

Workflow for SAP NetWeaver MDM – How, Why and What to Choose

Applies to:

The article applies to SAP NetWeaver MDM SP4 and SAP CAF SP9. SAP Business Workflow functionality which is mentioned in this article is available in any NW04 version (and possibly even earlier).

Summary

The article describes three architectural alternatives for realizing a workflow on top of SAP NetWeaver MDM: Guided Procedures, SAP Business Workflow and MDM Workflow. Some explanations about these alternatives are given and a few hints aiming to help opting for the right decision.

Author(s): Amit Levavi

Company: SAP, SSM Solution Office

Created on: 5 November 2006

Author Bio



Amit is a senior solution architect with over eight years of IT experience as an SAP customer, an SAP partner, and for the past three years, an SAP employee. Amit is currently working for SAP SSM Solution Office where he is involved with some of SAP's largest customers. He is based out of Leiden, the Netherlands.

Table of Contents

Why workflow for SAP NetWeaver MDM?	3
Related IT Scenarios.....	3
Central Master-Data Management	3
Master-Data Harmonization	3
Service Enablement	3
Examples of MDM-related workflows.....	4
Employee Details – Data Entry Verification.....	4
Product Data – Consolidation Verification.....	4
Influences on the data model.....	4
Status Handling	4
Versioning	5
What are the alternatives?	5
SAP NetWeaver MDM Workflow	5
CAF – Guided Procedures	6
Architecture	6
SAP Business Workflow.....	6
Architecture	7
What’s right for me?.....	8
Development Skillset.....	8
End-User Work Mode.....	8
Routing Complexity	9
Landscape Complexity	9
Integration with other systems	9
Misc. Features & Functions.....	10
To sum it up	11
Related Content.....	11
Copyright.....	12

Why workflow for SAP NetWeaver MDM?

Upon reading the title you may ask yourself what am I blabbering about and what workflow has to do with SAP NetWeaver MDM; after all, MDM's main "line of business" is data while workflow capabilities (not necessarily a particular workflow tool) are about managing processes. How do these two come together?

Let's consider what type of data is managed by SAP NetWeaver MDM – **Master Data**, meaning products, business partners and employees (and possibly other objects). This kind of data can have an enormous impact on business processes and therefore consistency and reliability are of utmost importance.

For this reason, master data is not simply entered into the system and becomes immediately active but rather, when master data is created, it very often involves review and approval processes. In order to facilitate these review and approval process we should couple MDM with workflow capabilities.

Related IT Scenarios

There are several IT scenarios that should "hint" an MDM-related workflow might be appropriate.

Central Master-Data Management

This scenario enables maintenance of master data within SAP NetWeaver MDM and the distribution of this data to different systems across the landscape.

When implementing central master data management it is almost always imperative to set a review & approval workflow since the data has never been validated in any other system. Moreover, incorrect data may spread across the landscape and could multiply the damage caused by data inaccuracy.

Master-Data Harmonization

This scenario describes an MDM usage where data is received from backend applications, consolidated (meaning checked for duplicates and other cleansing functionalities) and distributed to different systems.

In this case it is imperative to consider two things:

1. The reliability level of the data coming from backend systems – If the data is coming from an unreliable source it may be important to validate it when persisted into the MDM before it is distributed throughout the landscape.
2. The reliability of the complete set of merged data in MDM – Even if the data received from backend systems is reliable, it is sometimes the case when it arrives in several "chunks", and the role of the MDM is to merge these chunks into a unified data model which is then distributed as one entity (examples of this scenario are given later on). In that case it might be beneficial to provide a review and approval process which allows the reviewers to examine the entire set of data as a unified entity ("In Context") and approve the whole rather than each part independently.

Service Enablement

As a building block of the enterprise SOA paradigm, SAP NetWeaver MDM facilitates business processes by providing master data related services to composite applications. An example of such service enablement could be providing product data to an online shop application, or even triggering a business process as in a scenario of a new employee recruited to the organization, persisted in MDM, a step which triggers other processes in HR and possibly other administration steps.

Despite the fact that the business processes and composite applications mentioned above do require workflow capabilities to manage the process flow and do make use of master data, the workflow itself is not "master data centric". In such case, from an architectural point of view, the SAP NetWeaver MDM should only provide services (and launch events) to facilitate the business process, but should not "own" and manage the workflow itself.

