

SAP Exchange Infrastructure



New Features of the Integration Builder

SAP AG

Please note, that this document is subject to change and may be changed by SAP at any time without notice. The document is not intended to be binding upon SAP to any particular course of business, product strategy and/or development.

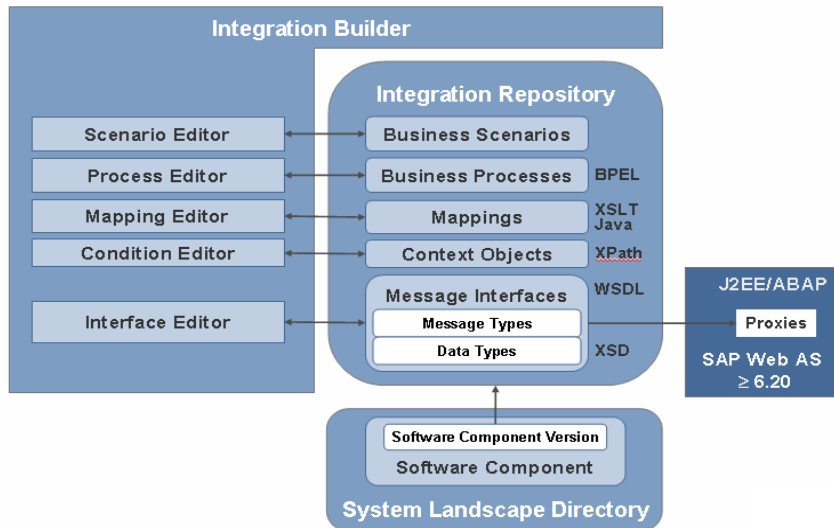
THE BEST-RUN BUSINESSES RUN SAP



Contents

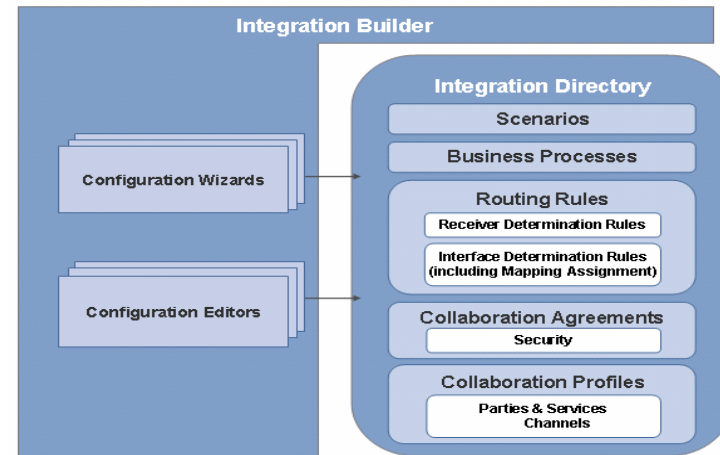
1

Integration Builder – Design Time



