



# SDL Campaigns 2016 R1 Upgrade Guide

# Version Management

## Document history

Version	Date	Author	Changes	Distribution
1.0	08 January 2016	L Watts & Pete Trevitt		Release
1.1	28 April 2016	J Davis & Pete Trevitt	For CM 5.1.2 service release - info added to Section 2.4 regarding Engine shutdown.	Release

## Associated Documents

Name	Location
SDL Campaigns 2016 R1 Release Notes	SDL Documentation Centre
SDL Campaigns 2016 R1 Architecture Guide	SDL Documentation Centre
SDL Campaigns 2016 R1 Load Process Guide	SDL Documentation Centre
SDL Campaigns 2016 R1 Data Flow and Structure	SDL Documentation Centre
iLoader Best Practice - Strategies for Loading Data	SDL Documentation Centre

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>1</b>
<b>2</b>	<b>Important Upgrade Considerations when upgrading from Campaign Manager 4.0.x .....</b>	<b>2</b>
2.1	Pre-Requisite software and supported platforms .....	2
2.1.1	Notification of no longer supported configurations.....	2
2.1.2	Campaign Manager 4.0 required as a pre-requisite if upgrading from 3.0.x .....	3
2.1.3	Email Manager 6.0.x .....	3
2.1.4	Deprecation of browsers .....	3
2.2	Campaign History DB/Tables names for migration.....	3
2.3	Hardware Profile Changes.....	4
2.3.1	Disk Usage.....	4
2.3.2	CPU/RAM on the Engine Server.....	6
2.4	System State pre-requisite for Upgrade.....	6
2.5	Time required for the upgrade.....	7
2.6	Load Process changes .....	8
2.7	Campaign Roll-Off .....	8
2.8	Data Engineering considerations for Campaign History tables .....	8
2.9	No support for Shared Repositories.....	9
2.10	Engine removal of API Support.....	9
2.11	Event Storage transition from MS SQL to Engine .....	10
2.12	Change to Engine install folder and process names .....	10
2.13	Engine Load Best Practice Review .....	11
<b>3</b>	<b>Upgrading to SDL Campaigns 2016 R1 .....</b>	<b>12</b>
3.1	Backup.....	12
3.2	Pre Upgrade Configuration .....	13
3.3	Installing Kettle .....	13
3.4	Run Upgrade .....	13
3.5	Upgrading from SDL Campaign Manager 4.0.x to SDL Campaigns 2016 R1 .....	18
3.5.1	Campaign History Migration Utility.....	19
3.6	Campaign Activity Data Migration Utility for SDL Campaigns 2016 R1 .....	23

3.7	Post-Migration Step .....	24
3.8	Post Upgrade Configuration.....	24
3.8.1	SQL Server .....	24
3.8.2	Engine .....	25
<b>4</b>	<b>Uninstalling SDL Campaigns .....</b>	<b>26</b>
<b>5</b>	<b>Troubleshooting .....</b>	<b>27</b>
5.1	CM Upgrade troubleshooting .....	27
5.2	Engine Upgrade troubleshooting .....	27
5.3	Engine Installation troubleshooting .....	27
<b>6</b>	<b>Campaign Manager Validation .....</b>	<b>28</b>
6.1	Fail: Upgrade from CM v999 to v999 is not valid .....	28
6.2	Fail: OS is Invalid .....	28
6.3	.NET Framework 4.5.1 required on App, Engine and SQL Server/s .....	28
6.4	Current User is not a member of the Built-in Administrators Group .....	29
6.5	IIS6 WMI Management Compatibility must be installed on the App Server .....	29
6.6	Invalid Kettle Zip File.....	29
6.7	Fail: Server is not accessible .....	29
6.8	Fail: User Credentials are invalid .....	29
6.9	Fail: Framework version is invalid .....	29
6.10	Fail: SQL is not valid .....	29
6.11	Fail: IIS is invalid .....	29
6.12	Fail: Email Manager Account Link Name must be specified .....	29
6.13	Fail: Email Manager unable to validate URL .....	30
6.14	Fail: Alchemy Service Credentials are invalid (or) Fail: Service Credentials failed DB Connection.....	30
6.15	Fail: AltEngineCampaign.DLL locked; Fail: EMTools.DLL locked; Fail: APIClientDLL locked; Fail: EngineMutex.DLL locked .....	30
6.16	Fail: Inaccessible xxxx Component (ID n) .....	30
6.17	Fail: Install Folder cannot be created and shared .....	31
6.18	HTTP Error 500.0 - Internal Server Error Calling LoadLibraryEx on ISAPI filter "C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_filter.dll" failed.....	31
6.19	Fail: Engine component 'nn' is active. Wait or shutdown .....	31
6.20	Fail: Engine already installed .....	31
6.21	Fail – SQL Filestream: full access not configured .....	32

6.22	Fail – Engine v6 or higher not installed .....	32
6.23	Fail: Engine NucBroker service Credentials are invalid .....	32
6.24	Fail: Engine ProjectBase path exists.....	32
6.25	Fail: Engine Install Path not accessible; Fail: Engine ProjectBase Path not accessible.....	32
<b>7</b>	<b>Appendixes .....</b>	<b>33</b>
7.1	Appendix A – Adding Email Manager URLs .....	33

# 1 Introduction

This document provides an overview of considerations that must be given to the existing environment prior to upgrade to SDL Campaigns 2016 R1 along with step-by-step guidance for the upgrade.

For an overview of common Campaign Manager Architecture configurations and guidelines for planning a system, please refer to the document *SDL Campaigns 2016 R1 Architecture Guide.pdf*.

If you require further assistance, please contact your SDL account representative.

The following sections are included in this document:

- Upgrades
- Uninstalls
- Troubleshooting

## 2 Important Upgrade Considerations when upgrading from Campaign Manager 4.0.x

The following section outlines areas of consideration for upgrades of existing **Campaign Manager 4.0.x** systems to SDL Campaigns 2016 R1. These include changes to architecture as well as the upgrade process and should be reviewed in depth as part of upgrade planning.

If you are upgrading from **Campaign Manager 2015 R1 (CM 5.0)**, you can skip this section and go directly to section 3 - [Upgrading to SDL Campaigns 2016 R1](#).

---

### IMPORTANT NOTE

SDL Campaigns 2016 R1 has significant enhancements that allow campaigns and their actionable data to be available faster than ever before.

These have meant some significant architectural changes to all products that form part of the SDL Campaigns suite as well as changes to the location and storage of some of the Campaign Manager History Data and Campaign Manager State Data.

This will impact hardware profile, existing processes and user experience of the application.

For this release, it is vital that all these aspects are considered as part of upgrade planning, and failure to do so may well leave a system under specified and therefore with performance issues that won't be resolved by software changes.

---

### 2.1 Pre-Requisite software and supported platforms

#### 2.1.1 Notification of no longer supported configurations

SDL Campaigns 2016 R1 cannot be used in parallel with AMS on a shared Engine repository, as Engine is no longer compatible with AMS. AMS, using Engine 5.3, will benefit from an extended support at the 'Mature' product phase of the SDL lifecycle. We also provide a commitment to continue this level of Support to 1-Jan-2020 to ensure you have plenty of time to consider and plan an upgrade to our new platform, SDL Campaigns.

## 2.1.2 Campaign Manager 4.0 required as a pre-requisite if upgrading from 3.0.x

SDL Campaigns 2015 R1 introduced some significant and exciting new functionality around the consolidation and near real-time processing of event data received from Campaigns executed in both Email Manager and SDL Campaign Manager.

To facilitate the automatic migration of campaign and deployment metadata especially when using a system integrated with Email Manager, it is vital to have a period running CM 4.0.x prior to upgrading to 2016 R1. This is due to collection of Email Manager events.

Any event generated by a campaign in Campaign Manager 3.0, i.e. opens, clicks etc. will not be collected by the new version of Campaign Manager, so the period of running Campaign Manager 4.0.x must be long enough to ensure all the required events from previous version deployments have been collected.

As you can upgrade directly from Campaign Manager 4.0.x to SDL Campaigns 2016 R1 without the need to deploy SDL Campaigns 2015 R1, access to SDL Campaigns 2015 R1 has been removed.

