Esteem to BAM Data Export

Author: Hugh Proctor

Initial Draft: 25/09/18

Contents

[Project Background 3](#_Toc525660966)

[System Overview 4](#_Toc525660967)

[Service Workflow 4](#_Toc525660968)

[a. “Process Catch-up” 5](#_Toc525660969)

[b. “Process Errors” 5](#_Toc525660970)

[c. “Start Now” 5](#_Toc525660971)

[d. “Back Process” 5](#_Toc525660972)

[e. “Purge Data” 5](#_Toc525660973)

[System Setup and Service Installation Procedure 7](#_Toc525660974)

[Windows Server Setup Pre-Requisites 7](#_Toc525660975)

[.Net Framework 4.6.1 8](#_Toc525660976)

[Esteem To BAM Reporting Website Setup 8](#_Toc525660977)

[Set IIS Handlers and Modules 8](#_Toc525660978)

[Windows Service Commands 10](#_Toc525660979)

[In Event Viewer 10](#_Toc525660980)

[RegEdit 11](#_Toc525660981)

[Start the Windows Service as Automatic Start 11](#_Toc525660982)

[Windows Service – In the Event of Failed Start 12](#_Toc525660983)

[Appendix 13](#_Toc525660984)

[SCAudit Data Query 13](#_Toc525660985)

[SCDeploy Data Query 13](#_Toc525660986)

# Project Background

Esteem is contracted by BAM Construction to manage the procurement and distribution of hardware – laptops, monitors, tablets, etc.

The logistical movements of these hardware items are audited on the internal computer systems by Esteem, records such as when they were purchased, asset tagged, distributed to a BAM user and returned to Esteem respectively.

BAM in addition has its own computer systems recording the movements of these inventory assets, which, to date has been maintained by an employee on the Esteem side and one or more employees on the BAM side.

**An Automated computer system has been built in order to elevate the workload and hopefully improve data quality.**

This document describes the workings of this system, and how to manage and maintain it for the longevity of its life. Hopefully.

# System Overview

The system consists of 2 main services:

1. **Data Export service** - the core system that directly accesses the Esteem Tesseract data store, cleans, sorts, filters, processes, exports to BAM Api service and reports the successes and failures to the Reporting Website.
2. **Reporting Web site –** a .Net MVC website used for managing product codes and reporting on the Data Export service processing progress.

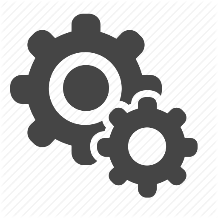
Both services are being hosted on one server – EST-TESSAPP-01 server.

1. **Data Export service – is a Windows Service.**
2. **Reporting Web site – is a Website hosted by IIS.**

**Both services will automatically start when the server is booted and windows is started** – in the event that the serverneeds to be restarted or crashes for whatever reason.

