

SDL Tridion

Large Scale Enterprise CMS Releases



- 1 INTRODUCTION 3**
- 1.1 AUTHOR INFORMATION 3
- 1.2 DOCUMENT OVERVIEW 3
- 2 SDL TRIDION 4**
- 2.1 DTAP AND ENVIRONMENTS. 4
- 2.1.1 DEVELOPMENT 4
- 2.1.2 TEST 5
- 2.1.3 ACCEPTANCE 5
- 2.1.4 PRODUCTION 6
- 2.2 BLUEPRINT STRUCTURE 6
- 2.3 WORKFLOW 7
- 2.4 APPLICATION ENVIRONMENTS 8
- 3 MANAGING RELEASES 9**
- 3.1 RELEASE TEAM PACKAGES 9
- 3.2 RELEASE SCHEDULES AND PLANNING 11
- 3.3 RELEASE DAY PLANNING 12
- 3.4 RELEASE PUBLISHING 13
- 4 CONSIDERATIONS 14**
- 5 ABOUT CONTENT BLOOM 15**



1 Introduction

This document provides information to assist Global Organizations develop a strategy for creating a release process within their SDL Tridion environments.

No two customers have the same internal infrastructure, technologies, teams or processes. It's also true that organizations in certain industries (for example Finance) will have different regulatory rules to consider as part of a release.

Content Bloom has a collective experience of over 50 years in working with SDL Tridion, we've helped some of the world's largest companies define release and governance strategies around CMS implementation. The current team at Content Bloom comprises of 5 SDL MVPs, and 22 SDL Certified Architects, Developers, Business Analysts, System Administrators and Trainers.

Should you have questions after reading this document, please contact our SDL team at sdl@contentbloom.com. Our website at <http://www.contentbloom.com> contains our global locations and telephone numbers should you wish to call.

1.1 Author Information

Paul Russell is a Senior Consultant, working at Content Bloom's office in Halifax, Nova Scotia, Canada. He has over 30 years experience in designing and delivering applications in more languages and technologies that he can count.

1.2 Document Overview

The remainder of this document is broken up into the following sections:

- Some key information related to the SDL Tridion Content Management Section
- Managing a global release
- Key considerations to include in your planning of release process creation.

2 SDL Tridion

SDL Tridion is a unique CMS in that is created to manage any number of web sites, applications, native mobile apps etc. from a single management interface.

2.1 DTAP and Environments.

When you create and build websites they are not typically created 'Live'. Normally a number of *environments* are created to help *development* teams build and business teams *approve*, but it is placed *Live*.

In large organizations it's common to see that 'DTAP' is the method used to define environments:

- **D**evelopment
- **T**est
- **A**cceptance (commonly known as U.A.T.)
- **P**roduction

Each environment has its own purpose. A brief overview of their purposes follows:

2.1.1 Development

The Development environment is typically used by the development team to build and update existing project work such as:

- New content, templates, designs, and web pages.
- Template building blocks, GUI extensions, Core Service related tools that are used within the Content Manager, Custom Dreamweaver functions, event and workflow code.
- Deployer and storage extensions that take place during publishing.
- Content Delivery and application code used by the published application.

It's also in development that system administrators will install and test SDL Tridion / SDL Web hot-fixes, patches or newer versions of software.

Once development of the work is completed in this environment, it is migrated to the test environment. This is typically referred to as a 'release'.

2.1.2 Test

The Test environment is used to test that the release is:

- Free of bugs.
- Migrates to another environment without issue.
- Does not cause any unexpected issues from content creation, publishing and content delivery etc.
- This environment may also be used to perform other types of testing, for example concurrency and load.

If the testing is success then the same package that was migrated from Development to Test is then migrated to Acceptance.

If the testing fails, the package is rolled back and issues and feedback are given to the development team.

2.1.3 Acceptance

The acceptance environment is used to demonstrate the release to the product owner(s) of the work, for them to 'Accept' that is correct.

The new features are:

- Reviewed against the functional requirements.
- Tested by the owner(s) to ensure they're happy with what has been built.

The product owner determines that the work is approved or not. If approved, the release is then deployed to the Production environment. If not changes and requests are given back to the development team.

