



**Release Management Policy
Aspen Marketing Services
Version 1.1**

John Toso
5/10/2010

Contents

Release Management Policy Overview:.....	3
Critical Success Factors	3
Service Level Management (SLM).....	4
Key Performance Indicators:.....	4
Service Level Agreements (SLA/OLA).....	6
Roles & Responsibilities	6
Service Manager:	6
Service Architect:	6
Release Manager:	7
Release Engineer:.....	7
Development Team:.....	8
Deployment Team:.....	8
Responsibility Accountability Matrix	8
Aspen Release Management Policy RACI Chart:.....	9
Release Tasks	9
Quality Gates.....	9
Release Readiness Checklist/System Verification Review (TEMPLATE)	10
Approvals	10
Release Management – Post Implementation Review (TEMPLATE)	11
Approvals	12
Current and Future State Release Management Process Comparison	Error! Bookmark not defined.

Release Management Policy Overview:

This document outlines the Aspen Marketing Services SDLC Release Management Policy. The goal of Release Management is to ensure that all aspects of a release, both technical and nontechnical, are considered together. Release Management as represented in this document is a subset of an overall IT Service Transition¹ solution which is comprised of Change Management, Release Management, Deployment, and Knowledge Management. Each of these sub processes inter-relates and a Release Management end-to-end solution must include key components of each process. The policies defined in this document focus on Release and Deployment Management. Change Management, Asset Management, and other service transition processes will be considered in future versions of this document.

The Release Management Policy governance was developed around the following areas:

- Ensure there is a distinction made between change management, development, implementation, and testing roles & responsibilities to optimize the integrity of the release process.
- Positioning additional quality gates in the Release Management life cycle to ensure consistency through the build and release process.
- Reducing the risk in failed releases by centralizing control, increasing visibility and automating manual tasks that can introduce error into the deployment process.
- Measure performance of Release Management to drive to continuous improvement of the process.

Critical Success Factors

The recognition and communication of Critical Success Factors (CSFs) for the Release Management process is crucial for enterprise-wide adoption of the policy.

- Planning for releases and delivery packages so these are delivered to production ready and tested to minimize the volume of changes into the environment.
- Release Manager will be empowered to veto a new release into the production environment based on acceptance criteria as defined in the Release Management process.
- Release Manager will ensure that adequate testing has been performed prior to submitting an approval for the release to go to production and the inclusion of a back-out strategy for all production changes.
- Measure performance of Release Management to drive to continuous improvement of the process

¹ As identified in the ITIL Service Management (ITSM) Lifecycle.

- Release Manager will have an active role and provide Change Management oversight through a formal Change Advisory Board (CAB) for all production changes (future).

Because it is recognized that portions of the Release process may be impacted by factors outside the organization, Release Management will report and highlight external issues where possible.

Service Level Management (SLM)

The objectives of Service Level Management are to negotiate, agree, document, and monitor service level agreements between business, IT, and service providers. Ensuring that high quality, repeatable, and predictable release deployments are achieved is the ultimate goal of the Release Management Policy. SLM metrics associated in this Release Policy will be organized into Key Performance Indicators (KPIs) mapped to existing Service Level Agreements (SLAs) or new SLAs will be established if necessary as the Release Policy Matures.

Key Performance Indicators:

Fundamentally, Key Performance Indicators (KPIs) measure progress towards a goal. Defining suitable KPIs is above all about deciding what exactly is considered a “successful process execution”. The table below provides recommended KPIs for Aspen Release Management Process with identified goals derived from information gathered from existing Aspen release deployment performance. These KPI goals align with expected deployment process improvements as this policy is implemented across the organization².

KPI Metric	Functional Role Responsible	KPI Goals	Measurement Instrument
Pre-Release Performance			
Number of incomplete assembled build package (deployment lead time SLA will influence measurement)	(primary) Release Engineer (secondary) Release Manager, Deployment Team	Less than 10% of deployment packages.	Deployment Request Document (TFS) <i>[Scheduling checklist]</i>
Number of releases without completed testing results	Release Manager	Less than 5% of deployment packages.	Deployment Request Document (TFS) <i>[Scheduling checklist]</i>
Number of releases that get rejected during Release	Release Manager	Less than 15% of deployment packages.	Deployment Request Document (TFS)

² See current and future state release management process for details.

