

Integration Framework of SAP Business One

Tips for Optimizing Scenario Package Development

03 2015



Welcome to the training on scenario development for the integration framework of SAP Business One.

This course covers recommended optimizations that can increase the performance of scenario packages that you develop.

Objectives



After completing this course, you will be able to:

- Identify areas in which you can optimize the performance of scenarios

After completing this course you will be able to:

- Identify areas in which you can optimize the performance of scenarios

Introduction

- **General recommendation:** For all scenarios, including scenarios that do not involve SAP Business One, you can increase performance using filters for sender and receiver systems
- **SAP Business One Integration:** For scenarios that interact with SAP Business One, you can optimize performance by using:
 - Queues to avoid long running DI API Transactions
 - Multiple DI Proxy Servers
 - Proxy Groups
 - SAP Business One Event Sender Filters
 - The DI API Single Transaction Option
- **Best practices document:** maintained in SAP Note **2192737**



- For all scenarios, including scenarios that do not involve calls to SAP Business One, consider setting up filters for sender and receiver systems.
- For scenarios that interact with SAP Business One, there are several optimizations you should consider:
 - Queues to avoid long running DI API transactions.
 - Multiple DI proxy servers and groups
 - SAP Business One event sender filters
 - DI API single transaction option

These optimizations are intended to increase the performance of your running integration framework. SAP maintains a best practices document listing these optimizations and other recommendations in SAP Note 2192737.

Let's look at these optimizations in more detail.

General Optimization Recommendation

Setting Sender and Receiver Filter Definitions

- You can define filter criteria for sender and receiver systems
- Using filters, the integration framework performs each scenario step for a specific sender/receiver combination
 1. If empty, always
 2. No, never perform
 3. Only perform, if condition is true

Sender/Receiver systems are defined at scenario package level.

© 2015 SAP SE or an SAP affiliate company. All rights reserved.

4

The final optimization applies to all scenarios, even if they do not interact with SAP Business One.

When setting up the sender and receiver systems of a scenario package, you can define filter criteria. This reduces unnecessary step processing. When you define filters, the integration framework will perform each scenario step for a specific sender or receiver system *only* if the filter condition is true.

The example shows the filter setup for a sender system. You can define similar filters for receiver systems.

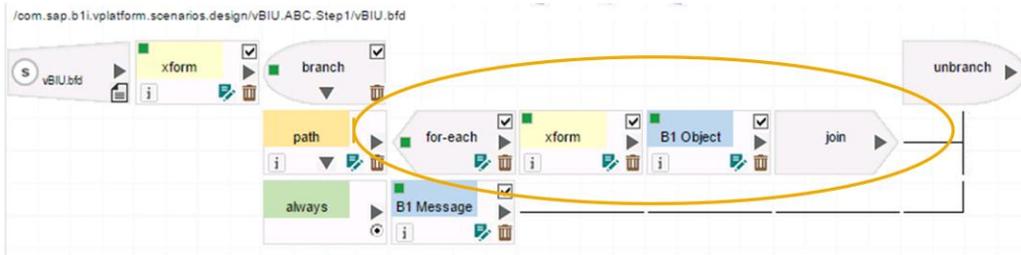
You enter values for each filter condition by choosing *Maintain Sender Filter Definitions* in the setup:

1. If you leave the filter condition empty, the integration framework *always* performs the scenario step.
2. If you enter 'no' as the filter condition, the step is never performed. The sender system is excluded for the scenario step.
3. If you enter an Xpath expression, the step is performed only if the defined condition is met.

SAP Business One Integration

Avoiding Long Running DI API Transactions

- The processing phase of a scenario step is processed as one transaction
- If a call to SAP Business One is embedded in a loop, the DI API transaction might block other user actions
- SAP recommends instead to split the processing into two separate steps connected by internal queues



1 Scenario Step = 1 DI API transaction

The first optimization is the use of queues to avoid long running SAP Business One DI API transactions. These can happen because the processing phase of a scenario step is processed as one transaction. If a call to SAP Business One is embedded within a processing loop, and the call updates SAP Business One objects from within the loop, this can result in a long lasting SAP Business One DI API transaction which might block other user actions in SAP Business One.