Therefore, these kinds of workflows are beyond the scope of this article.

Examples of MDM-related workflows

Employee Details – Data Entry Verification

Consider an application that provides the functionality of creating and maintaining employee details upon their recruitment to the organization.

While the end-user actually performing the data entry could be either an HR clerk or the direct manager, it is most often the case where both have to review the details and to make sure they adhere to the signed contract or company procedures.

In addition, depending on various attributes, for example, the gross salary, there could be other required approval steps such as a department manager's approval.

Notice that despite this being a very simple scenario, it still requires the workflow system to be able to determine value for "logical notions" such as "direct manager" and "department manager", and also to support conditional steps (based on salary value).

Product Data – Consolidation Verification

Consider a solution which consolidates product data from various backend systems into a unified repository within MDM.

Such product data could come from R/3 systems (logistic data), CRM systems (pricing data), planning systems such as SAP APO (forecast data) and also rich marketing content (such as product images).

While the different types of data may be created independently, it may be the case that syndication of data includes several types of data coupled together. For example, a re-seller who operates an online store and manages inventory requires both marketing data (to present on the web) and logistic data (to prepare the warehouse and reserve shelf space).

MDM possesses advanced capabilities to automatically match data coming from different systems according to various matching rules (not necessarily according to product unique ID). However, in some cases, especially when the matching "reliability level" determined by MDM is not 100%, it might be a good idea to validate the consolidated set of data through a review & approval process before syndicating it further.

Influences on the data model

If a review & approval workflow is implemented to manage MDM objects, it usually has direct influence on the data model in two aspects – Status handling and Versioning.

Status Handling

Status handling refers to managing the approval status of the objects in the MDM repository. For example, when an object is first created via the data entry application it might have "Draft" status. Then the creator might submit the object for approval which will change the status to "In review". Finally, after approval, the status might change to "Approved".

Naturally, the status chain above is only an example and could (and should) vary between organizations and objects. However, in any case it is important to maintain a field which manages the status of relevant objects.

Keep in mind that objects different from the ones in the main table might also undergo an approval flow and therefore would also require a status field. For example, consider a product category tree which is implemented as a hierarchical lookup table for products (the main table). Clearly, adding a product category is a cardinal event that might also require an approval flow. In that case, the lookup table should also contain a status field.

Another pending question is what type of field is the status field. The most "intuitive" choice is to make the status field a lookup field pointing to a status lookup table that holds all possible status values. However, keep in mind that in case the MDM component itself is not the one which manages the workflow (see alternatives below), that means that there is another component which is responsible for changing the status and that component also needs to have a list of all possible status values. In that case there should either be

an automatic provisioning process between the components or you should refrain from implementing the MDM field as a lookup and simply implement it as a string field.

In any case, the rule of thumb is to have only one place where you need to implement a change in case a new status value is added.

Versioning

In some cases, business processes require a versioning history of certain objects; a matter which has very little to do with MDM-related workflows. However, even when versioning history is not required, when an existing object is changed and that change is subject to an approval process there are at least two versions of the object available in the system:

- The “Active” version – This version is the last approved version of the object and is to be used in all process until a new version is approved.
- The “Inactive” version – This version represents the object undergoing review and approval.

When the approval process is completed, the “Inactive” version replaces the “Active” version (so there is only one version in the system with status “Active”).

One way to implement the active/inactive notion is to use MDM’s functionality of “Check In/Check Out”. Using this functionality, main table records can be checked out (practically an “Inactive” version is then created), changed and finally checked in thus overriding the former “Active” version.

The “Check In/Check Out” provides exactly the process which is described in this section so it is a very appealing candidate by which to implement version control. However, note that this functionality is only available for main table records. If a review and approval process is required also for objects which are not represented by the main table (as discussed before) then it is impossible to use the “Check In/Check Out” functionality for these objects.

Another implementation alternative is to simply maintain two records in the main table – one for the “Active” version and another for the “Inactive” version. One thing to keep in mind is that the ID of the object is no longer unique as both the “Active” record and the “Inactive” record refer to the same object ID. Another thing to consider (and develop) is a mechanism that replaces the “Active” version when the approval process is done.

