

SAP NetWeaver Process Integration 7.1
Integrate Applications Using the
Web Services Reliable Messaging Protocol



SAP NetWeaver Regional Implementation Group
SAP NetWeaver Product Management
December 2007

THE BEST-RUN BUSINESSES RUN SAP™





After reading this document, you will be able to:

- Understand the importance of WS-RM in information replication
- Recognize the benefits and scope of WS-RM
- Implement a WS-RM based scenario in SAP NetWeaver PI 7.1

Agenda



1. Introduction to WS-RM Protocol
2. WS-RM Implementation
3. Excercise Scenario Description

Agenda

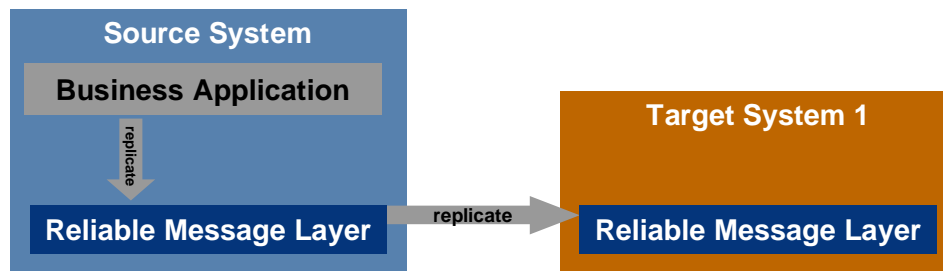


1. **Introduction to WS-RM Protocol**
2. **WS-RM Implementation**
3. **Excercise Scenario Description**

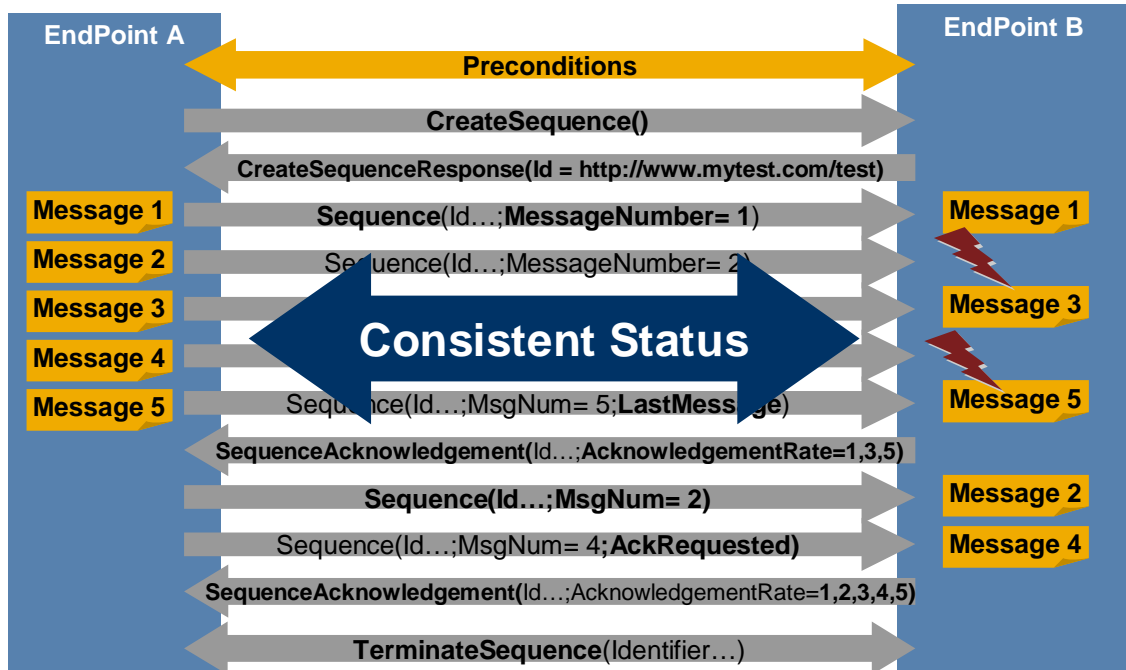
What is Web Services Reliable Messaging?



