_ST\_MARINE\_BEDSTOSEATS

GO

USE AMI\_Static

GO

EXEC dbo.UPDATE\_MAP\_FLIGHTINVENTORY\_NEPTUNE

GO

## NeptuneCRM SSIS package

Copy the NeptuneCRM test data feed files CRMIII\_\*.\* and rembook\*.asc.gz from FTP folder 10.84.244.10 /data/reports to folder G:\Marine\NeptuneCRM\data\FullLoad [5 files in total]. Amend the date within the filenames of the Client Preferences and Rembook files to the current date (also check / amend the date within the filename of the zipped file content of the Rembook file (easy to do using 7-Zip file manager))

Open the NeptuneCRM SSIS package in SQL Server Data Tools for Visual Studio 2013 (Connection Managers AMI\_Galaxy configured as V-AWD-SQL-11 and NEPTUNE\_Stage configured as V-AWD-SQL-11)

Run the ‘Open LOG’ component

Run the ‘Load Mail Contact’ sequence container

Run the ‘Load Enquiries’ sequence container

Run the ‘Load Family’ sequence container

Run the ‘Load Client Preferences’ sequence container

Run the ‘Load Rembook’ sequence container

Run the ‘Close LOG’ component

In MS SQL Server Management Studio run the following query:

USE NEPTUNE\_Stage

GO

EXEC dbo.BUILD\_ST\_MARINE\_BOOKINGS

EXEC dbo.BUILD\_ST\_MARINE\_BROCHURE\_REQUEST

EXEC dbo.BUILD\_ST\_MARINE\_CLIENT\_PREFERENCE

EXEC dbo.BUILD\_ST\_MARINE\_CLIENTS

EXEC dbo.BUILD\_ST\_MARINE\_REMBOOK

EXEC dbo.BUILD\_ST\_MARINE\_BOOKING\_LEAD

GO

## HUBSPOT

Change the WHERE clause of the BUILD\_HUBSPOT\_TRITON\_TM\_BOOKINGS, etc. SP selections to select the past 30 days (or other suitable time period) of booking, etc. records. Note that in the current Live interface the selection interval is 8 days.

In a Windows Explorer window for folder G:\Marine\MarineCRM\HUBSPOT, for ZZTEST account data run ‘HUBSPOT\_API - ZZTEST.bat’, for ZLBTEST account data run ‘HUBSPOT\_API - ZLBTEST.bat’, for both accounts run ‘HUBSPOT\_API - DWYT16TEST.bat’.

Note that the API code (Program.cs) may need tweaking to suit changing test account names, etc. Also there are dependencies on table HUBSPOT.dbo. SL\_HUBSPOT\_FIELD\_NAMES (might need new or amended records to suit differences between actual and test accounts in HubSpot) and SP HUBSPOT.dbo. SELECT\_SL\_HUBSPOT\_FIELD\_NAMES (might need tweaking to suit changing test account names, etc.)

Afterwards check the log file ‘HUBSPOT\_API\_yy\_mm\_dd\_hh\_mm\_ss.log’.

Then run ‘ExportRecordsToCSV - ZZTEST.bat’, ‘ExportRecordsToCSV - ZLBTEST.bat’, or ‘ExportRecordsToCSV - DWYT16TEST.bat’ as appropriate. This will generate Errors and Success files, and also export the records processed to CSV files, for detailed checking.