KPI Metric	Functional Role Responsible	KPI Goals	Measurement Instrument
Readiness/ System Verification Review			<i>[Scheduling checklist]</i>
Release Performance³			
Number of releases deployed to staging with issues.	(primary) Release Engineer, (secondary) Release Manager	Less than 25% of deployment packages.	Deployment Request Document (TFS) <i>[Post Implementation Results]</i>
Number of releases deployed to production with issues.	Release Engineer Release Manager	Less than 5% of deployment packages.	Deployment Request Document (TFS) <i>[Post Implementation Results]</i>
Deployments performed on-time.	(primary) Release Manager (secondary) Deployment Team	95% of all deployment requests.	Deployment Request Document (TFS) <i>[Post Implementation Results]</i>
Number of Emergency deployments ⁴ (planned verse unplanned measure)	Release Manager	Less than 25% of deployment packages.	Deployment Request Document (TFS) <i>[Scheduling tab]</i>
Release deployments longer than scheduled.	Deployment Team	Less than 25% of deployment requests.	Deployment Request Document (TFS) <i>[Post Implementation Results]</i>
Post-Release Performance			
Number of releases deployed to production with software defects. Severity SLA to be established.	(primary) Development (secondary) Release Manager	Less than 5% of deployments.	Deployment Request Document (TFS) <i>[Post Implementation Results]</i>

³ Issues examples - deployment procedures, faulty scripts, SW defects, documentation errors, environment issues

⁴ Emergency defined as less that 4-hours notice. Criteria subject to change .

KPI Metric	Functional Role Responsible	KPI Goals	Measurement Instrument
Number of release backouts	(primary) Release Manager (secondary) Deployment	Less than 5% of deployments.	Deployment Request Document (TFS) <i>[Post Implementation Results]</i>

Service Level Agreements (SLA/OLA)

Future policy will structure KPI goals against Operational and Service Level Agreements intended to measure how effective inter-department and function areas serve each other's needs. Effective SLAs are extremely important to assure improvements in the process and increase the success rate of deployments.

Roles & Responsibilities

Aspen Marketing Services has designated a team responsible for ensuring the new release process is adhered to and ensure improvements are being made. The following list the roles and responsibilities for Aspen's Release Management:

Service Manager:

The primary role of the Service Manager is to ensure executive support of the Release Management Policy and drive the long-term strategic directions of the release process. Key activities for this role include:

- Ensure alignment of the Release Management Policy to the Aspen overall IT delivery strategy.
- Provide policy ownership through design, implementation and continuous improvement activities.
- Communicate the Release Management policy's strategic goals to the business units and development organizations.
- Work with each of the development teams to ensure the Release Management policy is executed as designed.

Service Architect:

The primary role of the Service Architect is defining the technical solution for how the Release Management Policy will be delivered. Key activities for this role include:

- Implementing a technical strategy for the Release Management process.

- Provides technical ownership for design, implementation and continuous improvement of the Release Management tool.
- Establishes security permissions for the Definitive Media Library (TFS).
- Automates components of the Release Management process wherever possible.

Release Manager:

The role of the Release Manager is to manage the Release Management process end-to-end and coordinate the various functions and work activities for each build. The Release Manager is focused on each specific release rather than the entire release process. The Release Manager represents business unit, Product Management, or area interests regarding the release being requested. Normally this role is the leader of the project team responsible for the release. Key activities for this role include:

- Communicate with and manage the expectations of customers of Release Management.
- Primary contact to Product Management concerning all release/deployment issues. Coordinates and communicates all go/no-go decisions.
- Update and manage content within the Deployment Work Item.
- Request/schedule deployments (master release calendar).
- Ensure that proper testing occurs for all releases into the production and staging environments
- Schedule and lead release readiness/system verification reviews.
- Lead post-implementation reviews (with deployment team)
- Review releases and assign appropriate release testing tasks.
- Receive, log, qualify, and assign all release requests.
- Conduct periodic process audits.
- Participate in Change Advisory Board (Future).