## 2.1.3 Email Manager 6.0.x

Campaign Manager is only supported alongside Email Manager 6.0.x to support the new Contact and Response processing for Send a Message tactics. Email Manager is an SDL hosted application and SDL Cloud Services team will publish the Email Manager release schedule for each region to existing Customers.

## 2.1.4 Deprecation of browsers

SDL Campaigns utilizes a number of different web technologies including Microsoft Silverlight.

In September 2015 Chrome removed Silverlight support. This means that the Campaign Manager module is not available in Chrome. All other modules are available.

SDL Campaigns 2016 R1 is fully supported with Microsoft Internet Explorer 11 browser.

SDL Campaigns is also no longer tested or supported on Firefox or Safari.

## 2.2 Campaign History DB/Tables names for migration

As part of the enhancements in this release, a new Campaign History Data Schema is created in Engine to be used by the application going forward. This is built by the migration process, using existing History tables as the source data.

**Prior to migration, the following information must be gathered:**



- **Source History Database** – The database name that the existing Campaign History resides in.
- **Existing History Table (s)** – The current name for the Campaign History table as created by the overnight ArchiveTables process e.g. Demo\_Customer\_Cust ID.
- **Existing Events Table (s)** – The current name for the Events table as created in iLoader by the AppendEvents process.

Prior to migration, the following objects must be known, as these values need to be input during the migration stage:

- **Destination History Database** – The database name for the new Campaign History Data. This can be a new or existing database.
- **Master Table** – The name for the new Master Campaign Table for the defined Datasource. Defaults to Master Campaign.
- **New Contacts Table** – The name given to the Contacts table in the new Campaign History Data structure. Defaults to Contacts.
- **New Response Table** - The name given to the Responses table in the new Campaign History Data structure. Defaults to Responses.

---

**IMPORTANT  
NOTE**

Consideration should be given to how the new three table Campaign History Data Schema will fit into existing processes. The existing tables will persist for reporting/queries etc. but only from a historical perspective, they will no longer have data added to them.

New Campaign History Data Schema comes with the "out of the box" dashboards but any bespoke reporting will need to be considered.

---

## 2.3 Hardware Profile Changes

### 2.3.1 Disk Usage

This upgrade introduces new functionality in Engine allowing data to be intra-day loaded in bulk to the Campaign History Data Schema tables, using the "mirrored" table functionality in Engine, along with a new standard schema for the Campaign Manager History Data. For more info on this process, refer to the *SDL Campaigns 2016 R1 Data Flow and Structure Guide.pdf*

A migration utility, running as part of the software upgrade process, will create three new tables for the Campaign History Data Schema, derived from, but not editing, the existing two Campaign History tables for each campaign key.

For each environment being upgraded, the utility will cycle through all Client databases and for each Client Database, will cycle through each Datasource. In each Datasource, for each Campaign key, the administrator will be required to define the previously gathered information.

This process will increase disk usage by creating new tables based on existing Campaign History data, but this stage is mandatory to ensure campaign activity re-starts using the new tables.

The new tables are optimized for storage but will obviously introduce a rise in disk usage.

The table below gives estimates based on SDL Testing.

The first column shows the disk size of the original two table history. The second column shows the estimated size of the new three table history including the “mirror” copies.

DISK SIZE OF ORIGINAL HISTORY DB (Gb)	ESTIMATED SIZE OF NEW HISTORY DB (Gb) *does <b>not</b> include original data.
5.00	8.00
10.00	17.00
15.00	22.50
20.00	31.5

In summary, the new history is approximately 1.5 to 1.7 times the size of the existing history.

The existing two Campaign History tables can be manually deleted if not required after the upgrade, saving considerable space but this must be done in agreement with all stakeholders.

To minimize disk usage impact, it is recommended that prior to upgrade, the existing two Campaign History tables contain only required data, reducing the data to be migrated.

**Prior to upgrade, Administrators should:**

- Roll-Off of any campaigns that are no longer required to reduce the size of the tables and avoid migrating unnecessary data. Campaign roll-off in Campaign Manager 4.0.x only removed Contacts data, so Event data must be deleted manually from AMC.

- Delete Variables or Event Parameters no longer in use to avoid migrating unused columns. These will all be migrated if left.
- Review Indexing – Indexing on the new tables is already optimized for reporting, whereas on the existing two tables, all columns are indexed. Administrators can likely reduce the index footprint on the existing tables by indexing only those columns required for onwards querying.

### 2.3.2 CPU/RAM on the Engine Server

Daytime processor activity will increase with the intra-day reporting processing. This will require the Engine server to perform more processing during daytime activity. RAM and CPU should be carefully monitored. This is not as easy to estimate as disk space but if the current system is monitored prior to upgrade and found to be operating at high levels of CPU and Memory usage, pre upgrade improvement is advised. See the *SDL Campaigns 2016 R1 Architecture Guide.pdf* for high-level suggestions on specifications.

SDL guidance is that 64GB RAM is likely to be required on a production Engine Server.

## 2.4 System State pre-requisite for Upgrade

The state of the system when the upgrade occurs is of paramount importance. The Deployer contains an embedded executable that creates the new Campaign History Data Schema from the existing Contact and Response tables, as well as importing the Dashboards for immediate viewing after upgrade. This process needs the system to be in a certain state to perform correctly.

**The upgrade must occur with the system in a specific state.**

- 1) The disk space and CPU/Ram considerations have been reviewed and acted upon accordingly.
- 2) All Campaign activity paused and all users are off the system
- 3) All Datasources must be disabled.
- 4) ArchiveTables command must be executed to ensure there are no [\_TacticOutputTables] waiting to be archived and that the existing Contacts table is up to date.
- 5) SQL Event BCP export and subsequent AppendEvents commands must be run to ensure all Events are extracted from SQL and added to the existing Responses table ready for migration. The deployer will turn off the BCP export during upgrade.
- 6) All unwanted campaigns have been rolled off (to minimize the time taken to run the data creation utility and avoid filling the new tables with unwanted data).

- 7) Indexing reviewed on the existing tables to drop indexes on columns not used for querying or reporting to reduce disk space of these legacy tables.
- 8) All unwanted Variables or Event Parameters deleted from the system to minimize migration times but more significantly, to reduce disk storage.
- 9) Any engineering column on the existing Campaign History tables have been dropped to ensure a clean migration.
- 10) Any Attribution processes have been run.
- 11) Engine has performed a shutdown. Depending on campaign activity, this may take some time. If there is current activity, the shutdown will fail. You can retry the shutdown at any time, however under **no** circumstances should you use Windows Task Manager to stop any Campaign Manager or Engine processes.

## 2.5 Time required for the upgrade

The data migration function in the Deployer will take time to create the new Campaign History Data. The table below gives an indication of time to create the new three table Campaign History Data. The information is based on the overall totals of both History tables.

NUMBER OF ORIGINAL COLUMNS (both History tables)	ESTIMATED TIME FOR MIGRATION (seconds per 1 million rows)
0 - 50	5 s
50 – 100	5-10 s
100-150	10-15 s
150 +	15+ s

## 2.6 Load Process changes

The end-to-end Campaign Manager load process will need to be edited to remove the ArchiveTables and AppendEvents processes. These stages are no longer required, likely reducing Load Process times. The *SDL Campaigns 2016 R1Load Process.pdf* document will describe the new process stages required.

## 2.7 Campaign Roll-Off

It should be noted that in Campaign Manager 4.0.0, Campaign roll-off acted only upon the Contacts table, but will now roll-off from Contacts and Responses so this process is likely to take considerably longer than with the previous version but this enhancement is vital for disk space management going forward.

## 2.8 Data Engineering considerations for Campaign History tables

Any Data engineering required on the new Campaign History tables after upgrade will now be receiving data intra-day, and special consideration must be given to all Data Engineering on these tables.

---

**NOTE** All existing engineering on existing history tables must be dropped prior to upgrade. This section is about the approach to re-scripting and re-creating that engineering.

---

Some level of user training will be required to better understand the concept of Data Engineering on intra-day data tables. As the data in these tables is now changing, any derived or engineering data based on or created on these tables, will quickly go stale. There will now be methods to create either snapshot data, i.e. column data where the counts will remain

static after a data change, new data being added as null, or data that will remain dynamic and will be dropped after a data change, relying on the application to re-build on demand.