What are the alternatives?

There are three architectural alternatives for realizing data-centric workflow scenarios with SAP NetWeaver MDM.

SAP NetWeaver MDM Workflow

MDM Workflow is part of the standard functionality of SAP NetWeaver MDM and you might say it is simply a “module” within MDM.

The design-time environment of the MDM workflow is based on a Microsoft Visio plug-in, which facilitates modeling the workflow’s flow logic. The run-time environment is realized as a “Workflow” tab within the Data Manager client of MDM. No additional installation is required (on top of the standard MDM installation) to enable working with MDM workflow and the interaction is done solely through the MDM Data Manager client application.

The modeling environment of MDM workflow provides a limited number of activity types (e.g. Approve, Validate, Notify) which are orchestrated into a workflow. Each step is configured by a predefined set of parameters which control the behavior of the workflow step (e.g. timeout duration for an approval step). However, there is no way to extend the predefined step types or to develop a custom step.

The routing mechanism is based on the users and roles which are defined for the MDM repository. The usage of roles enables decoupling named users from the process and also facilitates multi-user steps. However, it is impossible to utilize any “organizational logic” (e.g. an approval by one’s manager) or to leverage existing user stores (such as a corporate LDAP).

CAF – Guided Procedures

“Guided procedures” (GP) is the process layer of SAP Composite Application Framework (CAF). Guided procedures enable orchestrating platform-independent and backend independent activities into a cohesive business process while providing an easy and intuitive UI.

The modeling environment of GP is accessed through SAP NetWeaver Portal, and it provides two main functionalities. The first functionality is constructing “callable objects” by either “wrapping” existing backend functionalities (like Remote Function Modules in ERP systems) or creating new functionalities (e.g. Adobe Interactive Forms).

The second modeling functionality is the orchestration functionality, where different callable objects are composed into a business process.

One of the advantages of GP is its openness, which enables constructing callable objects from various backend systems and technologies. Despite the fact that there is no standard integration with MDM, it is possible (and not very difficult) to integrate MDM and GP – All you need to do is expose MDM functionality as web services and then use web service callable objects to invoke the functionality.

However, one of the current problems of GP is the lack of a graphical modeling tool which would enable modeling complex workflows. Therefore, GP is currently aiming at simple “departmental” workflows with a few steps and a simple routing logic.

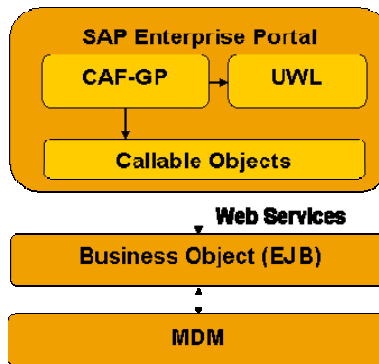
Another functionality which is not yet fully developed in GP is its routing capabilities and its integration to HR systems. It is already possible to determine one’s manager through connectivity to SAP HR, but any other rule must be coded and consumed through technical means (web services etc.).

Architecture

Connecting GP to MDM requires creating callable objects that expose MDM functionality such as “retrieve product data”, “change product status” and so on. MDM does expose its functionality as web services, however, in order to remain generic, the granularity of these services is very high and it could be difficult to use them directly as callable objects.

Therefore, when using GP and MDM, it is recommended to create an intermediate layer (possibly using EJBs or CAF Objects) which consume the low-level services of MDM and compose them to higher level services such as the ones mentioned above.

In addition, it is very likely that in order to realize complex routing logics additional “auxiliary” services will be required (and defined as callable objects).



MDM Integration with CAF Guided Procedures

SAP Business Workflow

SAP Business Workflow has been part of the ABAP infrastructure for well over 10 years. It is now part of any ABAP stack and can be installed either as a stand-alone ABAP WAS or can be leveraged from an existing ABAP-based system such as mySAP ERP, mySAP CRM etc.

SAP Business Workflow provides a graphical modeling tool called Workflow Builder that helps visualize the workflow logic as a flow diagram. Using the Workflow Builder, the workflow designer can orchestrate various activity types, from flow control operations to user decisions and other actionable activities.

