GESELLSCHAFT FÜR INFORMATIK E.V.
Zukunft gestalten.

Challenges across two worlds: The concerns of IT Change Managers to deal with SAP WebAS Java and WebAS ABAP

Matthias - Claudius Kneissl (Galileo Group AG)

University of Passau
Information Management
matthias.kneissl@galileo-group.de

Type of work: Academic Research Partnership with Galileo Group AG

Abstract: Since release 6.40 the SAP Netweaver platform consists of two different stacks, WebAS ABAP and WebAS Java. Although both stacks can be installed within one system, the technology and development processes are completely different. This article discusses the differences regarding IT Change Management of ABAP and Java and outlines potential problems, which have to be solved to ensure an ITIL compliant change management process for software factories.

1 Introduction

With the release of the Netweaver platform, SAP changed its strategy and offers parallel an ABAP and Java stack. Both technologies are platform independent. While the development process of ABAP is centred to the mainframe architecture, Java has a classical client - server development process. In a typical enterprise IT scenario the WebAS Java is used for web based portal technologies, whereas the master data and the business logic reside in the ABAP stack. Other possible and common examples for the usage of both stacks are Business Intelligence or the SAP Process Integration engine (PI), formerly known as exchange infrastructure (XI). Due to the current development and plans of several companies to migrate their J2EE developments on the WebAS Java, it is expected, that the importance of this technology and diffusion rate will increase.

The close linkage between WebAS ABAP and WebAS Java and their usage scenarios often enforce changes, which have to be performed in more than one system. The proper management of changes in different technologies, the linkage of them and their timely delivery gets more difficult with a complex system landscape and various system technologies. In enterprise IT environments, especially the differences of the change management process in WebAS Java and WebAS ABAP come up and have to be solved to ensure a streamlined IT software factory management process.

2 Organizational Differences

Regarding the scenario of a software factory and an ITIL compliant configuration and change management process, functional change requests have to be approved by the Change Advisory Board and have to get a confirmed release date for productive setting. The technology specific change requests inherit the release date. In case of different technologies, usually the responsible persons for the changes are different. One of the most important ITIL principles is, that no change should be performed without a proper assignment.

In the WebAS ABAP stack, this way of handling change requests is not a problem. A functional change request, which is technology independent, is linked to one ore more SAP ABAP change requests. Those requests can be assigned to the particular responsible person. The tasks, which can be created under this change request, may be assigned to the different developers working on the change request. After the release of the different tasks, the technology specific ABAP change request can be released. A change in the coding or a customizing can only performed after a change request and task has been created.

However, the corresponding WebAS Java workflow is totally different. The basic rule, that no change should be performed without a change request assignment is not in place. Software developers can change source codes, create local activities and can bundle those afterwards to a change request. The problem with this workflow is, that activities can be created without an assignment and subordinated to change requests. Those will go through the workflow oriented development cycle as long as the WebAS Java change management service is in place. While in the WebAS ABAP stack, the assignment to a change request can be performed by creating a task for a particular developer, this is not possible in Java. This means, that the management and assignment of change requests can not be done in the change management service of Java, but has to be done outside and somehow linked. It is an organizational challenge to introduce a handling procedure for WebAS Java change requests and their specific linkage to the overall technology unspecific change request.

One of the great challenges will be the definition of a technology independent development process to link the different steps to the actions, which have to be performed on system level. Without the support of a change management tool, which can handle those actions, it is likely to be a quite error-prone process.

3 Technical Differences

3.1 Linkage of change requests to releases and responsible persons

In the ABAP stack, several changes can be bundled into a single change request. This request usually is assigned to the responsible user and can consist of different tasks, which can be assigned to different developers. In the Java stack, it is currently not to pre-assign a change request to a responsible change manager, as the request is created upon releasing one ore more activities. Without proper change management logic, it becomes difficult to streamline the software development cycle.

3.2 Integrated workflow concept in WebAS Java

As long as the WebAS Java change management service is used, sources are managed within tracks. Those contain a workflow as well as the specific sources. The development process is workflow oriented and strictly linear. The problem, which comes up is, that the WebAS ABAP development process does not have a built-in workflow. While the delivery process in Java is managed by a workflow, the ABAP delivery process is normally managed by adding transport requests to the import queue of the particular system. The proper and correct definition of a workflow on the ABAP side, which is adequate to the Java track based approach, is a challenge for IT managers, which has to be solved organizationally. Furthermore, the specific functional environments of the WebAS Java have to be mapped to the WebAS ABAP functional environments.

3.3 Differences in the functional environments

For the standard ABAP driven development procedure, three functional environments are usually used (development, quality and productive system). Within the Java development workflow, four functional environments are provided. In case of this standard scenario, either the ABAP development process has to be extended by a functional environment, or the development process in ABAP has to be adapted. The track concept in WebAS Java can only deliver one system per step. In case of a more complex system landscape with more than one productive system, several tracks with the integrated workflow have to be connected properly, to provide an automated delivery of more than one productive system. A challenge for each organisation will be the adoption of the ABAP development process to this workflow and the execution of the specific steps within a small timeframe to avoid inconsistencies within each technology.

3.4 Different workflow concept for Process Integration

The management of XI change requests can also be handled by the SAP track construct. A big challenge is the fact, that in contrast to the Java typical four system landscape, process integration development workflows are slightly adapted and provide a three system landscape. Changes in XI often lead to change requests in WebAS ABAP and WebAS Java. While for the Java part a workflow is provided, this is not the case for the ABAP part. This limitation is a big challenge, to manually align the steps in the ABAP part to the Java part. A possibility to bundle technology specific change requests to a functional change request and a streamlined management of the functional change would be very important at this point.

4 Conclusion and Future Work

The comparison of the two different technologies shows, that IT change managers have to handle a high challenge to fulfil ITIL compliance and a proper configuration and change management if both stacks are used and are linked to each other. This scenario is likely to be quite common, as SAP offers more and more web based applications like employee self services or an enterprise portal. The linkage of three different SAP specific workflows (i.e. ABAP, Java, PI) and maybe also another development workflow for legacy systems having is a great challenge. The next step will be the definition of a technology independent process with several scenarios, mapping the technology dependent workflow to the technology independent workflow.

References

[EEE05] Hochstein, Axel/Zanrekow, Rüdiger/Brenner, Walter: ITIL as Common Practice Reference Model for IT Service Management: Formal Assessment and Implications for Practice, Proceedings of the 2005 IEEE International Conference on E-Technology, E-Commerce and E-Service, Honkong, 29.03.2005, Volume 00, IEEE, 2005.

[ITL00a] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for Service Support, 1. edition, TSO Publishing, United Kingdom (UK), 2000.

[ITL00b] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for Service Delivery, 1. edition, TSO Publishing, United Kingdom (UK), 2000.

[ITL00c] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for ICT Infrastructure Management, 1. edition, TSO Publishing, United Kingdom (UK), 2000.

[ITL00d] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for Application Management, 1. edition, TSO Publishing, United Kingdom (UK), 2000.

[KN05] Kösegi, A.; Nerding R.; SAP Änderungs und Traportmanagement, Galileo Press, Bonn 2006

[KTB05] Kessler, K.; Tillert, P.; Dobrikov, P.: Java Programmierung mit dem SAP Web Application Server, Galileo Press, Bonn, 2006

[StOr05] Stumpe, S; Orb, J.: SAP Exchange Infrastructure, 1.Auflage, Galileo Press, Bonn, 2005