For more information, please refer to the *Best Practice Guide - Data Engineering in Campaign Manager and Engine.pdf* and *iLoader Best Practice - Strategies for Loading Data.pdf*

## 2.9 No support for Shared Repositories

The use of shared repositories is not supported with this system. This option must be turned off if the system is currently using it, or you are planning to in the future - this configuration is not supported.

## 2.10 Engine removal of API Support

Historically, SDL and previously Alterian, supported direct access to the Engine API so that partners could write custom plug-ins that fulfilled niche requirements (although this was not explicitly stated in contracts).

As maintaining code compatibility with multiple custom applications is unworkable, and the majority of functional requirements are now fulfilled via the core product, access to the API from Engine is no longer supported.

Engine, and the improvements outlined in this document, are only possible because Engine development and QA is now focused exclusively on the requirements of SDL Campaigns, to the point where Engine is no longer available as a separate product.

Engine API usage may include the following types of customization:

- Custom GUI applications that plug directly into the Engine API
- Injected DLLs including bespoke iLoader plugins
- VB Scripts that call in Engine, via iLoader or AMC Scheduler
- Any Pentaho Kettle plugin that does not use the supported SDL iLoader Connector

Engine API support will continue with Engine 5.3, supported for use with AMS until 2020, and with Engine 6.0 on existing Campaign Manager systems.

Note that the Campaign Manager API will now be the sole source API access point for the application.

For more information contact Dave Hitchins: [DHitchins@sdl.com](mailto:DHitchins@sdl.com)

## 2.11 Event Storage transition from MS SQL to Engine

In previous versions of Campaign Manager, response events of all types, were stored in the MS SQL server initially then, during overnight processing, were exported to a file, transferred to the Engine server, loaded to staging tables and finally appended to the existing Responses table.

As well as adding to the overnight load process and the downtime incurred, this also required significant disk space and processing on the SQL server. This will no longer occur so after the upgrade, no events will be written to SQL.

## 2.12 Change to Engine install folder and process names

When the SDL Campaigns deployer runs, as part of an upgrade, Engine versions will be upgraded on all the required servers as declared in the multi-server system.

No subsequent manual configuration will be required.

---

<b>NOTE</b>	An installation of Engine is still required on the application server, and will also be upgraded as part of the deployer
-------------	--

---

For on-going support it is vital that these changes are understood.

### Installation Locations

Engine installs in a different location to previous versions of Engine. During the upgrade, the Deployer process will un-register all previous versions of Engine. (These will no longer work after the upgrade, as all configurations will point to the new locations.)

The new location will be as follows:

- Engine program files will be installed in a “Program Files\SDL Engine” folder on the same drive as existing Engine 6.x program files. For example, Engine is installed in D:\Application\Alterian, it will now be D:\Program Files\SDL Engine.
- The Configuration files and Logging for Engine are re-located to the “ProgramData\SDL Engine” folder on the same drive as existing Engine v6.x program files. As part of the upgrade process, the following Engine 6.x configuration is copied:
  - Engine license key
  - Connectionbroker.config projects configuration
  - AMC.xml file

---

**NOTE**

- The ProgramData folder is usually hidden by the operating system, so you may need to “show hidden files and folders” to view this.
  - Any AMC Scheduler services must be manually re-created as a post-upgrade step. The relevant AMCSchedulerWindowsService.xml configuration files are retained in the Engine 6.x Assembly Folder. AMC Schedulers can be created in Engine using the AMCSchedulerWindowsService.exe command located in the Program Files AMC folder.
- 

- Engine registry locations have been updated to HKLM\Software\SDL\_Engine for 64 bit components, and HKLM\Software\Wow6432Node\SDL\_Engine for 32 bit components.
- All application data, i.e. ProjectBase folders, Database Repositories etc. will remain in the old locations to avoid any changes to backup process etc.

### Executable and Service Name Changes

Some Service and Executable names have changed for the new version, namely the following:

- ConnectionBroker will now be called NucBroker
  - Nucleus.exe is now called NucEngine.exe
- 

**NOTE**

- As described above, the NucBroker service is installed as part of Engine, and replaces the ConnectionBroker service in previous versions. The deployer wizard will prompt for the Engine NucBroker service account details during the upgrade path. It is recommended the common user account is used for the NucBroker service. This account must be a local admin with the “log on as a service” on App and Engine servers. It is not necessary for it to be a domain admin. It is recommended to use the account previously configured to run ConnectionBroker on the existing system.
- 

## 2.13 Engine Load Best Practice Review

To facilitate the intra-day updates on the Campaign History Data Schema tables and to allow better support for intra-day updates to Customer Data Warehouse tables in future versions, Engine must be more aware of data changes that occur and provide options on managing



them. To ensure changes are managed, specifically in relation to objects that are based on changing data, Engine has a more robust dependency model. Consequently, objects such as aggregates and expressions are aware when data they are based on has changed and will change their state accordingly.

This will have impact on existing iLoader scripts, application usage and Customer Data Warehouse loads and all scripts should be reviewed, edited and UAT tested.

For more information, refer to the iLoader Best Practice - Strategies for Loading Data.pdf

## 3 Upgrading to SDL Campaigns 2016 R1

The SDL Campaigns Installer can be used to upgrade existing systems, provided those systems were originally installed using the Installer or Single Sign On.

Please note if you are not currently using single sign on, and wish to, we recommend you perform the upgrade and then contact support or refer to the *SDL Campaigns 2016 R1 Administration Guide* for information on how to add single sign on. If you already use NT Authentication you will need to change the value of the following held in the "Alterian\Alchemy\Alterian.Alchemy.HostingSite\web.configfolder":

```
<add key="authService" value="https://<USERDOMAIN>/auth/AlterianAuthentication.svc/singlesignon"/>
```

To:

```
<add key="authService" value="https://<USER DOMAIN>/auth/AlterianAuthentication.svc/httpscert"/>
```

### 3.1 Backup

SDL recommends that the following are backed up prior to beginning the upgrade:

- Campaign Manager websites
- SQL Server databases:
  - All client databases
  - AlchemyStore
  - AIMain
  - AlterianAuth
- Engine Cerlog – The upgrade process will overwrite this file due to a change in cerlog format to UTF-8 for better support logging of Unicode Engine data.

## 3.2 Pre Upgrade Configuration

Save the Installer package to the App Server. Ensure the required .NET 4.5.1 is installed. Note if this is not installed the Deployer will display an error and will not be permitted to proceed.

## 3.3 Installing Kettle

Kettle and Java Runtime Environment requires a manual install if you are upgrading, and have not previously installed Kettle. For information on how to manually install Kettle, and how to configure Kettle for use with SDL Campaigns see the ***Configuring Pentaho Kettle for use with SDL Campaigns*** guide.

## 3.4 Run Upgrade

1. Run the CM.Deployer.exe file using the 'Run as administrator' option to launch the Installer wizard.
2. Click **Next** on the Welcome screen.
3. Select **Upgrade Existing** as the installation type.
4. Type in the SQL Server name and click the **Test Connection** button to confirm that it is valid.

SDL Campaign Manager Platform Installer Upgrade

**Attach Host Server**

Enter the server name and instance which performs the role of Campaign Manager SQL Server Host.