- **WS-RM describes a protocol that allows messages to be delivered complying a certain quality of service.**
- **Independent of network technologies.**
- **Based on Web Services specification.**
- **The February, 2005 version was submitted to the OASIS Web Services Reliable Exchange (WS-RX) Technical Committee. Voting to approve the resulting WS-Reliable Messaging 1.1 as an OASIS Standard begins on May 16th, 2007.**



- In accordance with OASIS, the purpose of the WS-Reliable Messaging (WS-RM) specification is "... to create a generic and open model for ensuring reliable message delivery for Web Services."
- This specification describes a protocol that allows messages to be delivered reliably between distributed applications in the presence of either software component, system or network failures, and this protocol is described in a "transport independent way" allowing it to be implemented using different network technologies.
- The basic value of this specification is the description of a foundation level support layer for information replication.
- Version 1.1 will be an OASIS standard on May 16, 2007.
- Other vendors like Microsoft and IBM also support WS-Reliable Messaging specification.
- _____
- To support interoperable Web Services, a SOAP binding is defined within this specification.



- The WS-RM protocol uses a "Sequence" header block to track and manage the reliable delivery of messages.
- Each sequence has a "unique identifier element", and each message within a sequence, has a "message number element" that increments by 1 from an initial value of 1. These values are contained within a "sequence header block" accompanying each message being delivered in the context of a sequence.
- In this slide, you can see an example where two endpoints exchange a sequence of messages. The protocol is as follows:
 - First of all, some preconditions are arranged
 - Then, the consumer endpoint A, asks endpoint B (the provider) to create a sequence and endpoint B, answers with a sequence identifier.
 - After that, the consumer endpoint starts to send messages, including the number and the "last message" token when required.
 - Later Endpoint B (the provider side), sends back an acknowledgement with a list of the message numbers it received.
 - If the consumer realizes that the server missed some of the messages, it sends the missing messages again, adding an acknowledgement request at the end.
 - At that moment, the provider side issues the new acknowledgement message.
 - When the consumer finally makes sure that every message reached the server, it sends a "terminate sequence" command.

Agenda

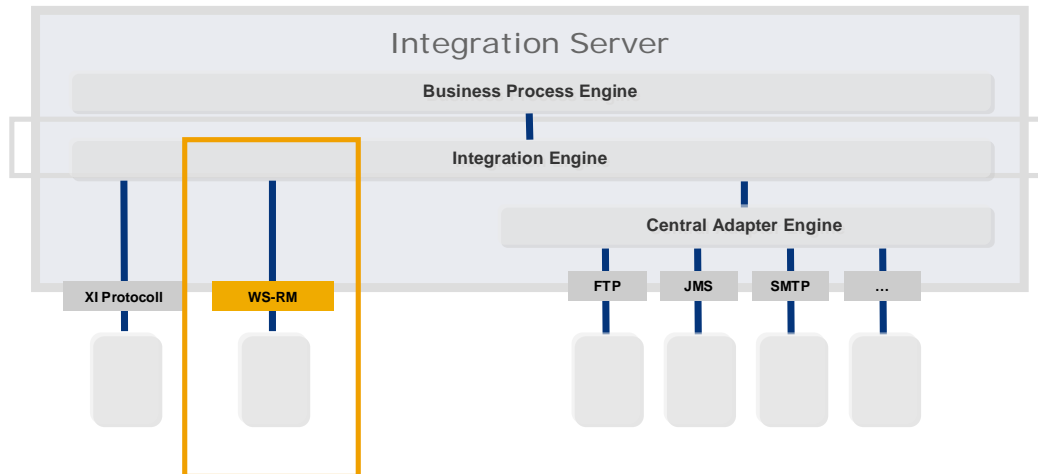


1. Introduction to WS-RM Protocol
2. **WS-RM Implementation**
3. Exercise Scenario Description



WS-Reliable Messaging (WS-RM)

- Asynchronous messaging (EO, EOIO) based on open WS standard
- Native support through Integration Engine (w/o adapter engine)

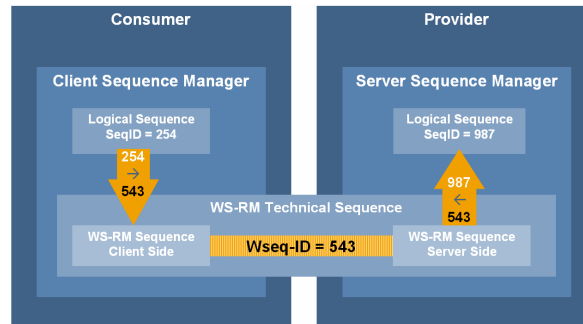


- The PI SOAP adapter as of today supports already asynchronous messaging, however with proprietary means since there was no standard available before.
- The WS-RM protocol is supported via an own entry in the integration engine (implemented in ABAP) and *not the adapter engine*. The XI 3.0 and XI 2.0 protocol will still be supported as well.