The objects on which SAP Business Workflow operates are either BOR (Business Object Repository) objects or ABAP Objects. In order to integrate MDM with SAP Business Workflow it is required to model the MDM object as one of these two alternatives (BOR or ABAP Object). This requirement imposes a minimal effort barrier for using MDM in conjunction with Business Workflow, but the effort is re-usable and can be leveraged again with any additional workflow.

Despite its origin in ABAP systems, SAP Business Workflow can be exposed quite easily in the portal environment using UWL (Universal Worklist) which is a generic portal inbox that can collect work items from various sources.

An important part of the maturity of SAP Business Workflow is manifested by its extensive monitoring and troubleshooting tools, as well as by its integration to the standard version and transport management of the ABAP stack.

When implementing a review and approval scenario, using MDM, SAP Business Workflow and SAP NetWeaver Portal, it is often required to display the object under review and to provide the standard functionality of approve/reject. This can be easily done by configuring UWL to react to “MDM work items” by opening a portal page which has two iViews – One iView is the standard UWL item processing iView which provides the necessary user actions. The second iView should be an iView that displays the data coming from MDM. This could be either one of the pre-delivered iViews in the MDM Business Package or a custom iView.

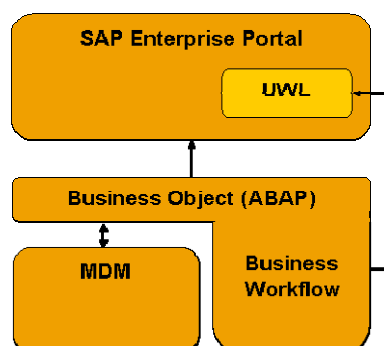
This way you leverage the extensive processing capabilities of UWL and the business workflow (such as ad-hoc workflow, attaching memos to the work items and more) while providing a view of the object at hand.

Architecture

SAP Business Workflow works with either BOR Objects or ABAP objects that exists in its ABAP environment. Unfortunately, there are no pre-delivered BOR/ABAP objects that represent MDM records.

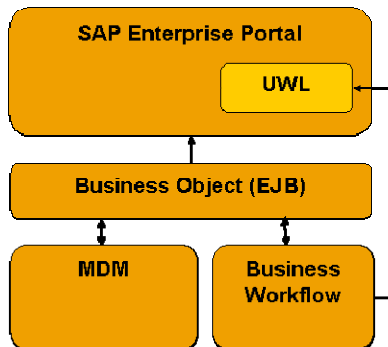
Therefore, a pre-requisite for connecting SAP Business Workflow to MDM is to build a BOR Object which represents the products in MDM. But how does this BOR Object really communicate with MDM? There are two possibilities to do that:

1. Using MDM ABAP API – Creating an ABAP code which communicates with MDM via MDM ABAP API (available as of MDM 5.5 SP3). Using the ABAP API directly from the Workflow environment simplifies integration between SAP Business Workflow and MDM, but makes integration to other systems more difficult as ABAP is not the greatest environment with regards to openness.



MDM Integration with Business Workflow – Using ABAP APIs

- Using MDM Java API – That form of integration requires constructing an EJB (or CAF object) in some J2EE environment which communicates with both the MDM and the workflow environment. For example, a method such as “Create Product” would create a record in MDM and would, in parallel, launch an approval workflow.



MDM Integration with Business Workflow – Using Java APIs

Regardless of which API you choose, the Universal Worklist in the portal should be configured to expose the SAP Business Workflow within the portal.

What’s right for me?

Naturally, there is no clear cut answer to when you should opt for which tool, but I will try to outline a few things you should take into account when making this decision.

Development Skillset

If your organization is an ABAP shop, and maybe you even already possess some SAP Business Workflow expertise, then integrating MDM with SAP Business Workflow should be no problem. This goes double if you’re using SAP NetWeaver MDM 5.5 SP3 where you can use the new MDM ABAP API and implement the business object layer also in ABAP. Otherwise keep in mind that you’ll have to utilize some Java expertise to implement the business object layer.

Guided procedures, on the other hand, is based on Java and web technologies. Most likely, missing functionalities will be implemented in Java and possibly with some advanced tools and frameworks provided by SAP such as Adobe Interactive Forms and Web Dynpro. Therefore, if you’re a Java shop, you might find this tool easier to handle than an ABAP-based tool like SAP Business Workflow.