The example shows a processing flow where the B1 Object atom is embedded within a branch. Imagine a case where the processing performs a set of calculations and writes the result of each calculation to an SAP Business One object using the B1 Object call atom. The B1 Object atom uses the DI API to call SAP Business One. The DI API transaction could run for a long time while the calculations are processed.

To prevent this, SAP recommends designing the processing as two separate steps connected by the internal queue mechanism of the integration framework. Let's see how internal queues can be used in this example.

SAP Business One Integration

Using Two Separate Steps -1

- By splitting the processing into two steps, DI API actions will be queued in the DI Proxy and executed asynchronously
- In **Step 1** the B1 Object call is deactivated and a Call Scenario Step atom is added to call **Step 2** which processes the B1 Object call



Note: The Call Step atom is available with SAP Business One release 9.1 PL08

For prior releases use the Put to Queue atom

By splitting the processing into two separate steps, you can avoid long SAP Business One DI API transactions. The DI API actions are queued in the DI Proxy and executed asynchronously, therefore there is no performance delay or long transactions.

The original scenario can be redesigned using two steps:

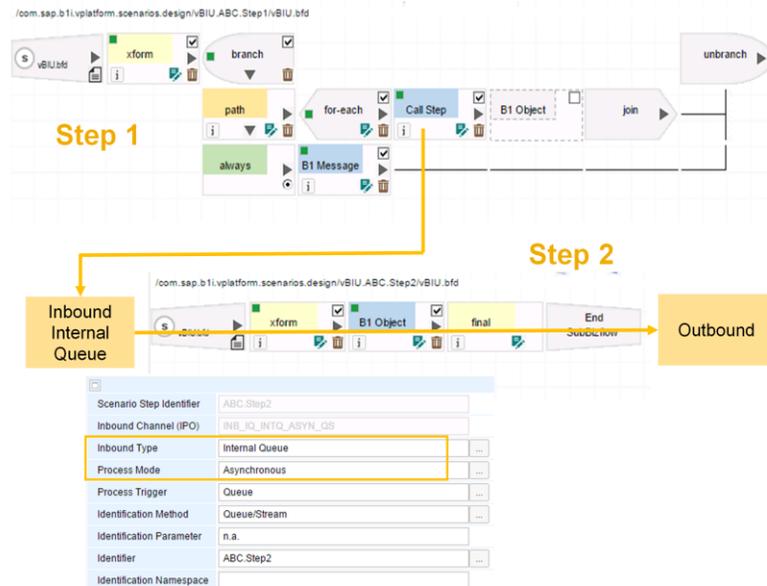
- In step 1, define the loop and include the calculations. But instead of calling SAP Business One directly, use the *Call Scenario Step* atom to call another step in the scenario. This is highlighted in the example. You can also see that the B1 object call is no longer active in this step.
- In step 2, include the call to SAP Business One. This step starts working as soon there is a message available in the internal queue. The step picks up the message including the calculation from the queue and makes the call to SAP Business One to update the objects with the calculation results.

Note: the Call Scenario Step atom is available with SAP Business One release 9.1 patch 08. If you are using an earlier version you can use the Put to Queue atom in this scenario.

SAP Business One Integration

Using Two Separate Steps - 2

- Step 1 calls step 2 asynchronously and hands over the message to the inbound internal queue
- Step 2 is defined with the inbound channel type set to Internal Queue



For more information on queues, see the *Working with Internal Queues* course.

The Call Scenario Step atom in step 1 calls step 2 asynchronously and the message is placed in the Inbound Internal Queue. Step 2 is defined with the inbound channel type set to Internal Queue.

This step starts working as soon there is a message available in the internal queue. The step picks up the message including the calculation from the queue and makes the call to SAP Business One to update the objects with the calculation results.