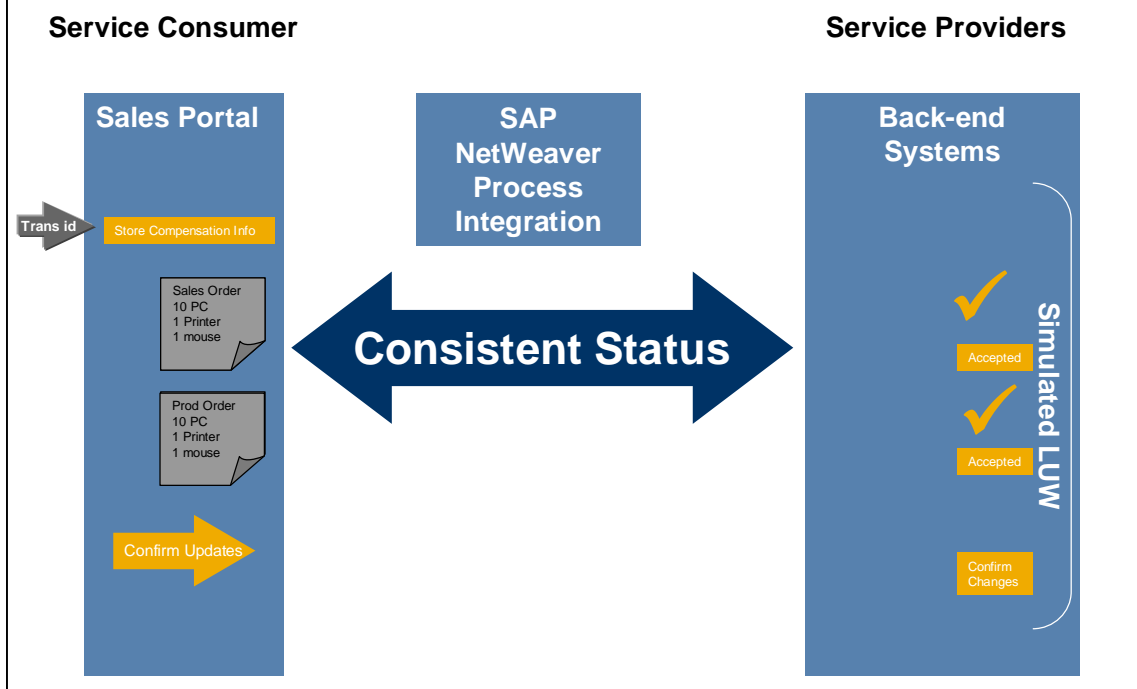
Offer abstraction from the Technical Sequences (WS-RM).

- Application works with a "Logical Sequence" (ID) and WS Runtime maps this to WS-RM Sequences.
- Transactional Behavior.
- On consumer-side either all or none of collected calls are sent to the provider.
- On provider-side each call is executed in a separate transaction.