2.1.4 Production

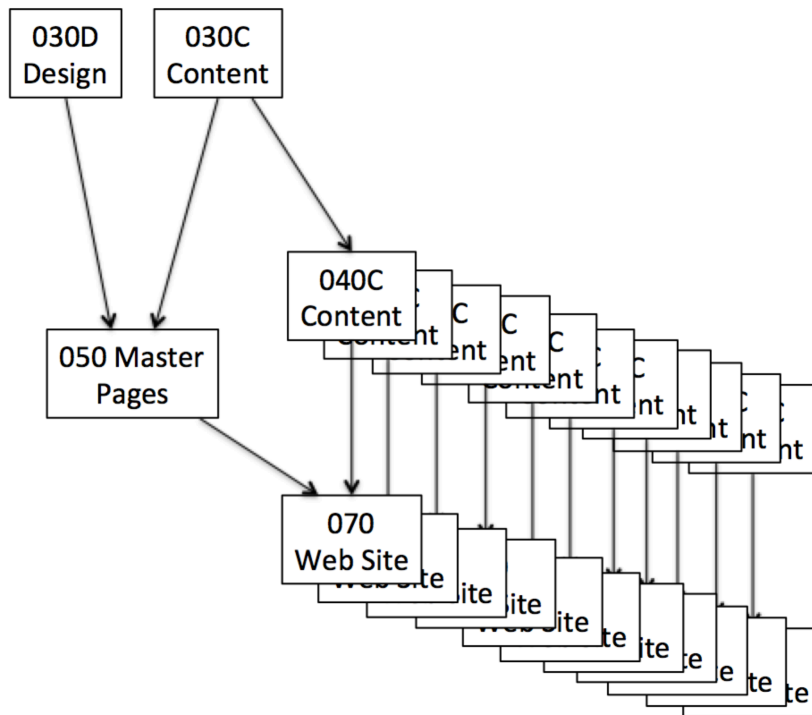
Production is the live environment. Functionality here is available to the visitors of your website, application or content etc.

2.2 Blueprint Structure

An enterprise Blueprint structure typically contains a large number of content and applications to manage:

- Internal internet / portals
- Multiple brand external websites
- Multiple content publications, which can contain tens of thousands of content components
- Multiple translation publications

The diagram below shows an example a Blueprint Structure within SDL Tridion



In our experience it is common for an SDL installation to contain over 400 Publications as part of a global enterprise implementation.

When migrating from one SDL Tridion environment to another, specific items (and often entire Publications) are exported from one environment to another.

SDL Tridion contains two useful elements to help facilitate:

- Content Porter – A standalone application that is used to select items to import / export between environments.
- Bundles – Bundles are containers that allow users to create collections of items. Content Porter can migrate a Bundle, which selects all elements for migration that are contained.

2.3 Workflow

Workflow allows Organizations to create a custom review process and flow for all content created within SDL Tridion.



When using Workflow within your SDL Tridion solution, the following points should be considered as part of the release process.

- Publishing of items are protected by workflow
 - Unapproved (minor versions) versions can be configured to publish to 'Staging'
 - Only Approved versions should be allowed to publish 'Live'
- Workflow bundles also need approval
- Once approved, some items need Translation, which can kick off new Workflow processes.