**→ Attach Host Server**

- Review Servers
- Modify Credentials
- Manage Configuration
- Orderly Shutdown
- Engine Upgrade
- Validate Settings
- Perform Upgrade

**SQL Server Configuration**

SQL Server Name:

SQL Instance Name:

**SQL Server Authentication**

☒ Use Windows Authentication Test Connection

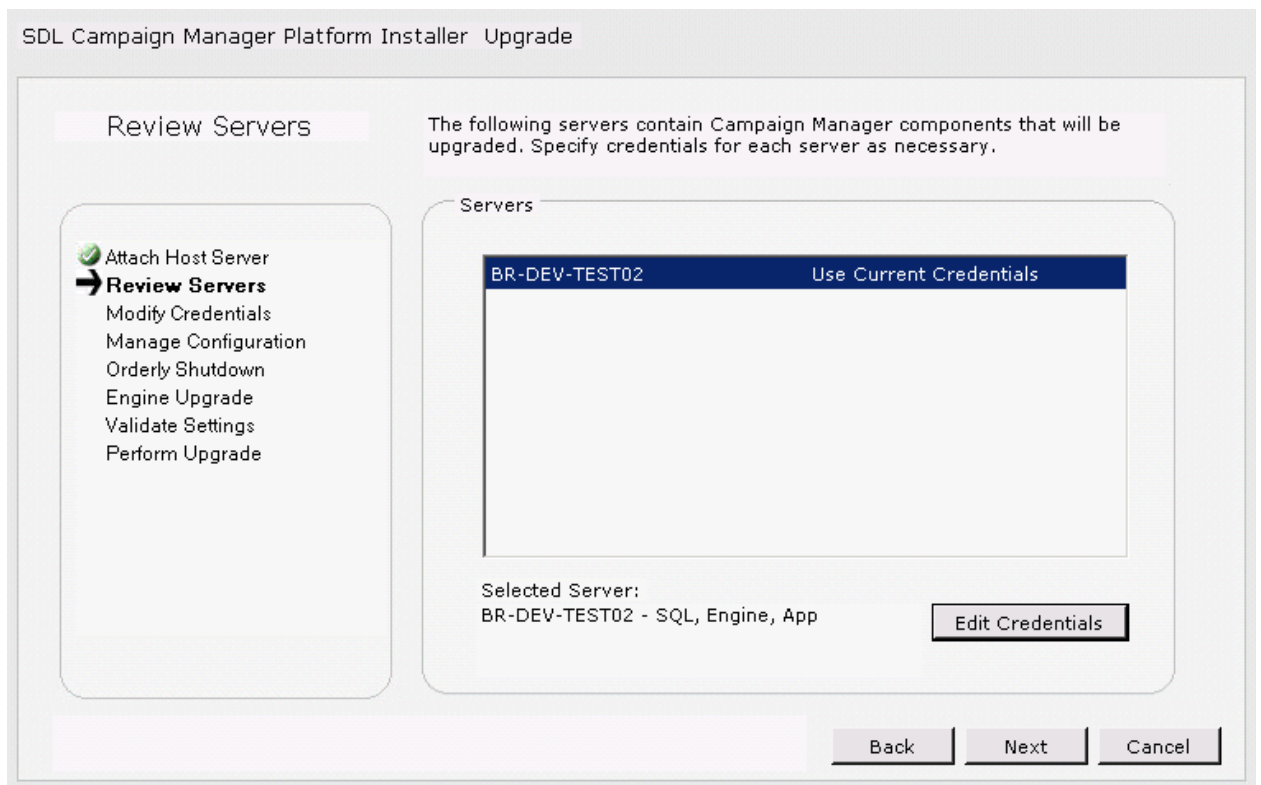
☐ Use SQL Server Authentication

User Name:

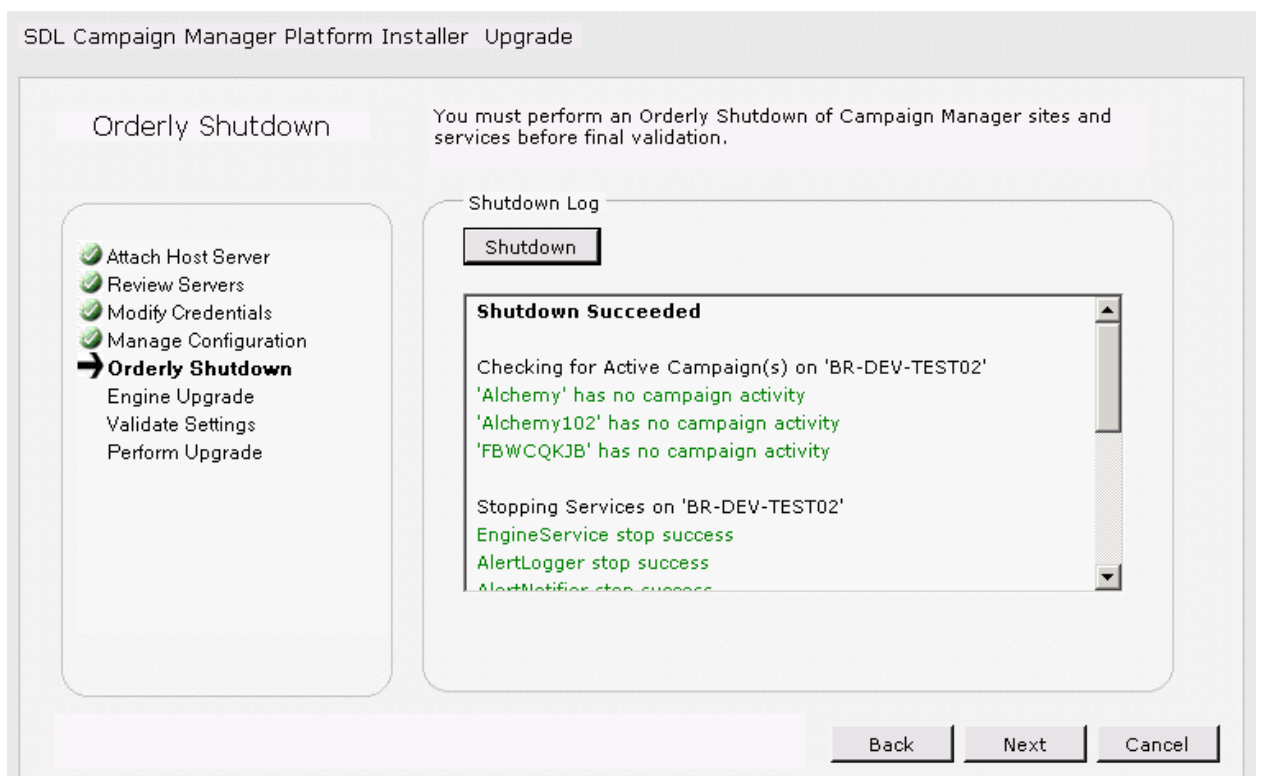
Password:

Back Next Cancel

5. Click **Next**.
6. The configured Application, SQL, and Engine servers are displayed. For a standard upgrade there should be no need to edit server credentials.
7. Click **Next**.



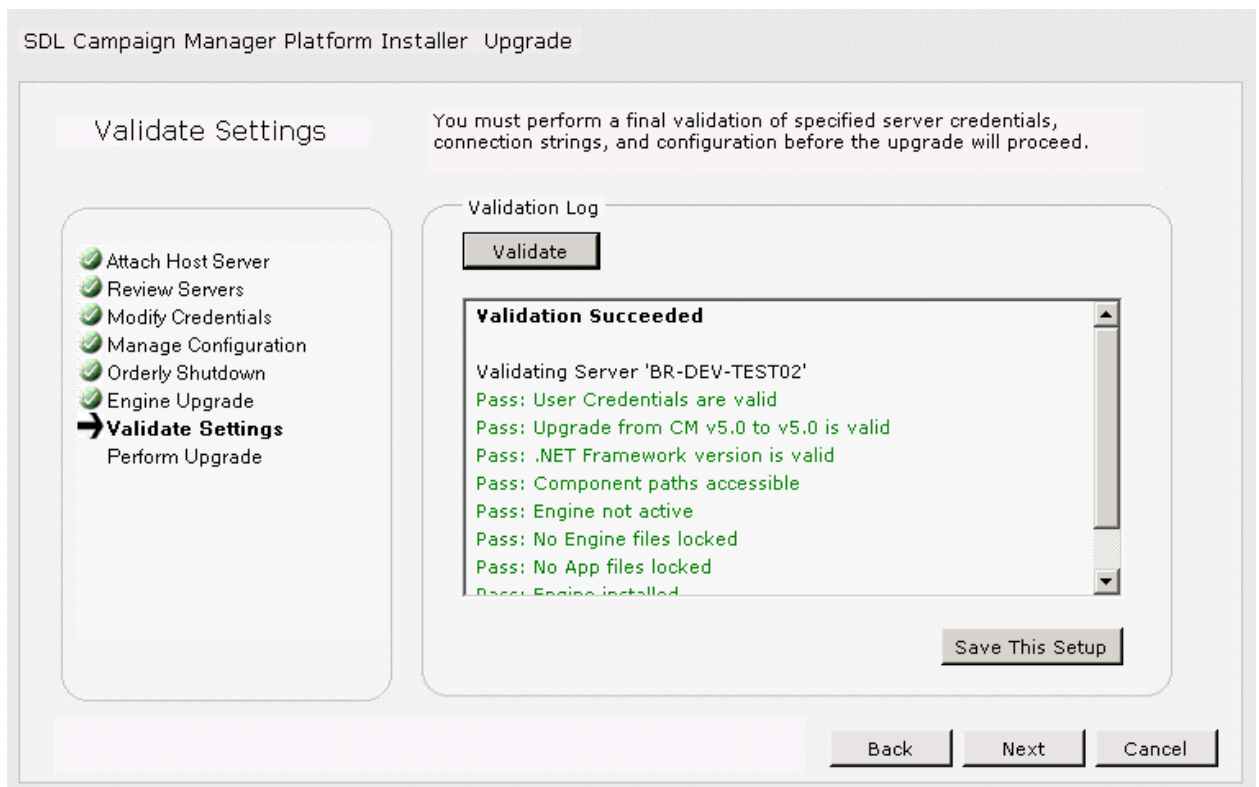
8. Perform an orderly shutdown by clicking **Shutdown**. This process shuts down the Campaigns sites and services.