- Logical Sequence = SOAP Sequence
- Technical Sequence = WS-RM Sequence

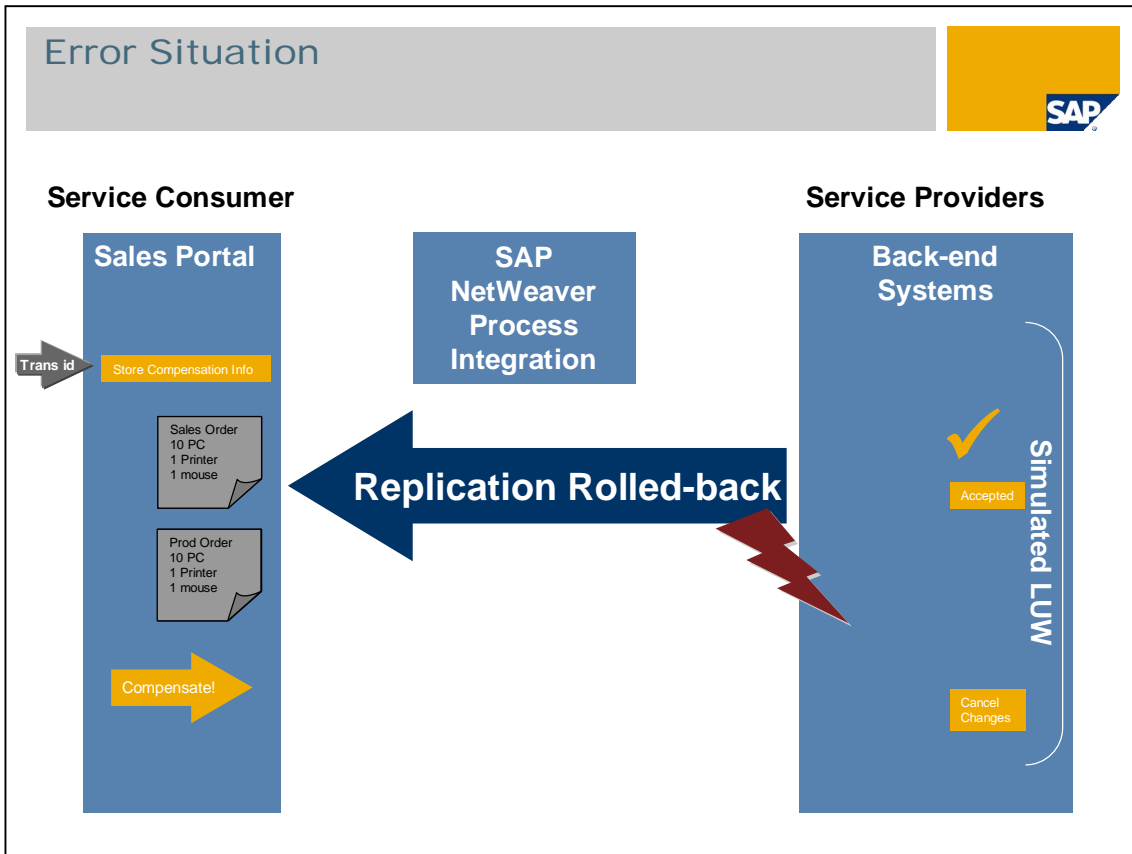
The SAP TU&C/C Replication Pattern



SAP designed a replication pattern based on sequencing.

The pattern is as follows:

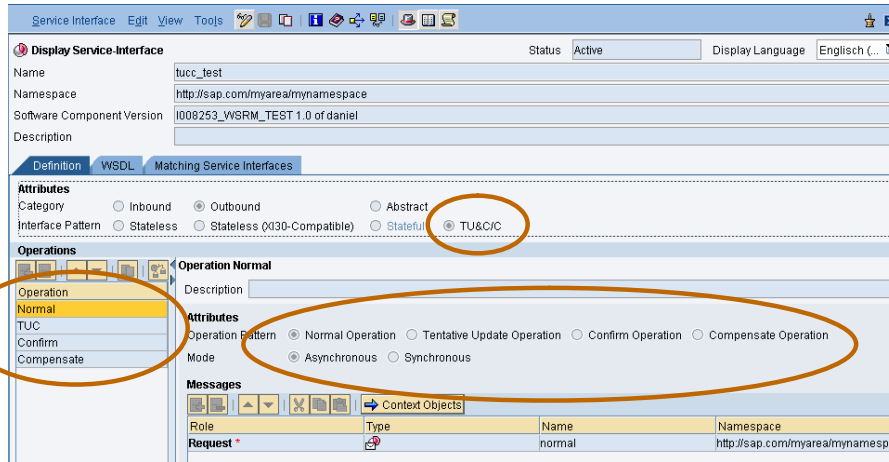
- First, a transaction ID is created on the consumer side
- After that, it is necessary to store message compensation information. It will be automatically used by the kernel if something wrong happens before the sequence is terminated.
- Then we can execute as many synchronous replications or lookups as necessary.
- Once we have successfully finished, we send the confirmation and explicitly issue a commit-work statement.
- The confirmation operation in an asynchronous replication with quality of service Exactly Once In Order, is executed by the kernel after the local commit-work procedure finished.
- The provider sides should at that moment also confirm the changes and delete any transient information used to simulate the logical unit of work.