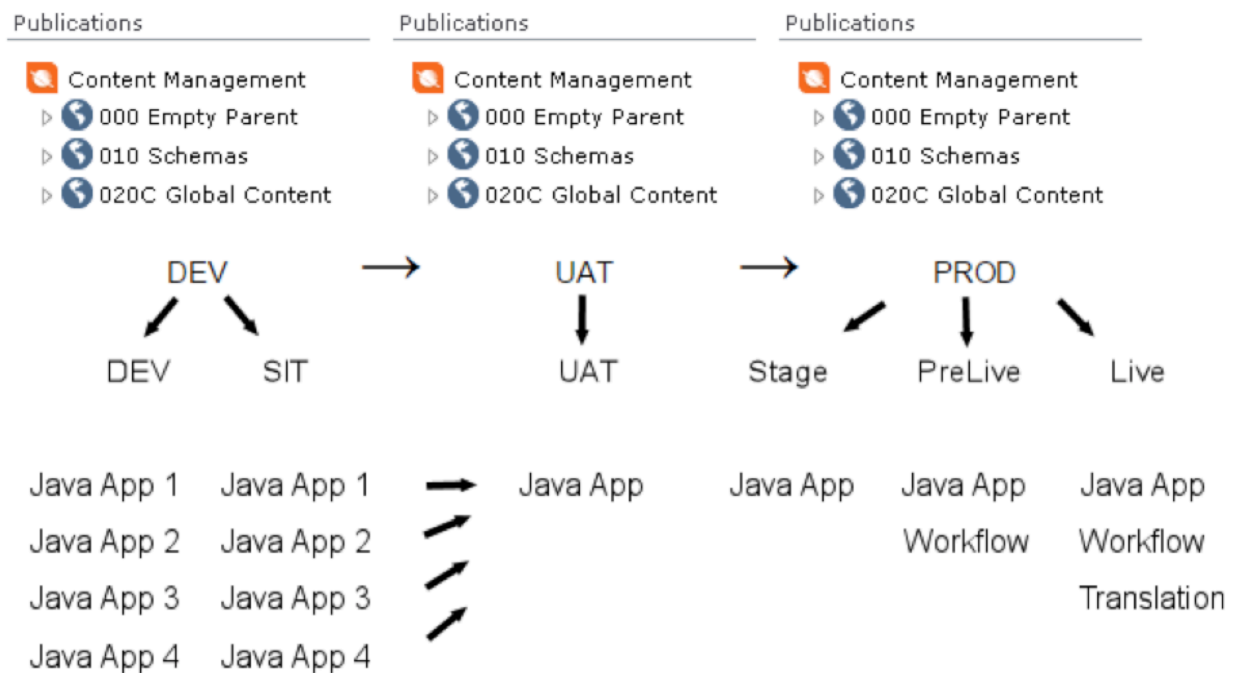
In this process, the content teams must verify and approve content and workflow bundles to ensure that upon release all content is available for 'Live' publishing.

2.4 Application Environments

As well as your SDL Tridion Publications and content items, application code releases must be considered as part of the release process.

The diagram below shows an overview of a typical the environment. Note the following properties:

- 3 SDL Environments (Development, UAT and Production)
- Publishing targets available at each environment
- Java Development teams working independently, integrating code into a single release for UAT testing
- Content Management, Workflow and Translation is managed by business teams in production



3 Managing Releases

Within large enterprises, there can be a number of development groups or sprint teams working on new functional items across multiple applications and regions.

Releases may include updates to application code, publishing content from SDL Tridion, and may have dependencies on other releases from other teams.

It is key that for large enterprise customers that a central owner (Release Coordinator) is responsible for managing the release schedule across all teams and stakeholders. Within an organization a number of releases are planned to take place over the course of the year.

3.1 Release Team Packages

The table below shows four sprint teams generating releases at the end of completed sprints:

| Sprint Team 1 | Sprint Team 2 | Sprint Team 3 | Sprint Team 4 |
|---------------|------------------|---------------|---------------|
| Release 1a | Release 2a | Release 3a | Release 4a |
| Release 1b | Release 2b | | Release 4b |
| Release 1c | Release 2c | | |
| | Release 2d | | |
| Patch 1a.1 | Patch 2b.1 | | |
| Patch 1b.1 | | | |
| | Patch 2b.2 | | |
| | Patch 2b.3 (UAT) | | |

These releases packages are bundled within SDL Tridion. To help automate use of these bundles, some teams are using a custom extension built by Content Bloom. This tool creates the packages and manages the dependencies within the packages. The tool also creates relevant scripts that are used with SDL's Content Porter software, the running of which can also be scripted to fully automate migrating a package from one environment to another.