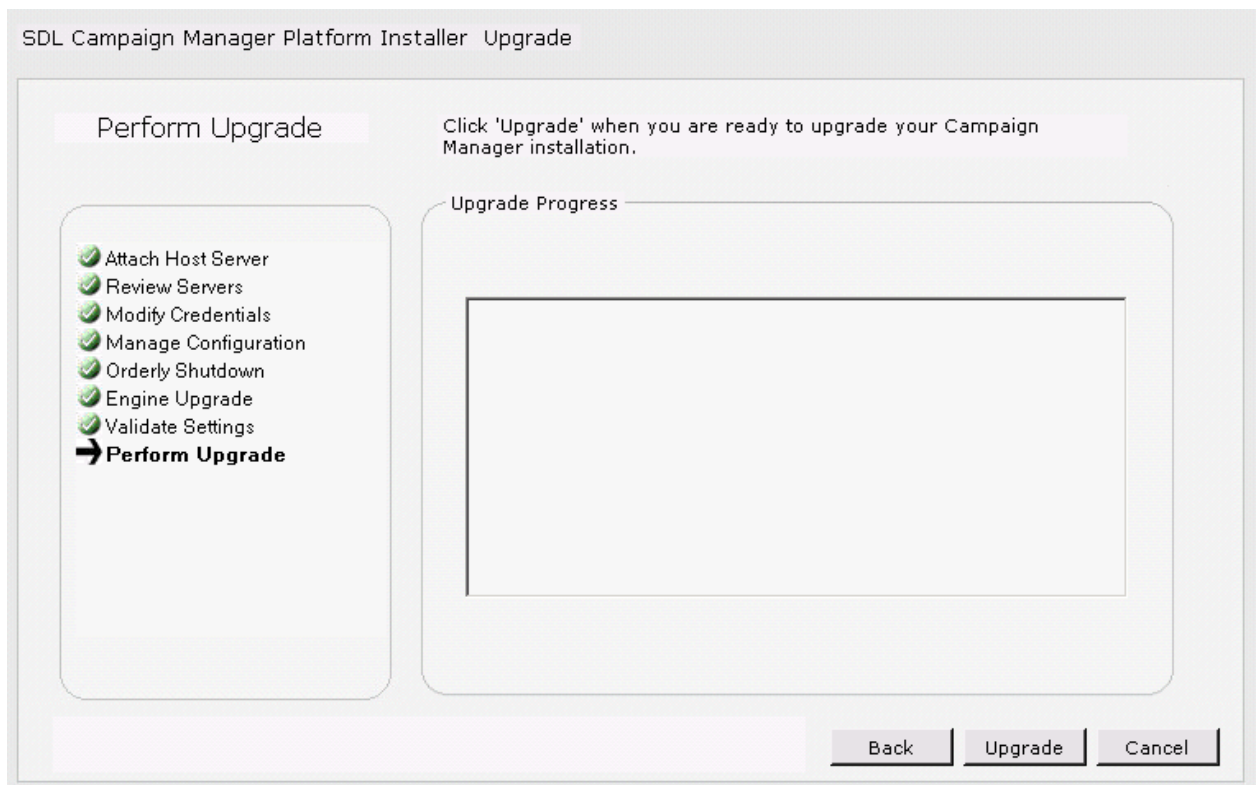
9. Click **Next** when the shutdown has completed successfully.
10. Specify the service account credentials for the Engine NucBroker service installed as part of Engine. The account must be a local admin and have the “logon as a service right” on App and Engine server/s. For upgrades, it is highly recommended to use the ConnectionBroker service account as previously defined.

The screenshot shows the 'SDL Campaign Manager Platform Installer Upgrade' window. The title bar reads 'SDL Campaign Manager Platform Installer Upgrade'. The main window has a light gray background. On the left, there is a vertical list of steps: 'Attach Host Server', 'Review Servers', 'Modify Credentials', 'Manage Configuration', 'Orderly Shutdown', 'Engine Upgrade' (highlighted with a right-pointing arrow), 'Validate Settings', and 'Perform Upgrade'. To the right of this list, a message states: 'The Engine upgrade will apply version 7.0.5508.32188 built on 30 January 2015'. Below this message is a section titled 'Service Account' with the instruction: 'Select the user account which should be designated as the owner for Engine NucBroker services'. There is a checkbox labeled 'Designate specific account for NucBroker services' which is currently unchecked. Below the checkbox are three input fields: 'User Name' (containing 'SYSTEM'), 'Password' (containing '\*\*\*\*\*'), and 'Domain' (empty). To the right of the 'User Name' field is a 'Browse...' button. To the right of the 'Domain' field is a 'Test Login' button. At the bottom right of the window are three buttons: 'Back', 'Next', and 'Cancel'.

11. Click **Next**.
12. Click **Validate** to check all entered credentials and configuration settings.
13. Click **Next** when the validation has successfully completed. Note you will not be able to click **Next** until validation has successfully completed.



14. Click **Upgrade** to start the upgrade.





When the software upgrade completes there is a further stage to carry out the data migration.

If upgrading from CM 4.x there are two migration steps for the creation of the new three Table Campaign history Data structure and updating of Campaign Activity Data.

If upgrading from CM 5.0.x there is one step for the updating of Campaign Activity Data.

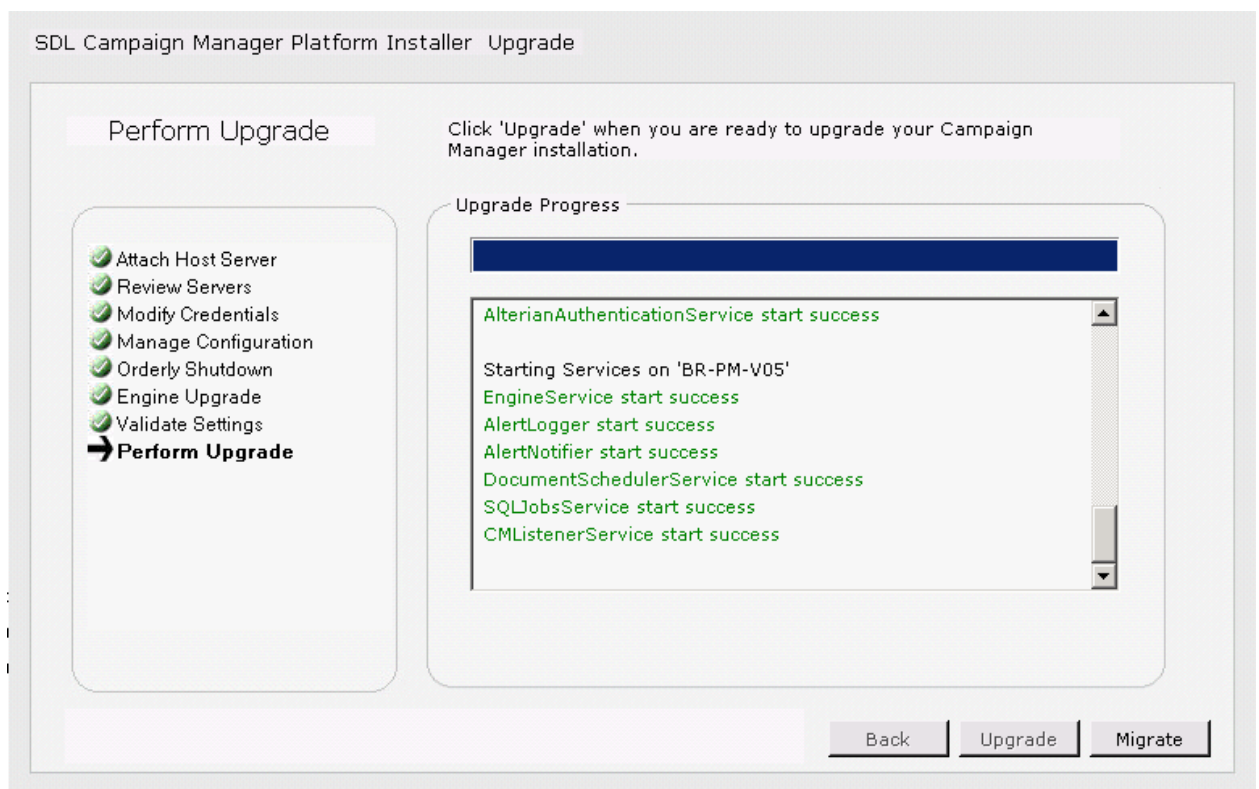
Migration steps are performed by a utility that will launch from the Deployer by clicking the **Migrate** button on the final screen.