The standard MDM workflow provides no opportunity to compliment its functionality with any kind of coding. Too bad for you hackers! The only skills you need when using MDM Workflow are MDM skills (which you need in any case).

End-User Work Mode

When considering the tools you provide users for executing their workflow-related activities, you should also take into account the tools the users use for other activities.

Some users work mainly within the MDM client (e.g. the MDM Data Manager, MDM Console etc.). This is often the case with data-centric roles and teams such as a “Data manager” role or a “Data Quality Validation team”.

In such a case the MDM workflow enables the users to keep working within this client while performing their workflow related tasks. This could prove very beneficial for easier adoption and a “smoother” roll-out.

On the other hand, it is often the case that the users involved in the review and approval process are not using the MDM client for any other activity. In that case, using the MDM client for mere workflow-related purposes might be an “overkill” that would hinder adoption. Keep in mind that the MDM client is a tool intended for professional users – It is not very technical but it is extremely rich with functionality and therefore difficult to manage on an occasional basis.

In such a case it is advisable to construct a “friendlier” UI using SAP NetWeaver Portal. Both SAP Business Workflow and GP are easily integrated to the portal through the universal worklist (UWL).

Having said that, starting with SAP NetWeaver MDM SP4, MDM workflow will also begin its journey towards integration with the portal and UWL. Don't expect too much from this release, but it's certainly going in that direction.

In case some of the users work with the MDM client while other are occasional users who fit better the portal way-of-working, it is worthwhile to consider a hybrid option – creating a web application that handles the workflow interaction (e.g. a Web Dynpro application) and exposing it to some users through the portal and to others through the Web Tab in the MDM client. That way the same application is leveraged in two different work concepts, according to the users' requirements.

Routing Complexity

The term “Routing Complexity” refers to the rules and logic that determines who are the users that are involved in each step of the process.

In some cases this logic is simple and static – For example, user A always enters data into the system and user B is always the one to approve data. In other cases the logic might be more complex – any user may enter data. The direct manager of the user who entered the data is the one whose approval is required.

Additional possibilities for complex routing logic include:

- Routing based on data attributes – Consider a scenario where a new product is entered into the system and the approver of the data is determined according to the product's group.
- Conditional steps – Approval of the department manager is only required if the product's value exceeds a certain threshold.

As a rule, MDM workflow facilitates only basic routing logic (through the combination of branching steps and validations), while GP and SAP Business Workflow can (sometimes through coding) facilitate any routing logic. To differentiate between the two I would say that SAP Business Workflow provides more “code-free” mechanisms than GP, to implement routing logic.

That means that GP will probably require more code (and more maintenance) than SAP business workflow. Therefore you should consider the stability of the routing logic when opting for the right tool.

Landscape Complexity

As mentioned above, MDM workflow is simply a module within MDM. Therefore, to utilize MDM workflow the only component needed in the landscape is an MDM server. Due to the fact that this article discusses MDM-related workflows, I would assume that is a given.

The GP alternative requires (in addition to the MDM server) an NW04s Java stack with a portal installed. The portal provides both the design time environment where GP modeling is performed, and the user interaction tool.

The SAP Business Workflow requires an ABAP stack with web flow capabilities (anything higher than WAS 6.1), an SAP NetWeaver Portal (but not necessarily an NW04s portal. Anything later than EP6 SP2 will do) and possibly a Java stack to implement the business logic layer (the portal's java stack can be used though it is not recommended to use the same machine for both a portal's and custom developed applications).

Please note that even though the SAP Business Workflow alternative seems the most complex, it offers many possibilities to leverage existing systems – The workflow could be implemented in any ABAP based system such as BW, mySAP ERP etc. and the portal could (and should) be an existing portal (especially since there is no need to use the latest releases for the functionality introduced in this article).

Integration with other systems

It might be the case that approval workflows will require integration to various systems (other than MDM) to provide context for the approval flow. For example, consider a product approval workflow where a product group manager is required to approve the entry of a new product. It is a reasonable requirement to enable

the product group manager to browse through existing systems (e.g. the ERP system) to look for similar products before approving the new entry.

The different alternatives that are presented in this article differ in their integration capabilities. When deciding which workflow alternative to choose, you should consider the number and type of systems that take part in the process.