Release Engineer:

The Release Engineer oversees the technical content in the deployment request documentation and verifies the technical quality of each build. Key activities for this role include:

- Primary Release Manager contact for deployment packages, deployment issues, etc.
- Provides details on build package, scripts, and other deployment assets in the Deployment Request Work Item.
- Verifies the assembled build package for the release.
- Secures software and other deployment assets in the Definitive Software Library (source code control). These include software versioning and branching policies.
- Compile, Review, and deploy all Testing Deliverables.
- Assemble build package and conduct installation procedure tests.
- Participate in Release Quality Gates (as needed).
- Primary contacts for QA on build/deploy, or defect issues.

Development Team:

The Development Teams will responsible for development of software features and defect fixes. Key activities include:

- Working with Release Management to publish build, testing, deployment plans and schedules.
- Develop business solution, test, and create configuration and installation scripts.
- Work with Release Engineers to validate configuration, and installation scripts.
- Develop roll-back procedures.
- Conduct functional, integration, user acceptance, operational readiness, and back-out testing.
- Participate in Release Quality Gates (as needed).
- Resolve deployment issues (works with Release Engineer).
- Participate in post-release validation.

Deployment Team:

The Deployment Team is comprised of individuals from the Operations team and is responsible for actual deployments to Staging (AUTO & DIGITAL teams) and Production (ALL teams). Key activities include:

- Maintain consistency between staging and production environments.
- Maintain environment permission schema.
- Maintain primary release calendar.
- Deploy releases to staging and production and provide primary contact to Release Managers.
- Communicate deployment status though updates to the deployment request Work Item.
- Reports exceptions to the Release process.
- Participate in Post-Implementation reviews (as needed).
- Update deployment log/metrics for KPI reporting.

Responsibility Accountability Matrix

The following RACI matrix describes the participation by various roles in completing Release Management (ITIL Service Transition) tasks or deliverables. The RACI chart is useful in clarifying roles and responsibilities in cross-functional/departmental projects. For each task or deliverable, roles are identified as **Responsible** (those who do the work to achieve the task), **Accountable** (those who are ultimately accountable for the correct and completion of the deliverable or task), **Consulted** (those whose opinions are sought), or **Informed** (those who are kept up-to-date on progress). In most cases, the role that is Accountable for a task or deliverable may also be Responsible for completing the task.

Aspen Release Management Policy RACI Chart:

	Service Manager	Service Architect	Release Manager	Release Engineer	Development Team	Deployment Team
Release Tasks						
Create Release Package						
Write up Impact & Risk Mitigation Plan			A	I	R	
Write up Test Results			A	I	R	
Update Deployment Work Item			A	C	R	I
Write up Back out Plan/Procedures			A	I	R	I
Propose Implementation Schedule			I	I	R	I
Submit to Release Manager	I	I		C	R	I
Review and Approve Release						
Verify the assembled build package is complete			I	R	C	
Conduct installation procedure tests			I	R	I	
Release Readiness/System Verification Review			R	A		C
Report exceptions to the Release process			R	I	I	I
Approve or Defer Release	C	I	R	I	I	I
Update Deployment Work Item and Release Calendar			R	I	C	C
Implement Release						
Deploy Release	I		I	I	I	R
Report exceptions to the Release			I	C	I	R
Provide support (if needed)			R	C	R	I
Update Deployment metrics for KPI reporting			C			R
Report status of Release and update Work Item			I	I	I	R
Post Implementation Review						
Write up PIR			R	C		C
Review PIR	I	I	R	C		A
Assign action items			R	C		C
Close Release			R	I	I	C

Quality Gates

Release Management Policy will include quality gates designed to provide effective controls and discipline to support a smooth Release Management process implementation. Increased productivity will be realized by reducing the amount of deployment issues and associated troubleshooting which occurs during failed deployments. The following quality gates will be implemented.

- **Release readiness checklist/System verification review** - This will aid in reducing issues attributed to sloppy work when the instructions in the deployment documentation are not 100% accurate.

- **Post-implementation review** - This will identify lessons learned from post release reviews to support continuous improvement.