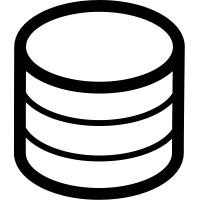
# Service Workflow





**Esteem BAM Export Service**

2

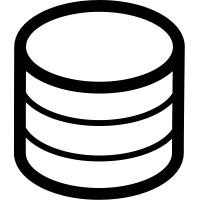


**Tesseract Database (Esteem)**

**Primary Audit Database**

1

**BAMEsteemExportDb (Esteem) Reporting Database**



6



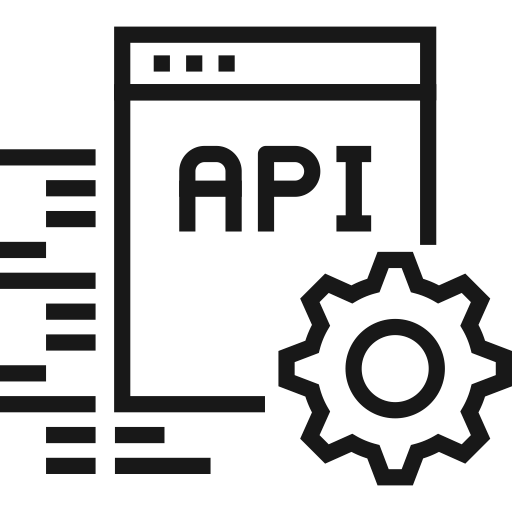
**BAM Esteem Reporting Website**

7

3



4



**BAM Api**

5

1. **Windows Service**[[1]](#footnote-1) – a **scheduled set of jobs** that run hourly:
   1. “Process Catch-up” – this job is intended to Catch-up upon extracts that may have been missed due to the ‘System Down’ scenario – when the service has come offline e.g. maybe the server was restarted or crashed.
   2. “Process Errors” – this job is intended to process any records that may have come from the result of the previous hourly scheduled jobs’ processing.

After, in the occurrence that the service was unable to process one or more records in an hourly scheduled job, the system will record details of the failed instances. The **Asset Unique Serial Number** will be used in conjunction with the **Start and End Time** of job, and the data re-queried, the system assumes and hopes that some human interaction has taken place to update the offending record and thereby enabling its’ successful export to BAM Api.

* 1. “Start Now” – ***this is the primary scheduled job.***

Its purpose is to select any and **all records from the past hour** and process them.

* 1. “Back Process” – **this job runs only once in the life cycle of the system, its sole task is to collect the past two years’ worth of records and process all the data.**

All records that are up-to-date and correct will be bypassed.

All records that are in conflict with the Esteem version will be updated.

* 1. “Purge Data” – this job runs once daily at midnight (or thereabouts). Its task is to clean the Reporting database of surplus reporting information. The scheduled jobs run 24 hours a day, 7 days a week and most of the reporting data is, after 2 months, redundant, this scheduled job will delete any redundant information from the reporting database that is more than 2 months old.

[[2]](#footnote-2) Event Logging is used to report upon errors starting the Windows Service

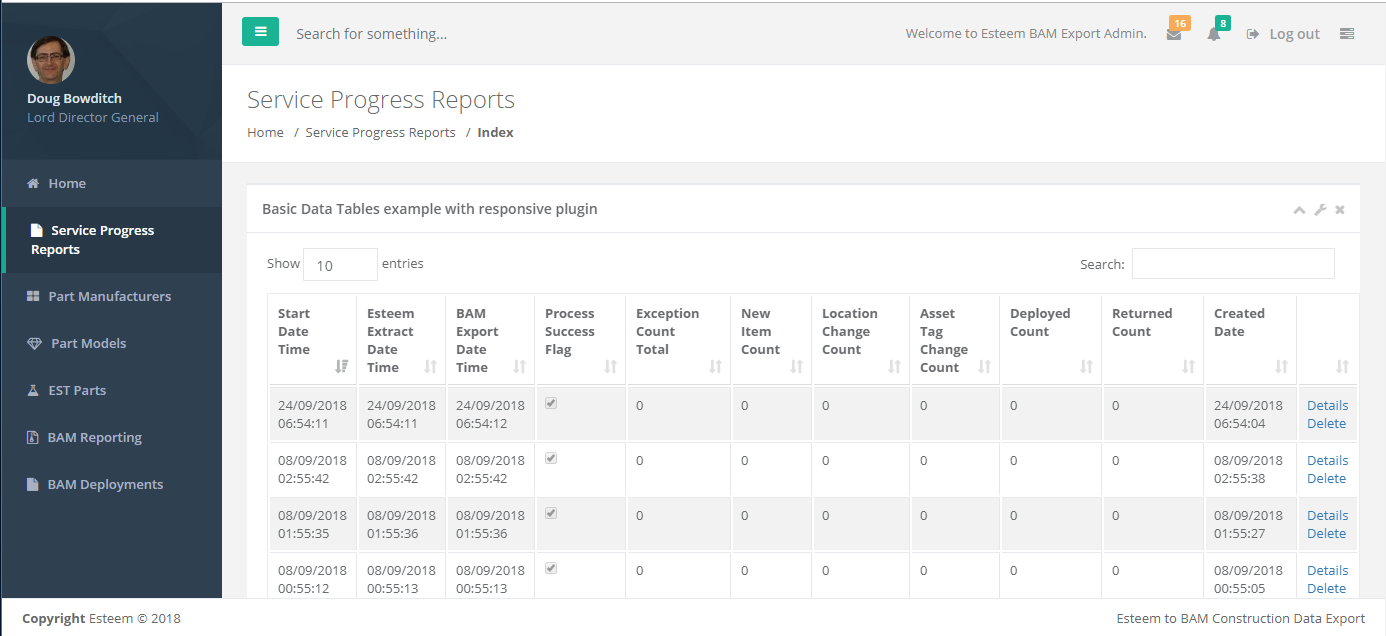
1. **Data is Extracted** from the Esteem Tesseract Database – the Esteem Logistic Audit database[[3]](#footnote-3)
2. **Data is Cleaned** – often one column in the Tesseract database contains two or more pieces of valuable information, this information is extracted and cleaned.