Let's now replicate the TU&C/C example, to see how it reacts in case of error:

- The processing starts as usual, but something goes wrong when replicating the invoice.
- The consumer program must somehow rollback the work, so as to let the kernel know that the compensation message has to be triggered.
- Since the consumer works in a single logical unit of work, everything is handled by the kernel
- But on the provider sides, since there is no logical unit of work, the changes have to be rolled back manually. That also means that the previously replicated sales order, has to be kept in the meanwhile somehow not available to the application layer. That is, the providers have to implement a logic to validate and keep the information in a transient status until it is finally either confirmed or deleted.
- _____
- SAP NW System >= 7.0 SPs 13 (since WS-RM will be down ported to SAP NW 7.0 SPS 13)

You can design the TU&C/C interfaces in the Enterprise Services Builder



- In the Enterprise Services Builder (ESB) you can create service interfaces with operations inside, also following the TU&C/C pattern. These operations correspond to a logical sequence.
- The new service interface object, that replaces the previous message interface object, is specially designed to be able to model TU&C/C sequences.
- The operations are components of the service interface and they can be:
 - Normal Operation: To validate or retrieve information
 - Tentative Update Operation: Used to replicate information
 - Confirm Operation: Is the one that sends the successful finish message, or
 - Compensate Operation: Always used in the TU&C/C pattern, but rarely used at runtime to inform the providers about replication problems.



The communication channel is configured to use the “Web Service” adapter type.

- automatically the WS-RM (version 2005/02) protocol is selected.

Edit Communication Channel Status: In Process Display Language: English (OL)

Communication Channel: Test
Party:
Communication Component: TestComponent
Description:
Parameters Identifiers Module

Adapter Type * WS http://sap.com/xi/XI/System SAP BASIS 7.10
Sender Receiver
Transport Protocol * HTTP 1.0
Message Protocol * WS 1.0
Adapter Engine * Integration Server
Authentication method User ID/Password (Authentication on HTTP Level)

Technical Transport Settings
Transport Protocol * HTTP
Target Host *
Service Name/Port *
URL Access Path *
Logon Language in Target System
Optimized XML transfer No Optimization
Transport Binding * SOAP 1.1 over HTTP
Server Side Session Timeout (Seconds; 0=System Default) 0

Reliable Messaging Settings
Acknowledgement Interval (Seconds) 600
Protocol WS-RM 2005/02

Settings of Web Service Addressing

- At configuration time, when you are creating your communication channels in the Integration Directory, there is a new type of adapter called “WS”.
- The protocol is automatically set to “WS-RM 2005/02”.

Summary



- **SAP Process Integration and the SAP Web Application general now support the WS-RM open standard protocol.**
- **SAP Process Integration tools allows you to centrally develop and configure the integration scenarios.**
- **The framework provides automatic confirmation or compensation services to guarantee that information is properly replicated.**



- The configuration is automatically propagated to the application systems!
- Application systems have to correctly design the replication process.

Agenda



1. Introduction to WS-RM Protocol
2. WS-RM Implementation
3. **Excercise Scenario Description**

[Exercise Scenario Description] General Description

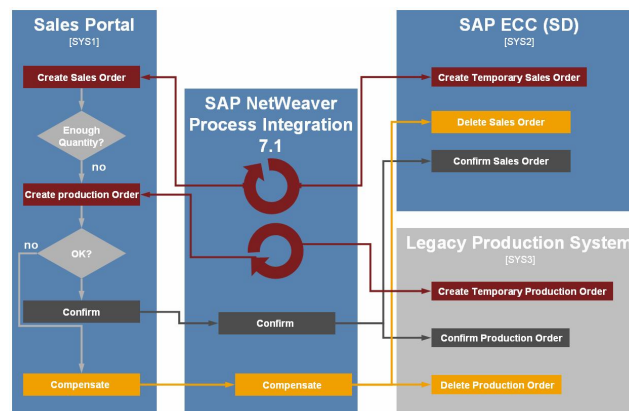


3 “logical” backend systems:

- 1 consumer → Sales system
- 2 providers → SAP ECC SD, Legacy Production System

Scenario:

1. End-user creates a sales order.
2. If the quantity is not available, the program requests a production order.
3. If something goes wrong the compensation message is triggered.



