

Executive Summary

Release Management Framework: Part II

by Sebastian Konkol, Senior Consultant,
Cutter Consortium

Enterprise Architecture Executive Summary Vol. 12, No. 2

This second part of a two-part *Executive Report* series on release management is intended to offer a more pragmatic view than was presented in Part I¹ by providing strictly practical comments related to the subject of release implementation.

Although the accompanying *Executive Report* focuses on release management from a technology point of view, it provides some insights about the way IT could influence business to ensure benefits for the company. Release management can be thought of as a method of coordinated introduction of various software modifications to the production systems environment.

The report discusses various issues related to organizational structure, process framework, and governance rules that extend and enrich the release management approach. Being aware of their existence and impact makes release management implementation safer and more efficient for the organization.

The logic of the release implementation process should be organized around decision points, whose role should be to improve the quality of modifications implemented under the release. In order to have clean software in the production environment, some gates should be set in the release implementation process, and appropriate rules should be defined to enforce quality on each of these gates. The number of gates and their quality criteria should be designed to fit the specific requirements of the company, but typical gates include requirements stabilized (each modification to be included in the release is described by means of

requirements, which are accepted/signed off on by users); software delivered (each modification's deliverables are provided and integrated in a test environment); and releases validated (each modification has been tested, and the results have been accepted by users). In addition to these typical gates, financial acceptance is added to the release implementation process quite frequently; this gate concerns verification of each modification's business case. Upon reaching each gate, each modification carried out by the release is verified according to quality criteria required to pass the gate; not passing means rejecting a particular modification from the release, as it is potentially risky to the production environment, and usually moving it so it is verified again in the scope of later releases. Each gate adds new quality criteria, so to pass a certain gate, the modification must fulfill the quality criteria set by the gate and all the gates located before the one in the release implementation process.

Release management is focused on introducing modifications to a set of interrelated information systems used for the purpose of serving production and commercial business matters — the production environment, for short. The production environment refers to all the parts of IS that are used for that purpose. Since each release represents a modification to the production environment, it contains the baseline of the production environment and all modifications — the delta of modifications to the baseline. Each release represents a different version of the future production environment and, as such, must be handled separately from other release contents; to validate a release at a certain gate, it must exclusively reside in some special environment — that gate's environment.

The number of gates, which would be defined in the scope of the release implementation process, is correlated with the number of environments that should be available in order to validate the quality criteria of each gate. The idea is to have the release that is being implemented verified and validated at each gate, meaning the required quality criteria should be applied and noncompliances should be identified and resolved. Before getting to the next gate all identified,



problems and bugs should be fixed, so the release is moved to the next gate in a clean state.

There are interrelations between release management processes, especially in the context of release granularity units and architecture management, in terms of both enterprise and systems architecture. Such interrelations should be controlled by means of common artifacts shared between the release management practice and architecture management, as well as governance rules applicable to both release implementation and architecture definition efforts. Deciding on release granularity units affects the architecture at both enterprise and individual system levels.

Running a release implementation process without a deep understanding of the architecture (business, data and application, technical) would be difficult to manage, so the release management team should know the architecture to be able to understand the consequences of the particular way the release is to be implemented as well as its risks and implications. Such knowledge and understanding should be supported with governance rules applicable to the release implementation process and should be included in some gates' quality criteria set. The governance rules should be supported with common artifacts, which should be shared between architecture and release management teams. The set of artifacts should include architecture definition (enterprise architecture blueprints, individual systems architecture designs), approaches to release implementation (initial approach and detailed plans), and fallback procedures planned for each release.

Apart from the release contents, which are managed throughout the release implementation process, the process delivers some tangibles related to the process

planning and execution. Those release management deliverables are created and updated during the whole release implementation process to incorporate the release contents changes and express the consequences of such decisions. The release implementation process could define various deliverables, in compliance with a company culture and requirements stated, but the central tangibles of the process itself are release deployment and rollout approach, release implementation plan, detailed release deployment and rollout instruction, and release notes.

Release management will not work if these fundamental rules are violated and will significantly lose efficiency if they are stretched. If the rule "stretching" required by a particular software development endeavor is too broad, independent modification implementation on the production systems environment could always be an option to consider. It could be executed not in the scope of the release implementation process, but according to rules specially designed for that endeavor and not under the control of the release management practice but based on that endeavor.

Release management has been introduced in many organizations, and various mistakes have been made. Learning from the mistakes of others, we can understand some of the typical pitfalls to avoid and get some hints on how to improve the release management practice in your organization, both of which are addressed in the report.

ENDNOTE

¹Konkol, Sebastian. "Release Management Framework: Part I." Cutter Consortium Enterprise Architecture *Executive Report*, Vol. 11, No. 11, 2008.



About the Service

SENIOR CONSULTANTS

Michael Rosen, Practice Director; Paul Allen; Douglas Barry; Dan Berglove; Udi Dahan; Max Dolgicer; Pierfranco Ferronato; Clive Finkelstein; Michael Guttman; David Hay; Tushar K. Hazra; J. Bradford Kain; Bartosz Kiepuszewski; Sebastian Konkol; Jean Pierre LeJacq; Arun K. Majumdar; Jason Matthews; Terry Merriman; James Odell; Ken Orr; Wojciech Ozimek; Jorge V.A. Ronchese; Oliver Sims; Borys Stokalski; John Tibbetts; Sandy Tyndale-Biscoe; William Ulrich; Mitchell Ummel; Jeroen van Tyn; Jim Watson; Tom Welsh; Bryan Wood

Cutter Consortium Enterprise Architecture Advisory Service

FOR MORE INFORMATION

For more information on Cutter Consortium's Enterprise Architecture Advisory Service or its other services, contact: Tel: +1 781 648 8700; Fax: +1 781 648 8707; E-mail: sales@cutter.com.

The *Executive Summary* is a supplement to the Enterprise Architecture Advisory Service's *Executive Report*. ©2009 Cutter Consortium. All rights reserved. ISSN: 1530-3462. Unauthorized reproduction in any form, including photocopying, faxing, and image scanning, is against the law. Reprints make an excellent training tool. For information about reprints and/or back issues of Cutter Consortium publications, call +1 781 648 8700 or e-mail service@cutter.com.