---

**IMPORTANT  
NOTE**

This is a mandatory stage and under no circumstances should be skipped, as failure to perform this migration will leave the system in a non-functioning state.

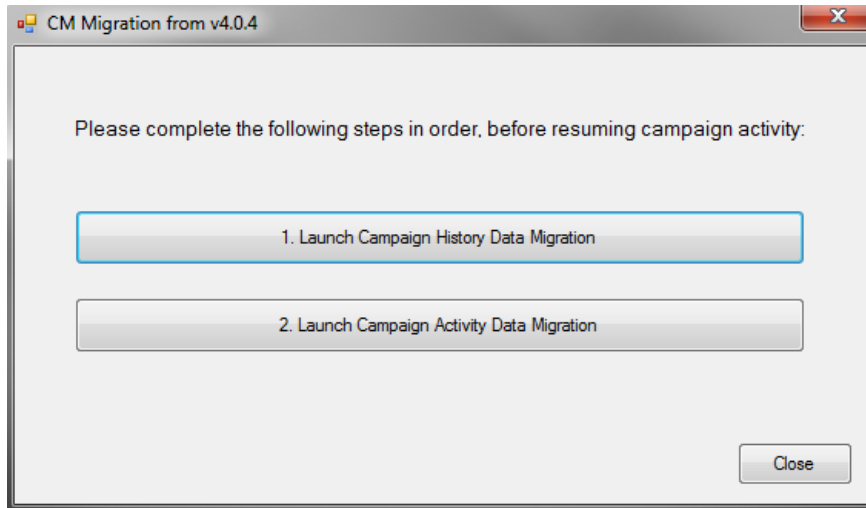
---



### 3.5 Upgrading from SDL Campaign Manager 4.0.x to SDL Campaigns 2016 R1

Only carry out the tasks in this section if you are upgrading from SDL Campaign Manager 4.0.x to SDL Campaigns 2016 R1

The following menu only appears when upgrading from Campaign Manager v 4.x. Complete step 1: Launch Campaign History Data Migration before launching and completing step 2: Campaign Activity Data Migration .



### 3.5.1 Campaign History Migration Utility

The migration utility is a mandatory stage of the upgrade process from Campaign Manager 4.0.x to SDL Campaigns 2016 R1 and executes a process to create the new three tables Campaign History Data Schema derived from the existing two Campaign History tables.

1. The initial screen gives an overview of the functions the utility will perform. Read this message and, once understood, click **Next**.



CM 5 Migration

SQL Server Name with Instance

**Data Upgrade**  
This utility will create the new Campaign History table structure from existing standard Campaign History data to support the Campaign Manager 5.0 processing and dashboards.

**Actions**  
This utility will perform a series of actions to meet the needs of the Single Event View schema and dashboards. It will run against each client, datasource and campaign key in the system. It will perform the following actions:

- 1) Create a new Master Campaign table from data in the existing Campaign History and Response tables
- 2) Create a new Contact History table from existing data
- 3) Create a new Response History table from existing data
- 4) Create all necessary links for the tables
- 5) Create a report outlining the success of the tasks.

**IMPORTANT NOTE:** Existing tables will NOT be altered by this process but will no longer be appended to. These tables should be reviewed and archived/deleted when no longer required.

2. Complete the screens as required with the information gathered previously. Next and Back can be used to double-check all configurations. The actual migration of all data is a one-off process for all Campaign Keys at the final stage of the utility so does not perform one key at a time.

**CM 5 Migration**

SQL Server Name with Instance:

---

Client Name:   
 Datasource Name:   
 Server:  Project:   
 Source History Database:   
 Destination History Database:   
 Master Campaign Table Name:   
 Datasource enabled: ☐ Datasource id:

The following campaign keys exist for this datasource. Please define names for the new Contact and Response history tables. NOTE that these cannot be edited after this process has ran.

Campaign Key	Existing History Table	New Contact Table	Existing Events Table	New Response Table	Do Not Migrate Data
[Demo].[Customer].[Cust ...]	Demo_Customer_Cust ID	Contacts	Responses_History	Responses	<input type="checkbox"/>

NOTE: The '*Do Not Migrate Data*' option is a one-time decision not to migrate data in Campaign History. This must only be used for Datasource or Campaign keys that are no longer going to be used. If a Datasource has more than one campaign key, all MUST be migrated at the same time.

The following scenarios are available when migrating tables:

- **Scenario one:**

Campaign Key, Contact Table, Response Table

- **Scenario two:**

Campaign Key, Contact Table, no Response Table

- **Scenario three:**

Response table

- **Scenario four:**

Campaign Key

In all scenarios if you wish to use the Campaign Key as it was in version 4.0.x then you must use the migration tool in order to use the new three table structure.

Enter a Name for the Master Campaign Table, if no name is given then a default of Master is applied.

All Campaign Keys and any responding history and results tables for that key will default into the migration tool. Should you not wish to migrate the data for this key select the **Do Not Migrate Data** checkbox.

Note the Existing Events Table is not automatically populated and needs to be entered manually.

In all scenarios a name must be given to the New Contact Table and New Response table. This name is unique and once used cannot be used again in the same datasource.

CM 5 Migration

SQL Server Name with Instance: Data Source=BR-PM-V05;Initial Catalog=almain;Integrated Security=True;Application Name=Campaign Start Up