Data requested of the BAM Api is stored in memory, e.g. Manufacturer Type, Device Type, User Contact info; as the data is cleaned, when and where appropriate, we match Esteem data to the respective BAM data.

1. **Data is Filtered** – split into 8 categories:
   1. **New Item** (e.g. Purchase order)
   2. **Location Change** (primarily used to assign the location default of Esteem)
   3. **Asset Tag Change** (when for any reason the Asset Tag has been altered or amended)
   4. **Deployed to BAM User** (location is no longer Esteem, primary user is set)
   5. **Returned from BAM User** – location is Esteem, and User is nullified. (we also stop charging BAM for the Unit)
   6. **Retired Asset**
   7. **Disposed Asset**
   8. **Swapped Asset** – when an asset is swapped one for another to the same user
2. The data is formatted with respect and in correlation to BAM System API and data is transmitted to them.
3. A report is created for each hourly job; time stamps are created throughout the 5 step process previously described, appended to the report and saved into the BAMEsteemExportDb database.

In addition, the Query Start Parameter is recorded, the T-Sql Query string that was used (SCDeploy only), and the Service Job Type ID the log relates too (as described in section 1 of this list)

1. In the website, when one navigates to the Reporting page, this data is displayed and if any errors in processing had occurred the user would be directed to the Exceptions page and the respective exception list displayed.



# System Setup and Service Installation Procedure

## Windows Server Setup Pre-Requisites

In order for the Server to be compatible with the EsteemToBam Service the following steps and features must have been installed. (Assuming a fresh server installation and all data security patches have been applied)

Windows > Programs and Features > Turn Windows features on or off

**> Server Roles**

Add the following:

* Web Server
  + Application Development
    - .Net Extensibility 4.5, Application Initialization, ASP.NET 4.5, ISAPI
  + Security
    - Enable All
  + Health and Diagnostics
    - Enable All
  + Common HTTP Features
    - Enable All

**> Features**

Add the following:

* .Net Framework 4.5 Features > ASP.Net 4.5
* IIS Hostable Web Core
* SMTP Server (Include all the dependent features if and when asked e.g. IIS 6 Management)

Click Next

**> Web Server Role (IIS)**

Add the following:

* Common HTTP Features
  + Default Document
* Management Tools
  + Management Service
* Application Development
  + ASP.NET 4.5 (Include all the dependent features if and when asked e.g. IIS 6 Management)

Click check box – **Restart the destination server automatically if required**

**Click Install…. and wait… ☺ …**

Wait… wait… hmmm... wait? Has it worked? Yay

Restart the server for prosperity

Check for any Windows Updates (none expected)

### .Net Framework 4.6.1

Download .Net Framework 4.6.1 Offline Installer onto your local machine and copy the installation package to the server, install and restart; the site:

<https://www.microsoft.com/en-us/download/details.aspx?id=49982>

## Esteem To BAM Reporting Website Setup

Locate the compiled, published code at:

> repos > EsteemExport > \_Publish > Website

Copy the Website folder to the Server, I’ve used the folder structure

C:\\_EsteemBAMService

\Service

\Website

Open IIS Manager (from the Server Manager screen > IIS

Right-Click the first instance in the list and select **Internet Information Services (IIS) Manager**

From IIS Manager

Right-Click the Server name (EST-TESSAPP-01) > Add Website

Fill in form:-

Site name: EsteemToBAMExport

Physical Path: > Click … and choose C:\\_EsteemBAMService\Website

Double click on the new Site > EsteemToBAMExport

In the middle pane, Double-Click **Authentication**

Set **Windows Authentication** to **Enabled**

### Set IIS Handlers and Modules

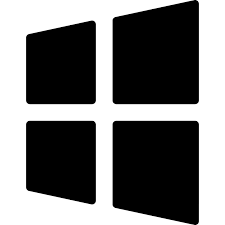
In IIS, click the server e.g. EST-TESSAPP-01

1. Open IIS Manager
2. Click the server name in the tree on the left
3. Right hand pane, Management section, double click Configuration Editor
4. At the top, choose the section system.webServer/security/authentication/anonymousAuthentication
5. Right hand pane, click Unlock Section
6. At the top, choose the section system.webServer/security/authentication/windowsAuthentication
7. Right hand pane, click Unlock Section

## Windows Service Commands

The following code and commands are used within Windows Command console (CMD), their purpose is to enable you to:

1. Install and Uninstall a Windows Service
2. Start a Windows Service Manually (from command line, but can also be done from the Services within Windows Control Panel.

Open a command console by clicking Windows  and typing - cmd . Right-Click and **Run as Administrator**

In command console type:

**cd C:\Windows\Microsoft.NET\Framework\v4.0.30319**

To Install type (all in one line – copy and paste):

**installutil /name="\_Esteem.ExportToBAM" /displayname="\_Esteem.ExportToBAM" /description="\_Esteem.ExportToBAM Windows Service" "C:\\_EsteemBAMService\Service\Esteem.ServiceHost.exe"**

To Uninstall:

**installutil /u /name="\_Esteem.ExportToBAM" "C:\\_EsteemToBAMExport\Esteem.ServiceHost.exe"**

To Delete a Windows Service manually:

**sc delete \_Esteem.ExportToBAM**

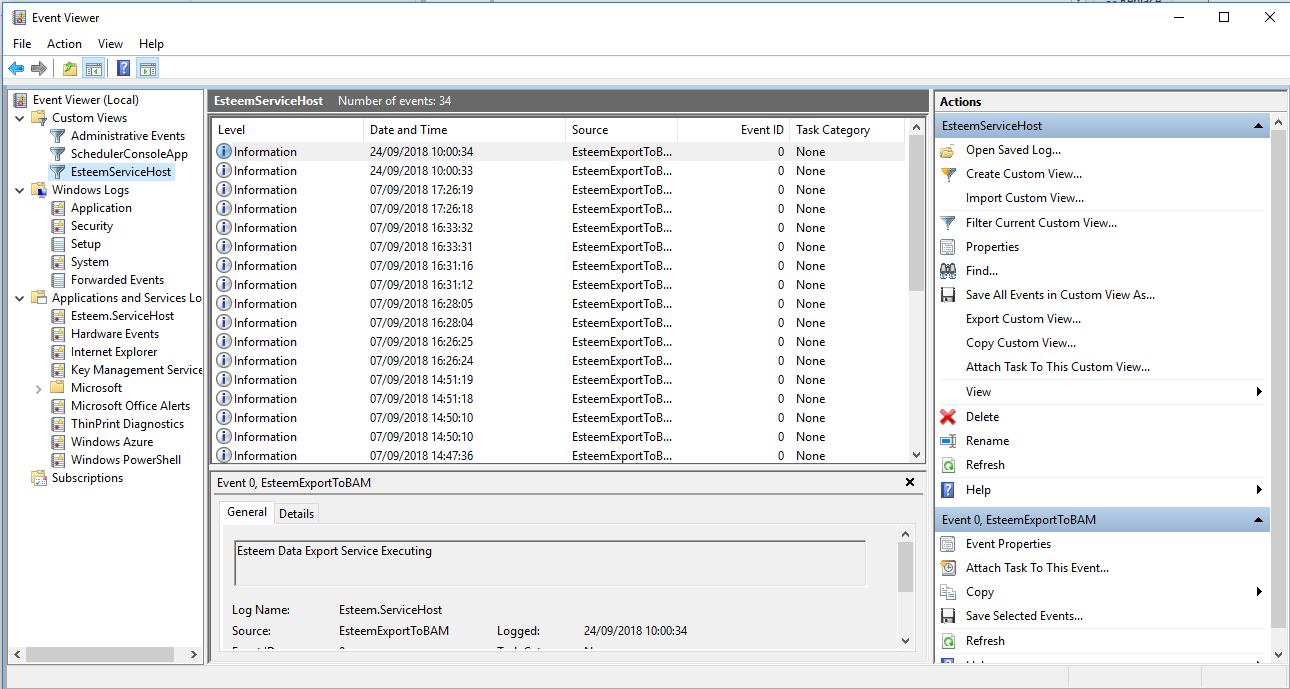
To Start the Windows Service:

**net start \_Esteem.ExportToBAM**

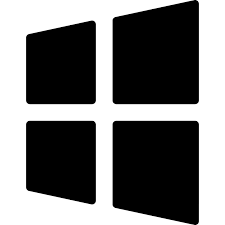
## In Event Viewer

In the Event Log, under > “Applications and Services Log” you will find “Esteem.ServiceHost”

This is the unique registered key for the service event logging. If the service fails to start for whatever reason then this should give you an indication of the source error.

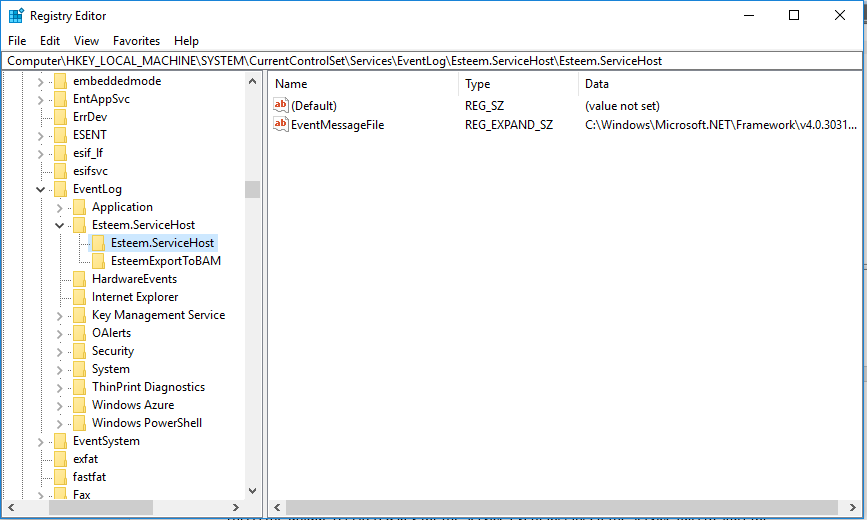
Right-Click on the entry in Applications and Services Logs to **Create Custom View…**

## RegEdit

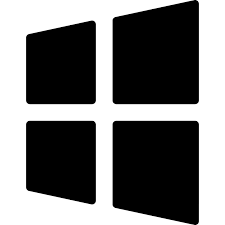
A registry key is created for the windows service and this allows for the use of the Event Log entry

Navigate to the Registry by clicking Windows and typing regedit .