MDM workflow is almost a “closed” solution with very little integration possibilities. The only possibility to integrate an external application to the MDM client is to integrate a web application through the web tab. Even that is limited and there is little support for single sign-on or other essential integration capabilities.

SAP Business Workflow is an ABAP-based systems and therefore probably not the easiest integration platform available. Having said that, consuming web services from ABAP environments has become quite easy during the past releases.

Nevertheless, when exposing SAP Business Workflow through the enterprise portal, it is possible to integrate different systems only on the portal level (UI integration) and thus dismissing the problems of integration within an ABAP system.

The GP environment however, is specifically suited for integrating various sources to one coherent process. GP supports a number of pre-defined source types for integration, such as Web Dynpros, BSP applications, backend RFCs, Web Services, KM objects, BI objects, Portal iViews (inc. visual composer iViews), CAF objects and Adobe Interactive Forms. New types of sources (or callable objects in GP terminology) are introduced with every release. Therefore, the GP environment is probably the best suited environment for cross-system integration.

Misc. Features & Functions

The tasks of workflow definition, monitoring and management could have any number of detailed requirements. While some of these requirements are core to workflow operations, others might be nice-to-have.

As a rule, the more mature the workflow platform is, the more “extras” it provides to workflow users, implementers and administrators. Therefore, SAP Business Workflow, being the most mature workflow platform provides more features than the other two alternatives.

Below is a short list of such additional features which could make your life easier when establishing a workflow:

- Deadline monitoring – Being able to set an upper limit for the time the process “waits” for a response on a certain work items. This feature is essential for maintaining a “clean” workflow repository. Otherwise you might find extremely “veteran” workflows still being kept active and thus invalidating all workflow reporting and analysis.
- User substitution – Users, even those who use workflow, sometime go on vacation, or fall ill or don't turn up for any number of reasons. In order to keep the operations going it is important to be able to handle workflows whose assigned user is absent. This can be done by either a static definition of “user A is covering for user B” or by an ad-hoc intervention of a workflow administrator.
- Concurrent Versioning – Workflows are ever changing as they represent the business processes in the organization, which are ever changing themselves. It is important to be able to change the workflow definition while some workflows are still active. The behavior of the system in that case could be to enable the existing workflows to run with the “old” definition, while new workflows are created according to the new definition.
- Supporting both structured and ad-hoc workflow – In many applications the defined business process is merely the baseline for the actual process, while in “real life” workflows also take some ad-hoc “turns” when some user asks the opinion of another user. Supporting this in the workflow environment could enable better controlling the process without going into “e-mail chaos”.
- Logging and monitoring – Workflows are often analyzed; whether for operational purposes with reports such as “All documents awaiting my approval” or for longer term analysis with reports such as “What is the average time for approving a document? Where is the bottleneck?”. Mature workflow environments should provide these reports, or at least a framework on which reports could be built.

Basically, when choosing the workflow tool, one should also try to evaluate the benefits of these little “extras” and consider whether they are valid differentiators between the different options.

To sum it up

Three workflow environments are available for constructing MDM-based workflows - MDM workflow, Guided Procedures, and SAP Business Workflow.

MDM Workflow offers the least functionality, but on the other hand is easily available within the MDM environment. If the functionality of this tool suffices – use it! It can't get any easier.

The two other environments offer more capabilities while Guided Procedures offer more openness but will require (in its current state) more custom coding. In addition, the modeling environment of Guided Procedures would make it difficult to model and manage complex workflows.

SAP Business Workflow requires some ABAP Knowledge, but provides a mature environment with plenty of features and a very capable modeling tool.

I am sure that if you got as far as this line you are well equipped to make the right decision ☺

Related Content

For some more interesting reading please refer to the following links

- A wonderful [blog](#) by workflow guru, Ginger Gatling, telling you all you need to know (and more?) about workflow tools in SAP.
- A [technical article](#) by Laszlo Szervaczius describing how to integrate MDM Workflow with UWL (brand new MDM SP4 functionality!)
- If you have a lot of spare time – Have a look at the [CAF Tutorial Center](#) which also provides plenty of information about Guided Procedures.
- For SAP business workflow documentation feel free to browse for hours inside help.sap.com.

Copyright

© Copyright 2006 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

Any software coding and/or code lines/strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.