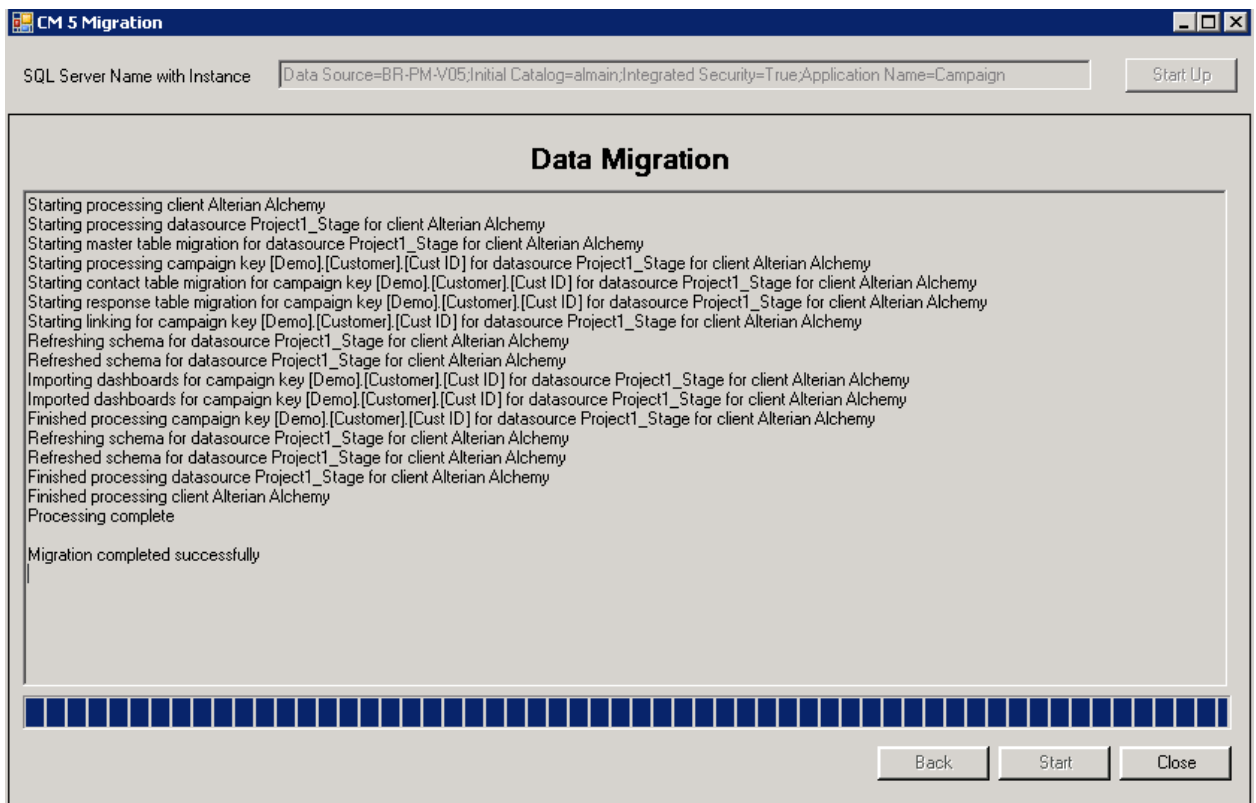
Client Name: Alterian Alchemy  
Datasource Name: Project1\_Stage  
Server: BR-PM-V05 Project: Project1\_Stage  
Source History Database: CM4\_History  
Destination History Database: Campaigns\_History  
Master Campaign Table Name: Master Campaign  
Datasource enabled: ☐ Datasource id: 106

The following campaign keys exist for this datasource. Please define names for the new Contact and Response history tables. NOTE that these cannot be edited after this process has ran.

Campaign Key	Existing History Table	New Contact Table	Existing Events Table	New Response Table	Do Not Migrate Data
[Demo].[Customer].[Cust ...]	Demo_Customer_Cust ID	Contacts	Responses_History	Responses	<input type="checkbox"/>

Back Next Finalize

3. Click **Finalize** when configuration is complete for all Campaign keys, across all Datasources, and across all Clients All previous steps were configuration only, it is not until this point that the data migration occurs.
4. A final **Start** button is displayed with a report page that shows progress.
5. The Migration process report is displayed.



The report is also available in the AdditionalSteps\5.0-CampaignHistoryMigration folder of the unzipped Campaign Manager deployer package in the CM 5 Migration Log.txt file.

6. Click **Close** when the history data migration has completed.

### 3.6 Campaign Activity Data Migration Utility for SDL Campaigns 2016 R1

Run the Campaign Activity Data Migration Utility whether you are upgrading from SDL Campaign Manager 4.0.x or SDL Campaigns 2015 R1

If upgrading from Campaign Manager 4.0.x launch the utility from the menu button after completing the History data migration. If upgrading from SDL Campaigns 2015 R1 launch the utility directly by clicking the **Migrate** button on the final deployer screen.

The Campaign Activity Data migration utility updates existing Campaign documents so any future Email deployments are reported correctly in the Campaign Activity Monitor page.

The utility only updates API Paused campaigns. Any campaigns paused in the Silverlight UI are skipped, but future Email deployments are reported correctly in Campaign Activity Monitor after the campaign is saved and started.

The utility reports progress in a command window, followed by a summary count of updated campaigns. When completed, follow the instruction *Press the Return key to exit* to close the window

To troubleshoot any issues, a log file is generated in the AdditionalSteps\5.1-CAMDataMigration folder in the unzipped deployer package on the App server.

## 3.7 Post-Migration Step

After completing the migrations steps, the system is ready for review, UAT testing and final approval prior to re-enabling Datasources and re-starting all campaign activity.

## 3.8 Post Upgrade Configuration

### 3.8.1 SQL Server

This section details the extra optional configuration steps, taken on the SQL Server once the upgrade has successfully completed.

1. Within SQL server management studio select Security > Logins > [AlchemyDSA]
2. On the **Server Roles** tab
  - Check the *'bulkadmin'* and *'public'* roles.
  - Check the *'dbcreator'* role if you are planning to create additional client databases.
  - Clear the *'sysadmin'* server role.
3. On the **User Mapping** tab:
  - Check the *'public'* and *'db\_owner'* roles are checked for the *'Alchemy'*, *'AlchemyStore'*, *'Almain'*, *'AlterianAuth'*, and any additional client databases.
4. Provide access to the alchemy domain service account to xp\_cmdshell. In SQL Management Studio.
5. Click the master database.
6. Select **New Query**.
7. Enter the following in the New Query window:

```
EXEC sp_xp_cmdshell_proxy_account 'DOMAIN\user','password'  
  
GRANT exec ON xp_cmdshell TO [DOMAIN\user]
```

8. Change *'DOMAIN\user'* and *'password'* to the username and password for the alchemy domain service account.

### 3.8.2 Engine

1. Check and ensure that the Engine is configured to use external surrogates when used with Campaign Manager to increase resilience as follows:
  - a. Use AMC -> Engine and Server Configuration -> Process Configuration.
  - b. In 'machine settings' change the machine name from 0.0.0.0 to 127.0.0.1
  - c. Click **Update Machine** then **Save**.
2. Check the following setting is correct for campaign manager use:
  - a. Use AMC to set the following configuration option using the Engine Configuration Applet:
    - AllowDuplicateColumnsInTemplate=TRUE
3. In AMC select the Engine project and navigate to Engine & Server Configuration then Process Configuration and enter an internal location for the machine setting temp folder in both the General and Machine Setting areas. Failure to do this means that the c:\windows\temp area is used which may result in running out of disk space. This step needs to be repeated for every Engine project.

## 4 Uninstalling SDL Campaigns

A full uninstall of SDL Campaigns can be carried out using the Deployer. Note that this will remove all SQL Server databases, all shares and system files, and all related services. Please ensure that the BCP folder has been removed from the SQL server, and the temp folder has been removed from the App Sever. Backups should be taken if there is a requirement to retain any of these files. Prior to uninstall, the administrator should shut down Engine.

1. Ensure Engine is shut down.
2. Save the Installer package to the App Server.
3. Run the **CM.Deployer.exe** file. This will launch the Installer wizard.

**Note:** If you have locked down the SQL Server permissions these will need adjusting so that the user is a member of the sysadmin role, otherwise the uninstall will fail due to insufficient permissions.

4. Select the Uninstall option and run navigate the wizard selecting applicable options.
5. Uninstall Engine using the Control Panel Add/Remove programs option from App and Engine server/s.

# 5 Troubleshooting

## 5.1 CM Upgrade troubleshooting

For each deployer run, a CM.DeployerLog.txt file is created in the same folder as the deployer EXE. This contains information logged during usage of the wizard and execution of the install/upgrade/uninstall. If reporting an issue with SDL Support, please also include the log file to assist with investigation.

## 5.2 Engine Upgrade troubleshooting

The Engine upgrade sequence involves the following steps:

- Cleanup of Engine 6.x components
- Installation of Engine 7.0.0
- Migration of Engine 6.0->7.0 configuration

A separate SysAdminApp utility is executed to perform the Cleanup and Migration steps for Engine instances on App and Engine server/s.

If the utility returns an error:

- A “failure” message will be displayed in the Deployer GUI
- The deployer run will be aborted before attempting the CM upgrade
- Detailed logging for the failure will be included at the end of the CM Deployer Log file

After resolving the issue re-run the CM Deployer to attempt upgrade again.

## 5.3 Engine Installation troubleshooting

The Engine Suite component is installed by the Campaigns Deployer on the App and Engine server/s. If the Engine installer returns an error:

- The MSI error code will be displayed in the deployer GUI
- The Engine installation will be rolled-back, and the deployer run will be aborted before attempting SDL Campaigns.
- Detailed logging for the failure will be included at the end of the CM Deployer Log file



Summary information for MSI return codes can be found here: <http://msdn.microsoft.com/en-us/library/aa376931%28v=vs.85%29.aspx>

To investigate any failure, the full Engine MSI and bundle log files will be also located in:

- The Deployer package Integrations folder when Engine is installed on the App server
- The Alchemy share TempInstall folder when Engine is installed on a dedicated server.