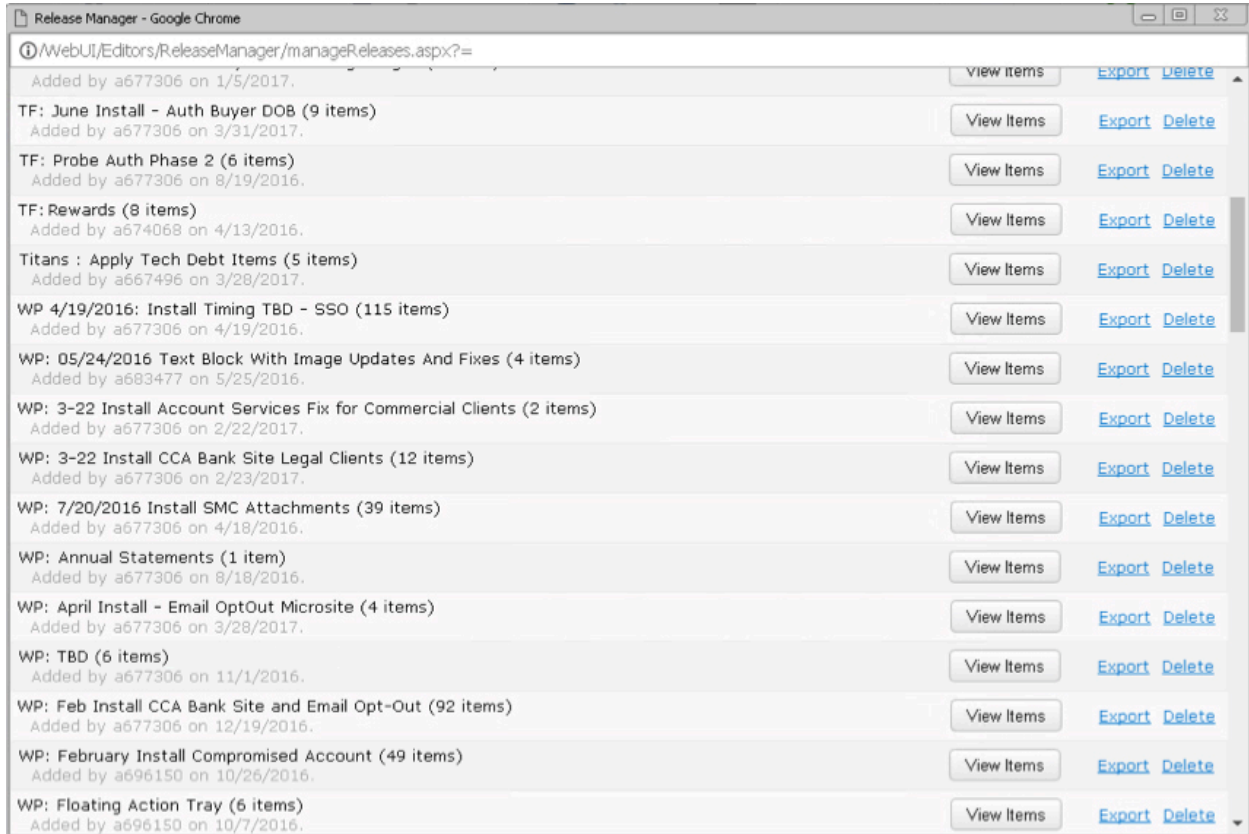


Fig 1. The Release Manager Tool Interface.

3.2 Release Schedules and Planning

A release schedule is mandatory to help the all stakeholders understand the volume of and planning around future releases.

| Install Date | Install Type | New or Updated Pages (per client) |
|--------------|--------------------|-----------------------------------|
| 1 Feb | Minor Install | 10 |
| 15 Feb | Major Install | 60 |
| 1 Mar | Minor Install | 15 |
| 15 Mar | Enterprise Install | 90 + Unpublish |
| 5 Apr | Minor Install | 14 |
| 19 Apr | Major Install | 40 + Unpublish |
| 3 May | Minor Install | 0 |
| 17 May | Major Install | 15 + Unpublish |
| 31 May | Minor Install | 3 |
| 7 Jun | Enterprise Install | List Not Complete |

A large number of releases are minor tweaks, so it is worth adding categorization and some basic information to the schedule.

In the sample release chart (above) a number of install types have been defined:

- **Major** - planned, regular in frequency, and usually incorporates new features or updates across multiple teams
- **Minor** - minor updates to existing features, usually requires little liaison across a number of teams

- **Enterprise** - Major updates that feature major application and technological changes and can include new infrastructure, third-party software, and networking updates.

3.3 Release Day Planning

To ensure that a release is successful, the Release Coordinator must plan and present the actions/steps that will take place during the day of release to all relevant stakeholders, typically the day before execution.