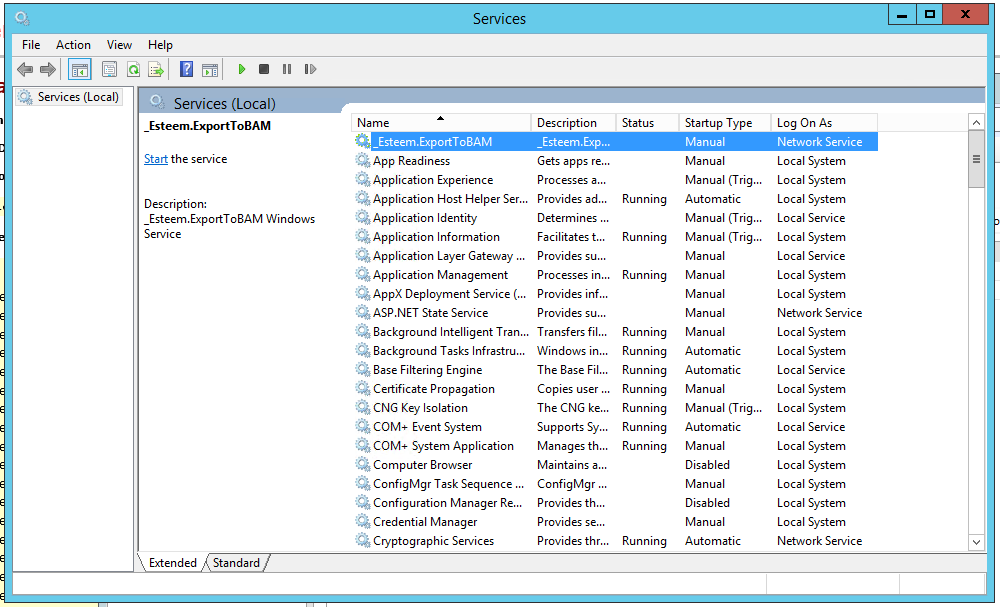
Locate: Computer\HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Esteem.ServiceHost\Esteem.ServiceHost



### Start the Windows Service as Automatic Start

Open a command console by clicking Windows  and typing - services

At the top should be **\_Esteem.ExportToBAM**

****

Right-Click the **\_Esteem.ExportToBAM** entry > Click Properties

In the **Startup type:** dropdown set it to **Automatic**

Click **Apply**

Click **Start**

### Windows Service – In the Event of Failed Start

Errors are logged in 3 places:

1. Event Viewer
2. Exceptions Folder in: C:\\_EsteemBAMService\Service\Exception
3. The BAMEsteemExportDb

# Appendix

## SCAudit Data Query

SELECT TOP 10000000

NULLIF(LTRIM(RTRIM(SCA.[Audit\_Ser\_Num])), '') AS [SERIAL\_NO],

NULLIF(LTRIM(RTRIM(SCA.[Audit\_Part\_Num])), '') AS [PART\_NO],

NULLIF(LTRIM(RTRIM(SCP.[Part\_Desc])), '') AS [PART\_DESC],

NULLIF(LTRIM(RTRIM(SCP.[Part\_Alt\_Part\_Num])), '') AS [PART\_DESC\_ALT],

NULLIF(LTRIM(RTRIM(SCA.[Audit\_Source\_Site\_Num])), '') AS [SOURCE\_SITE\_NO],

NULLIF(LTRIM(RTRIM(SCA.[Audit\_Dest\_Site\_Num])), '') AS [DESTINATION\_SITE\_NO],

NULLIF(LTRIM(RTRIM(SCP.[Part\_Type])), '') AS [PART\_TYPE],

NULLIF(LTRIM(RTRIM(SCA.[Audit\_Rem])), '') AS [REMARK],

NULLIF(LTRIM(RTRIM(SCA.[Audit\_User])), '') AS [USER],

SCA.[Audit\_Move\_Date] AS [MOVE\_DATE],

SCA.[Audit\_Last\_Update] AS [UPDATE\_DATE]

FROM [SCAudit] AS SCA (NOLOCK)