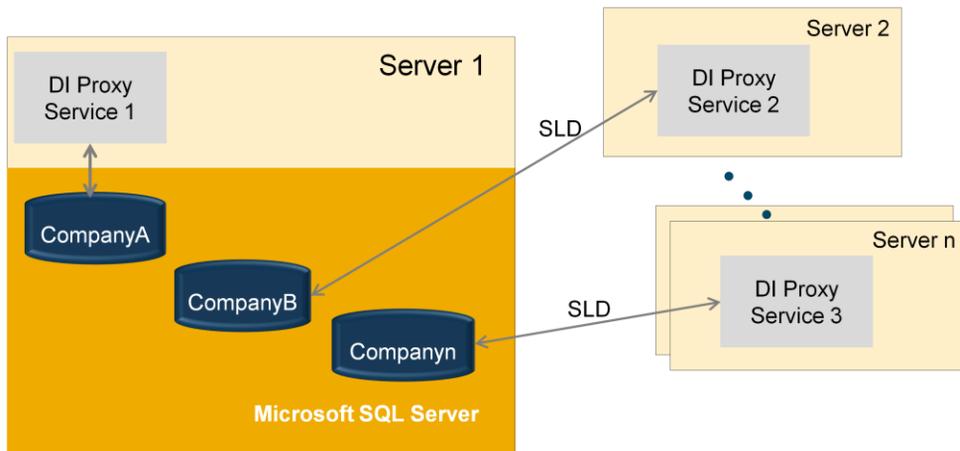
To hand over content to the step 2, you need to define the payload using an XPath expression. In this scenario example, this would be the result of the calculation which will be used to call the B1 object.

To learn more about internal queues and how to use them, see the companion course *Working with Internal Queues*.

SAP Business One Integration

Multiple DI Proxy Servers -1

- If you are working with multiple SAP Business One companies, a recommended optimization is to use multiple DI proxy services on separate physical machines
- This allows you to distribute the workload among different DI Proxy servers



© 2015 SAP SE or an SAP affiliate company. All rights reserved.

8

Another optimization to consider is the use of multiple proxy servers. As you know, the DI proxy channels data between the SAP Business One DI API and the integration framework.

The default installation provides one DI proxy service for all requests; however, if you are working with multiple SAP Business One companies, a recommended optimization is to use multiple DI proxy services on separate physical machines. This is depicted in the graphic.

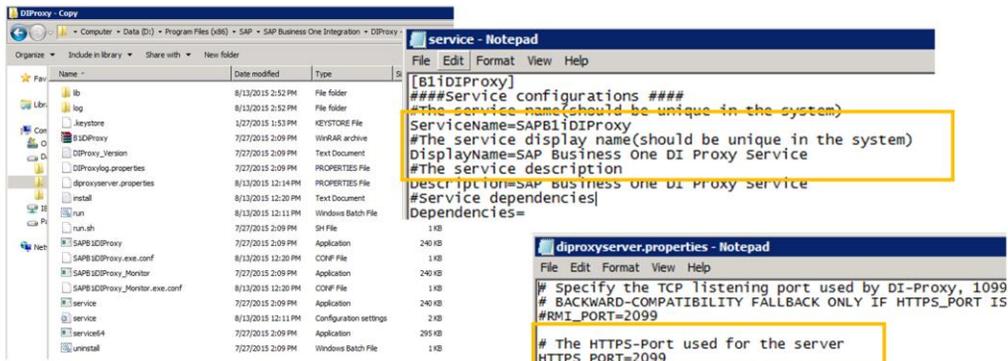
Having more than one DI proxy service allows you to allocate the message load for a company database to a specific DI proxy host and thereby distribute the workload among different DI Proxy servers.

SAP Business One Integration

Multiple DI Proxy Servers - 2

To define multiple proxy services:

- Copy and rename the default *DIProxy* folder
- Change the entries in the *service.ini* file
- Change the port number in the *diproxyserver.properties* file
- Run the service install to create a new service
- Define the address of the new proxy host in the SLD for the SAP Business One database



© 2015 SAP SE or an SAP affiliate company. All rights reserved.

9

To define multiple proxy services:

- Copy and rename the default *DIProxy* folder which is located in the SAP Business One Integration folder
- Change the service name and display name entries in the *service.ini* file in the copied folder
- Change the https port number for the new server
- Run the *service.exe* file from the copied folder to create a new service

After you have created the proxy service, define the DI proxy port in the System Landscape Directory for each SAP Business One database.

For detailed instructions on this process, refer to SAP Note 1993784 and the attached how-to guide *How to Use the Enhanced DI Adapter*. You can download the guide from the Note.