In large organizations, it's typical to perform an upgrade outside of high-demand hours. The table below shows an example release plan broken down by time starting at midnight.

The chart below shows an example of a day release schedule, starting at midnight:

| Time | Java App update | Tridion update | Testers |
|------|----------------------------|---|---------|
| 0000 | | Publish site metadata Publish priority clients | |
| 0100 | Route traffic to A servers | | |
| 0110 | Upgrade app on B servers | | |
| 0140 | | Priority clients complete | |
| 0145 | | Publish all clients | |
| 0230 | B upgrade complete | | |
| 0235 | Route traffic to B servers | | |
| 0245 | Upgrade app on A servers | | |
| 0415 | A upgrade complete | | |

| | | | |
|------|--------------------------|--------------------------------|---------------------------------------|
| 0430 | Load balance app traffic | | |
| 0450 | | Publish site metadata complete | |
| 0500 | | | First Event Validation testing starts |
| 0730 | | All publishing complete | Manual testing starts |
| 1000 | | | Testing Complete |

3.4 Release Publishing

Publishing actions should be clearly understood as part of the release planning process.

Consider a snippet from the release schedule table described earlier in this document:

| Install Date | Install Type | New or Updated Pages (per client) |
|--------------|---------------|-----------------------------------|
| 1 Feb | Minor Install | 10 |
| 15 Feb | Major Install | 60 |

The release on the 15th of February requires that 60 pages be published as part of this rollout. This is a Major install that requires all 400 child Publications are also published.

$$\begin{array}{rcccl}
 60 & & & & \\
 \text{Content Pages} & & \times & & 400 \\
 & & & & \text{Child publications} \\
 & & & = & \\
 & & & & \mathbf{24,000} \\
 & & & & \text{Publishable items}
 \end{array}$$

In order to manage this process, Content Bloom has created a tool to configure the rendering and publishing automatically. This tool can integrate with our scripted migration process or be run stand-alone.

4 Considerations

In addition to the information contained in this document a number of considerations should be taken into account when planning your enterprise release process.

1. Multiple releases may be taking place at the same time.
2. There may be tight coupling between an application and content coming from the CMS.
3. The release of new application code may be tied to configurations and content that must be published in order for the updated application to correctly function.
4. Pre-rendering within SDL Tridion is possible and out of the box, the problem is that the volume can be very intensive on the Content Management server. Having the DevOps / Infrastructure team team available to monitor the Content Management (and add more firepower) is often necessary.
5. Zero downtime in production applications is a must. A common recommendation is to have new machines containing the updated migration content and applications and once tested switching load to the new machines.
6. Using SDL's Content Porter does not support the removal of expired content, but Content Bloom does provide tools to perform such operations.
7. Content may need to go through a translation process as part of the release rollout to global countries and regions.
8. Communication and planning that is managed by a central release coordinator is key to ensure the facilitation and smooth release for large-scale global enterprises.
9. It is good practice to keep all environments as architecturally close as possible and to keep them maintained with the latest content and configuration. Once again Content Bloom can provide a number of useful tools to ensure environments are correctly aligned.

5 About Content Bloom

Content Bloom designs and implements large-scale web solutions using enterprise content management systems and cutting-edge technologies to provide measurable value to clients worldwide.

Content Bloom provides expert-level development and consultancy services for the following CMS vendors:

- SDL Tridion / Web
- Adobe AEM
- Drupal
- WordPress

We work with large companies and creative agencies that need technical expertise to provide:

- Web content management
- Customer experience management
- Custom application development in .NET and Java frameworks
- Software integrations
- Multi-language applications
- Architecture & consultancy

Content Bloom has offices in Canada, USA, India and has remote development teams across Europe:

Content Bloom Corp

106 East Sixth Street
Suite 900
Austin, Texas,
78701, USA

Content Bloom INC

First Canadian Place
100 King Street West, Suite 5700
Toronto, ON Canada M5X 1C7

<http://www.contentbloom.com>

Content Bloom Corp

106 East Sixth street
Suite 900
Austin, Texas,
78701, USA

Content Bloom INC

First Canadian Place
100 King Street West, Suite 5700
Toronto, ON Canada M5X 1C7
BN: 82392 9401