Release Readiness Checklist/System Verification Review (TEMPLATE)

The purpose of the Release Readiness Checklist/System Verification Review is to ensure that the pending release has followed the approved Software Development and Release Management processes, and that the project team has identified any system interdependencies and risks that may have an adverse impact on the software application/system deployment. It is expected that this checklist will be included as part of the deployment request document.

Approvals

Release Manager, Quality Assurance, Release Engineers

Quality Gate: Release Readiness Checklist				
Project/Application Name	Project Number		Date Created	
<List the name of the project/application. >	<List the Project Number). >		< List the date. >	
Project Manager	Release Manager			
<List the project manager. >	<List the release manager. >			
Description				
<Provide a brief description of the project. Identify both what will be released and the schedule for the release. Include the release procedures. >				
Type of Release				
<input type="checkbox"/> Staging Initial Release <input type="checkbox"/> Production Initial Release <input type="checkbox"/> Staging Re-Release <input type="checkbox"/> Production Re-Release <input type="checkbox"/> Other <Provide explanation if "Other" is selected.>				
Software/System Interdependencies	Yes	No	N/A	Comments
1. Have all systems, software applications, data sources, files, etc. that may be affected by the installation of the release been identified?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
2. Has the potential impact to the affected system been communicated to all business and operations teams?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
Risk Mitigation and Contingency	Yes	No	N/A	Comments
3. Have any deployment risks to the organization, systems, or other related entities that may develop as part of the release been identified?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>

Quality Gate: Release Readiness Checklist (continued)				
4. Is a mitigation risk strategy or contingency plan for reducing or eliminating the risk in place? This includes a roll-back strategy.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
Software Configuration Management	Yes	No	N/A	Comments
5. Have all required components been placed under control during development? (All files, including Executables, libraries and stored procedure files been placed under control.)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
6. Verify correct version of software has been tested in QA environment and all code changes frozen?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
7. Notifications: Have all impacted parties been notified (in addition to the deployment team) about the Release Plan and Deployment window?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Write in the names of those who have been notified>
8. Have all database and IS changes been identified? Any additional tasks or deployment steps been documented?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
Testing	Yes	No	N/A	Comments
9. Has all formal testing been completed including unit, integration, and system tests? Have QA tests been completed?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
Deployment Documentation	Yes	No	N/A	Comments
10. Has all development and release engineers properly filled in their sections in the deployment document?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
11. Have all the deployment steps been validated in the QA environment for accuracy?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
12. Have back out procedures been listed including the latest possible date/time for a go/no-go decision?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>

Release Management – Post Implementation Review (TEMPLATE)

The purpose of the Post Implementation Review is to ensure that any issues encountered during release process have been identified, and the project team has agreed to a course of action for remediation of same-type issues in any future releases.

Quality Gate: Post Implementation Review		
Project/Application Name	Project Number	Date Released
<List the name of the project/application. >	<List the Project Number). >	< List the date. >
Project Manager	Release Manager	
<List the project manager. >	<List the release manager. >	
Description		
<Provide a brief description of the project. Identify both what will be released and the schedule for the release. Include the release procedures. >		
Type of Release		
<input type="checkbox"/> Staging Initial Release <input type="checkbox"/> Production Initial Release <input type="checkbox"/> Staging Re-Release <input type="checkbox"/> Production Re-Release <input type="checkbox"/> Other <Provide explanation if "Other" is selected.>		

Approvals

Release Manager, Deployment Team

Quality Gate: Post-Implementation Review				
Release Management Process Review	Yes	No	N/A	Comments
A. Where all systems, software applications, data sources, files, etc. affected by the installation of the release identified?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
B. Were there any deployment risks to the organization, systems, or other related entities that were not identified?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
C. Did any issues arise that could have led back to the release not tested properly (unit, integration, user test)?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
D. If the release was rolled back, was the backup plan identified properly and did the plan work?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
E. Were the deployment instructions /steps validated for accuracy?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<Comments>
Action Items	Responsible Party		Due Date	Comments
1. <Action Item #1>			MM/DD/YY	<Comments>
2. <Action Item #2>			MM/DD/YY	<Comments>
3. <Action Item #3>			MM/DD/YY	<Comments>
4. <Action Item #4>			MM/DD/YY	<Comments>