SAP Business One Integration

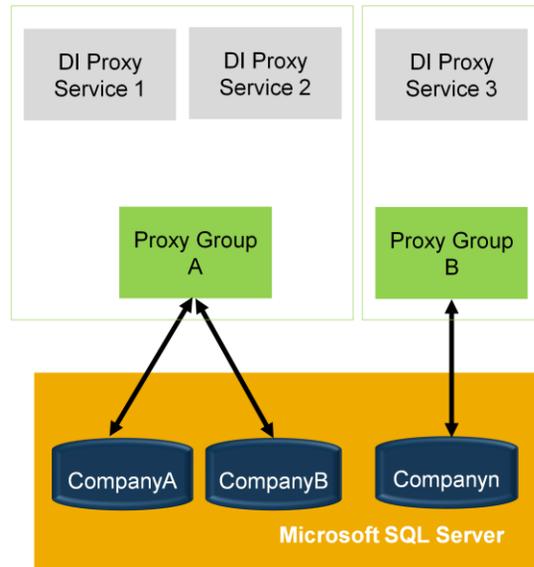
Using Proxy Groups - 1

In addition to multiple DI Proxy servers, you can use Proxy groups

Proxy groups provide an overall load balancing mechanism, for example:

- One proxy group for integrating mobile applications and company databases
- A second proxy group for all other scenarios

Requests can come from scenario steps that are independent of each other. If you process a step using a certain proxy, it uses the proxy during the complete step processing.



Using multiple DI proxy services distributes the load by company database. You can add a load balancing mechanism to further improve performance by adding proxy groups. Each proxy group manages multiple proxy services and you can group the proxy services to suit the workload.

For example, you might decide to use two proxy groups:

- One proxy group is dedicated to the integration scenario for mobile applications and multiple company databases
- The second proxy group is used for all other integration scenarios.

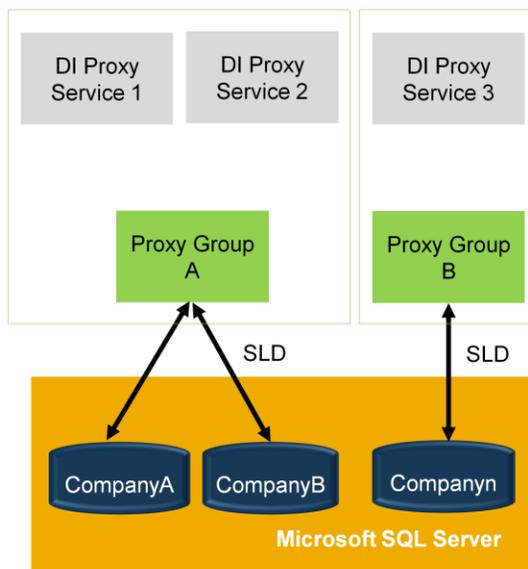
Requests can come from scenario steps that are independent of each other. If you process a step using a certain proxy, it uses the proxy during the complete step processing.

SAP Business One Integration

Using Proxy Groups - 2

To use proxy groups, first define multiple proxy services, then:

- Define the names of the proxy groups together with the hostnames or IP addresses of the assigned proxy servers in the *DI Adapter Global Configuration* (Path: Tools > Control Center > Configuration > Global Adapter Config)
- Define the proxy group instead of the proxy server in SLD entries of the SAP Business One company databases
- See SAP note 1993784 and attached How-To Guide



To define proxy groups you first need to define multiple proxy services, by copying and editing the default *DIProxy* folder.

Then, define the names of the proxy groups, together with the hostnames or IP addresses of the assigned proxy servers, in the global adapter configuration properties for the B1DI adapter.

Lastly, enter the proxy group names instead of the proxy server in the company database entry in the SLD.

For more information about setting up both multiple proxy services and proxy groups, refer to SAP Note 1993784 and the attached how-to guide *How to Use the Enhanced DI Adapter*. You can download the guide from the Note.