FULL JOIN [SCPart] AS SCP ON SCA.[Audit\_Part\_Num] = SCP.[Part\_Num]

WHERE [Audit\_Last\_Update] >= CONVERT(DATETIME, '01/11/2017 00:00:00', 103) AND [Audit\_Last\_Update] <= CONVERT(DATETIME, '01/12/2017 00:00:00', 103)

and [Audit\_Part\_Num] like 'BNL%'

--and [Audit\_Rem] like 'Added PO%'

and [Audit\_Source\_Site\_Num] = 'BNLSCRAP'

and [Audit\_Dest\_Site\_Num] is null

and [Part\_Type] = 'R'

ORDER BY [Audit\_Source\_Site\_Num] desc

## SCDeploy Data Query

SELECT TOP 10000000

NULLIF(LTRIM(RTRIM(SL.[Fsrl\_Id\_Num])), '') AS [SERIAL\_NO],

NULLIF(LTRIM(RTRIM(SL.[Fsrl\_Ret\_Id])), '') AS [SERIAL\_NO\_RETURNED],

NULLIF(LTRIM(RTRIM(SL.[Fsrl\_Part\_Num])), '') AS [PART\_NO],

NULLIF(NULLIF(LTRIM(RTRIM(SL.[Fsrl\_Ret\_Part\_Num])), ''), 'NONE') AS [PART\_NO\_RETURNED],

NULLIF(LTRIM(RTRIM(SL.[Fsrl\_Part\_Desc])), '') AS [PART\_DESC],

NULLIF(NULLIF(LTRIM(RTRIM(SCP.[Part\_Desc])), ''),

'PART USED NONE RETURNED') AS [PART\_DESC\_RETURNED],

NULLIF(LTRIM(RTRIM(SCP.[Part\_Type])), '') AS [Part\_Type],

NULLIF(LTRIM(RTRIM(SCP.[Part\_Alt\_Part\_Num])), '') AS [Part\_Code],

NULLIF(LTRIM(RTRIM(SL.[Fsrl\_Call\_Num])), '') AS [Call\_Num],

NULLIF(LTRIM(RTRIM(SC.[Call\_Contact])), '') AS [Call\_Contact],

--NULLIF(LTRIM(RTRIM(SC.[Call\_Num])), '') AS [Call\_Num],

NULLIF(LTRIM(RTRIM(SC.[Call\_Ref])), '') AS [Call\_Ref],

CASE

WHEN LEFT([Call\_Phone], 2) LIKE '0%' THEN NULL

ELSE LTRIM(RTRIM([Call\_Phone]))

END AS [Cost\_Code],

SL.[Fsrl\_Last\_Update] AS [UPDATE\_DATE]

FROM [SCFsrl] AS SL (NOLOCK)

FULL JOIN [SCPart] AS SCP ON SL.[Fsrl\_Part\_Num] = SCP.[Part\_Num]

FULL JOIN [SCCall] AS SC ON SL.[Fsrl\_Call\_Num] = SC.[Call\_Num]

FULL JOIN [SCCalt] AS SCT ON SC.[Call\_Calt\_Code] = SCT.[Calt\_Code]

WHERE [Fsrl\_Last\_Update] >= CONVERT(DATETIME, '01/11/2017 00:00:00', 103)

AND [Fsrl\_Last\_Update] <= CONVERT(DATETIME, '01/12/2017 00:00:00', 103)

--AND ISNULL(Fsrl\_Ret\_ID, '') <> ''

AND [Part\_Type] != 'D'

--and [Fsrl\_Id\_Num] like 'BAM%'

and [Fsrl\_Part\_Num] like 'BNL%'

ORDER BY [UPDATE\_DATE] desc

1. Example Windows Service – create, destroy, start and stop command (CMD) code is supplied in Appendix [↑](#footnote-ref-1)
2. Example of how to access the Event Viewer and Registry Key can be found in the Appendix [↑](#footnote-ref-2)
3. Example database query is included in the Appendix for the 2 base queries used [↑](#footnote-ref-3)