Inside-Out:

- Development either already exists or starts in the application systems
- The configuration is performed also in the application systems

Outside-In (also referred as "Central"):

- Development starts in the Enterprise Services Repository
 - Later the SAP Application Systems are able to access and automatically create proxies for the service interface operations
- The configuration is carried out using the Integration Directory



Steps:

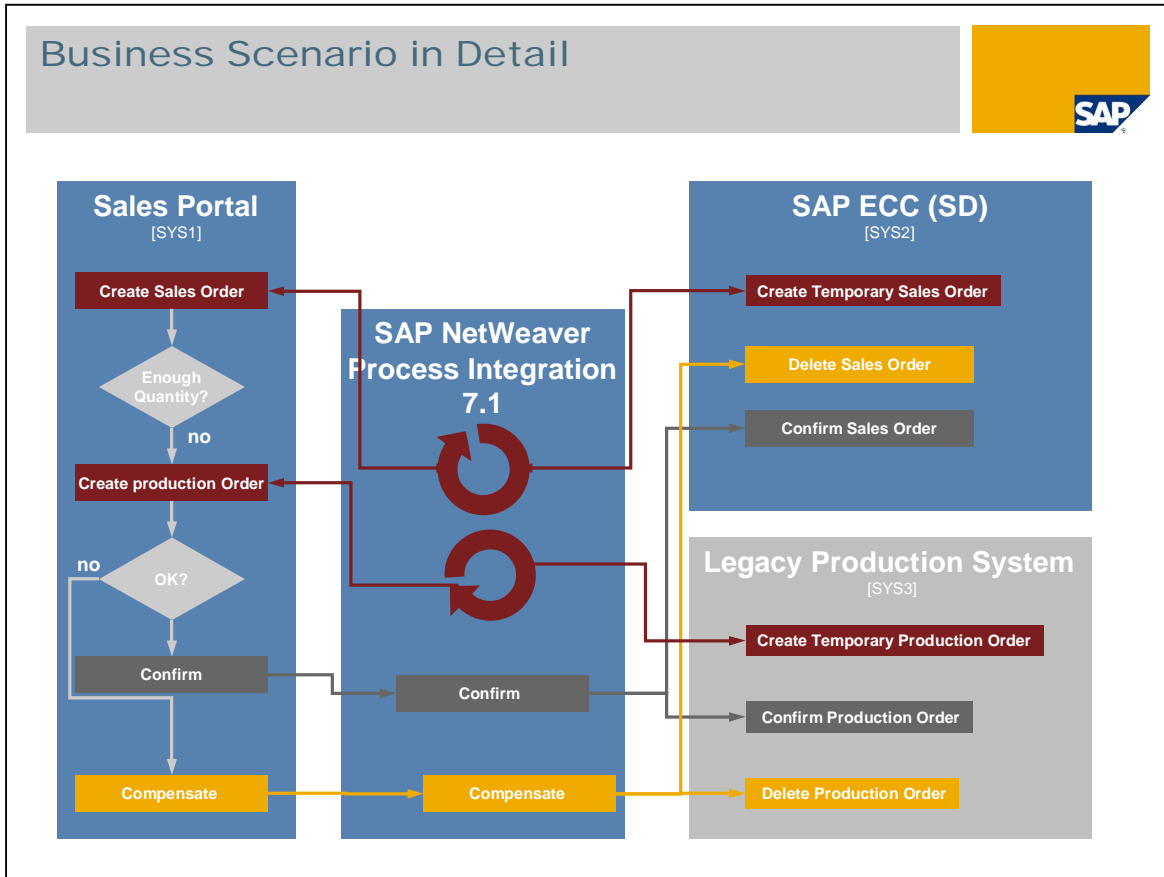
- Create Basic Component is ESR.
- Create Service Interfaces.
- Create the Operations Mappings.
- Perform Configuration in the Integration Directory.
- Create Consumer Proxy.
- Create Consumer Program.
- Create Provider Proxy.
- Implement Provider methods.



■ Other Configuration Steps:

- WS-RM /SOAP configuration
- Configure users authorizations
- Configure WS runtime in application systems.
- Configure RBAM authorization at provider side.

Business Scenario in Detail





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, Excel, 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, System i, System i5, System p, System p5, System x, System z, System z9, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWERS, 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.

The information in this document is proprietary to SAP. No part of this document may be reproduced, copied, or transmitted in any form or for any purpose without the express prior written permission of SAP AG.

This document is a preliminary version and not subject to your license agreement or any other agreement with SAP. This document contains only intended strategies, developments, and functionalities of the SAP® product and is not intended to be binding upon SAP to any particular course of business, product strategy, and/or development. Please note that this document is subject to change and may be changed by SAP at any time without notice.

SAP assumes no responsibility for errors or omissions in this document. SAP does not warrant the accuracy or completeness of the information, text, graphics, links, or other items contained within this material. This document is provided 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 have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials. This limitation shall not apply in cases of intent or gross negligence.

The statutory liability for personal injury and defective products is not affected. 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.