SAP Business One Integration

Event Sender Setup Filter Definition - 1

- SAP Business One writes events for new, changed and deleted data to a table
- Based on filter definitions, the *EventSender* service hands over events to the integration framework
- You can reduce the number of events by setting the correct filters using the *Event Filter Generator*

- Select the Event Filter Generator from the *Tools* menu
- Select the required scenario packages
- Click *Generate*
- The filters used are shown in the Event Sender Filter field

Path: *Tools* → *B1 Event Filter Generator*

Deselect All			
<input checked="" type="checkbox"/>	sap.B1RFQ	2.0.0 (2014/06/29 21:16:07)	Description
<input checked="" type="checkbox"/>	sap.B1System	1.0.9 (2014/09/15 10:22:36)	Description
<input checked="" type="checkbox"/>	sap.DATEV-HR	1.3.4 (2014/04/30 13:58:39)	Description
<input checked="" type="checkbox"/>	sap.PTLegal	1.0.0 (2013/08/23 07:39:26)	Description
<input type="checkbox"/>	sap.SME.EDI	1.3.0 (2013/09/02 13:35:18)	Description
<input checked="" type="checkbox"/>	sap.Xcelsius	1.0.12 (2011/04/13 15:25:44)	Description
<input checked="" type="checkbox"/>	Z.WSTest	1.0.0 (2015/02/02 17:03:46)	Description

The next optimization concerns the Event Sender.

As you know, the Event Sender is part of the integration framework notification mechanism. Events are created in SAP Business One, for example, the creation of a sales order. SAP Business One writes events for new, changed and deleted data to a table. Based on filter definitions, the *EventSender* service accesses the table, picks up data and hands over the events to the integration framework for further processing.

You can reduce the number of events by using filters to screen events as needed. When you set the filters, you prevent the creation of unnecessary events for your scenarios.

To define filters, open the *SAP Business One Event Filter Generator* from the *Tools* menu.

- The SAP Business One event filter generator displays packages triggered by events, as shown in the screenshot.
- Select the required scenario packages
- Generate the filter settings based on the selected scenario packages
- The filters are displayed in the *Event Sender Filter* field.

SAP Business One Integration

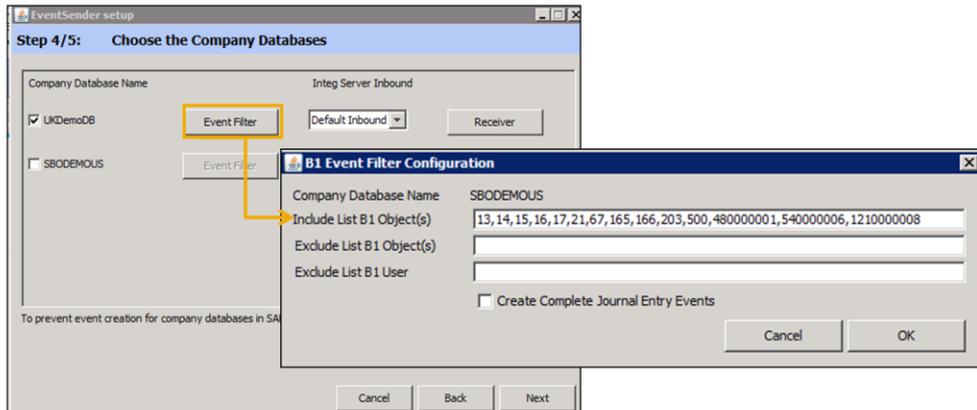
Event Sender Setup Filter Definition - 2

After you generate the event filters, start the Event Sender Setup wizard

Start → All Programs → Integration solution for SAP Business One → SAP Business One EventSender

Paste the filters from the *Event Filter Generator* to the *Event Sender Setup* wizard

As a result of the filters, the number of SAP Business One events is reduced.



After you have generated the list of event filters, start the *Event Sender Setup* wizard using the path shown in the slide.

Paste the generated filter objects from the Event Sender Filter field to the *Include List B1 Object(s)* field in the event sender setup wizard.

As a result of the filters, the number of SAP Business One events is reduced.