The main MSI log file will be named in the following format "Engine.InstallLog-YYYY\_MM\_DD\_HH\_MM\_SS\_pkgEngineMsi.txt" .

Take the MSI error code displayed in the Deployer, and locate all instances of this in the MSI log file (e.g. using notepad). The MSI log file should help provide further details on the issue. After resolving the issue, re-run the deployer to attempt install again.

## 6 Campaign Manager Validation

The validation step performs a range of checks, to highlight potential issues with the install/upgrade.

The section provides additional information, to help with resolving error messages that may appear.

### 6.1 Fail: Upgrade from CM v999 to v999 is not valid

Check the existing installed CM version. The deployer only supports upgrade from CM versions 4.x or 5.0.x to this release

### 6.2 Fail: OS is Invalid

Check all servers have Microsoft Windows Server 2008 R2 SP1 (x64) English Language Standard, Enterprise, Web Server edition installed, or Windows 2012 R2 Standard edition

Microsoft Windows 7 Professional is also allowable, but not supported.

### 6.3 .NET Framework 4.5.1 required on App, Engine and SQL Server/s

Ensure that .NET framework 4.5.1 is installed on each server.

## 6.4 Current User is not a member of the Built-in Administrators Group

Recheck the pre-reqs for each server and add the common domain account as a local admin. Ensure the account has “logon as a service” right on each machine.

## 6.5 IIS6 WMI Management Compatibility must be installed on the App Server

Recheck the IIS pre-requisites for the App Server to ensure this is included.

## 6.6 Invalid Kettle Zip File

Contact support as the supplied kettle zip could be corrupt.

## 6.7 Fail: Server is not accessible

This indicates the server cannot be accessed via Ping. Check the items in the installation guide for example, firewall configuration between servers. Check the common domain account has local administrator access to all servers; check the servers are accessible on the domain for example by using the command line Ping utility to verify accessible.

## 6.8 Fail: User Credentials are invalid

Check the common domain account has local administrator access on all servers.

## 6.9 Fail: Framework version is invalid

Check all servers have Microsoft .Net framework v4.5.1 64 bit installed.

## 6.10 Fail: SQL is not valid

The deployer expects SQL 2008 R2 Standard or Enterprise Edition product version 10.50.nnnn.n, SQL Server 2012, or SQL Server 2014 SP1.

To check the product version use `SELECT @@Version` on the SQL Server.

## 6.11 Fail: IIS is invalid

The deployer expects IIS 7, IIS8 or above.

## 6.12 Fail: Email Manager Account Link Name must be specified

The link name must be completed in Step 17 – Configure Email Manager link. Clicking the [Test Connection] button will populate this with a default link name, based on the Email Manager account details.

### 6.13 Fail: Email Manager unable to validate URL

Check the Email Manager URL is reachable from the App server (e.g. using Internet Explorer)

### 6.14 Fail: Alchemy Service Credentials are invalid (or) Fail: Service Credentials failed DB Connection

Check the service account has local admin rights on each server.

Check the account has a corresponding login on the SQL Server. On a clean install, this should be a member of the sysadmin role, before proceeding with locking down the permissions after the install.

### 6.15 Fail: AltEngineCampaign.DLL locked; Fail: EMTools.DLL locked; Fail: APIClientDLL locked; Fail: EngineMutex.DLL locked

The CM Deployer needs to update files that can be held open by Engine sessions.

At the validation stage, the CM Deployer should have already completed an orderly shutdown of CM sites and services.

To release locked Engine files, wait for active processing to complete, before using the AMC Orderly shutdown option on the affected Engine server/s.

The Microsoft Windows Process Monitor tool can also be used to confirm which process is holding the file open.

### 6.16 Fail: Inaccessible xxxx Component (ID n)

The deployer checks that each component is accessible prior to upgrade.

The component information is obtained from the Almain.AL.ProcessIndex table, which contains an ID, hostname, and networkPath for each component.

The validation message will provide an ID for a component entry that cannot be reached.

This may be due to accessibility problems (e.g. the deployer cannot reach the file share indicated by the networkPath). To verify this, use Windows Explorer to ensure the networkPath can be reached, and is writeable (e.g. create a temp file).

If the component entry is invalid (e.g. the networkPath doesn't exist) please contact Support. This may be due to a previously failed install, and the component should be reviewed and removed if necessary before retrying validation.

## 6.17 Fail: Install Folder cannot be created and shared

This fail occurs if the deployer cannot create or share the installation folder due to insufficient permissions. Perform the following checks:

- That the deployer was started using the 'Run as Administrator' option
- The deployer user is a local admin
- That the File and Print sharing feature is enabled on the server
- That the firewall is disabled
- The Remote Registry Windows service can be started (i.e. startup mode Automatic)

## 6.18 HTTP Error 500.0 - Internal Server Error Calling LoadLibraryEx on ISAPI filter "C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet\_filter.dll" failed

This fail occurs if an ISAPI filter entry exists for "ASP.NET\_4.0.xxxxx" referencing the 32 bit .Net framework. To resolve the issue:

1. Open IIS Manager on the App Server.
2. Click the server and double click ISAPI Filters.
3. If an ISAPI filter entry exists for "ASP.NET\_4.0.xxxxx" referencing the 32 bit .Net framework, remove this entry.
4. The .NET framework 4.0 entries for "ASP.Net\_4.0\_32bit" and "ASP.Net\_4.0\_64bit" are **valid** should be **retained**.
5. Browse to Campaign Manager after removing the ISAPI entry.

## 6.19 Fail: Engine component 'nn' is active. Wait or shutdown

Engine needs to be inactive for the upgrade to complete. If this refers to an Engine process, wait for active processing to complete before attempting AMC Orderly shutdown. If the specified process refers to a user application (AMC, iLoader or MMC) then close the application before retrying validation.

## 6.20 Fail: Engine already installed

For a clean install of CM 5.1, the Engine server must not contain any previous version of Engine. Please uninstall Engine via the Control Panel Add/Remove programs option from the specified server and retry.

## 6.21 Fail – SQL Filestream: full access not configured

Check the SQL instance has Full filestream access enabled in SQL Server Management Studio. In the SQL Server Configuration Manager, check the server has Filestream enabled for Transact-SQL and File I/O access.

## 6.22 Fail – Engine v6 or higher not installed

For upgrade to SDL Campaigns 2016 R1 only an existing Engine v6 or later is supported.

## 6.23 Fail: Engine NucBroker service Credentials are invalid

Check the service account has local admin rights on each server – see section 2.1

## 6.24 Fail: Engine ProjectBase path exists

For clean install, an existing Engine installation Projectbase folder must not exist. Remove or rename the specified folder.

## 6.25 Fail: Engine Install Path not accessible; Fail: Engine ProjectBase Path not accessible

Check the specified path can be accessed on the Engine server. The drive should be writable for access by the CM Deployer user account.

# 7 Appendixes

## 7.1 Appendix A – Adding Email Manager URLs

This appendix documents how to add an Email Manager URL to Campaign Manager 5.1 if the Email Manager URL was not created at the time Campaign Manager was deployed.

db\_owner access to Almain, AlterianAuth and CM client databases are needed to run the utility. The utility uses Windows Authentication to connect with the SQL Server.

1. Run the SysAdminApp.exe utility located in the PauseResume folder under the CM install location on the Engine Server.
2. Open a command prompt and change directory to the PauseResume folder location.
3. Run the utility using the following command:
  - Sysadminapp /sqlserver=*sqlservername* /addEMSite

If you have multiple Campaign Manager URLs configured the utility will ask which Campaign Manager URL to associate the new Email Manager site with.



Please replace sqlservername based on your SQL Server name. If the Campaign Manager SQL Server isn't the default instance, supply the sqlserver as sqlservername\instancename. If SQL Server is installed as the default instance, please use the machine name rather than (local) or ".".



The utility only adds an Email Manager site if an Email Manager site has not already been configured.