SAP Business One Integration

Using the DI API Single Transaction Option - 1

- Normally the integration framework only commits or rolls back at the end of the processing phase, not after each call to SAP Business One
- There is an option to switch to single transaction mode for all DI API calls. To use single transaction, select the *SAP Business One DI Single Transaction* checkbox

Path: *Scenarios* → *Step Design*

Scenario Step Design

Selection Criteria [127.0.0.1:8080]	
Scenario Package Filter	sap.SalesDocuments
Scenario Step Identifier	sap.R3INV2B1INV
Scenario Package Identifier	sap.SalesDocuments
Inbound	R3((DOC.ASYN)) - Handover - dynamic
Processing	3 processing steps
Outbound	OUT_B1 (DI Object-Insert/Update-18.DocEntry)
Version	1.0.0
Internal	<input type="checkbox"/>
MsgLog Exclusion	<input type="checkbox"/>
B1/SN88 Competibility Mode	<input type="checkbox"/>
SAP Business One DI Single Transaction	<input checked="" type="checkbox"/>
Actions	

Explanations

Scenario Package Filter: Filter for Scenario Step Dropdown List
Scenario Step Identifier: Scenario Step Identifier
Scenario Package Identifier: Scenario Package Identifier

Message from webpage

Each DI call runs in a separate transaction. Change behavior of SAP Business One DI calls?

OK Cancel

The final optimization for SAP Business One is the use of the single transaction option.

Normally the integration framework only commits or rolls back at the end of the processing phase, not after each call to SAP Business One. This can result in long running DI API calls.

The single transaction option was introduced in version 1.20.5 of the integration framework. To use single transaction mode, select the *SAP Business One DI Single Transaction* checkbox in the Scenario Step Design.

SAP Business One Integration

Using the Single Transaction Option - 2

The result of the single transaction option is:

- Each call to SAP Business One is a separate transaction
- Scenario step processing is no longer one atomic transaction
- You are responsible for data consistency

Single transaction option recommended only for:

- Reading data from SAP Business One
- Writing data **once** only to SAP Business One

Refer to How To Guide *How to Use the Enhanced DI Adapter*

Not recommended for:

- Multiple calls to SAP Business One for writing data, especially if a call is dependent on another call

To make multiple calls to SAP Business One for writing data, distribute the calls to several scenario steps that you connect to each other using the Call Scenario Step atom

The result of the single transaction option is that each call to SAP Business One is a separate transaction. The scenario step processing is no longer processed as an atomic and consistent transaction, therefore you are responsible for data consistency.

Because of this, SAP recommends using the single transaction option *only* for these situations:

- Accessing SAP Business One for reading data.
- Writing data to SAP Business One **only once**, if the call to SAP Business One is the last in processing.

Do not use the single transaction function if you perform several calls writing to SAP Business One, especially if the write operations depend on each other. To make several calls to SAP Business One for writing data, distribute the calls to several scenario steps that you connect using the Call Scenario Step atom.

Summary

In this course, you were introduced to several techniques for optimizing performance of the integration framework:

- A general recommendation is to set up filters for sender and receiver systems to avoid unnecessary step processing
- For scenarios that interact with SAP Business One:
 - Avoid calls to SAP Business One from loops and use internal queues instead
 - Consider setting up multiple DI proxy services, and proxy groups to balance the requests
 - Define event sender filters to screen SAP Business One events
- Use the single transaction option for reading data from SAP Business One, or writing data **once** only to SAP Business One
- Check SAP Note **2192737** periodically for updated recommendations

In this course, you were introduced to several techniques for optimizing performance of the integration framework:

- A general recommendation is to set up filters for sender and receiver systems to avoid unnecessary step processing
- For scenarios that interact with SAP Business One:
 - Avoid calls to SAP Business One from loops and use internal queues instead. From SAP Business One release 9.1 PL08, use the Call Scenario Step atom or for earlier releases use the Put to Queue atom
 - Consider setting up multiple DI proxy services, and proxy groups to balance the requests from SAP Business One companies
 - Define event sender filters to screen SAP Business One events
- Use the single transaction option for reading data from SAP Business One, or writing data once only to SAP Business One
- Check SAP Note **2192737** periodically for updated recommendations



Thank You!

For more information visit SDN at <http://sdn.sap.com>

© 2015 SAP SE or an SAP affiliate company.
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 SE or an SAP affiliate company.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. Please see <http://global12.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.

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

National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE or SAP affiliate company 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.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.