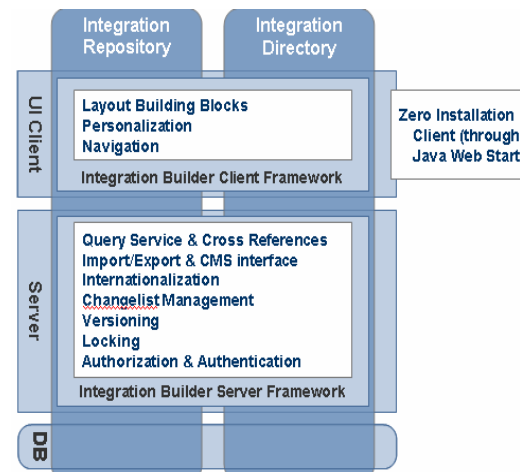
2

Integration Builder – Configuration Time

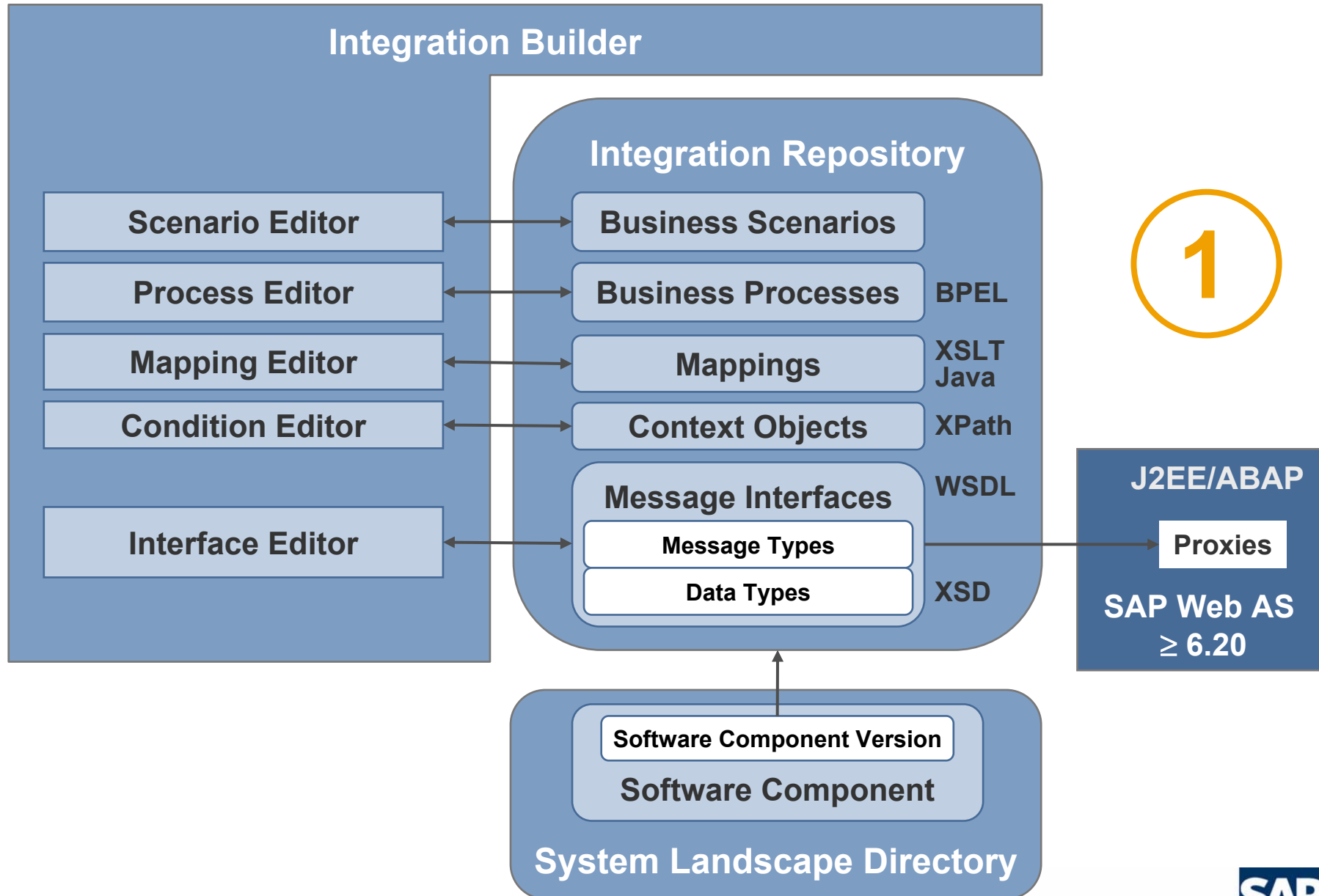


3

Integration Builder – Generic Functions



Integration Builder – Design Time



New: Business Process Objects

You can design new objects for executable **business processes** by using the graphical Process Editor.

You can use different modeling elements and patterns to design a stateful, cross-component business process.

The following modeling patterns and elements are supported:

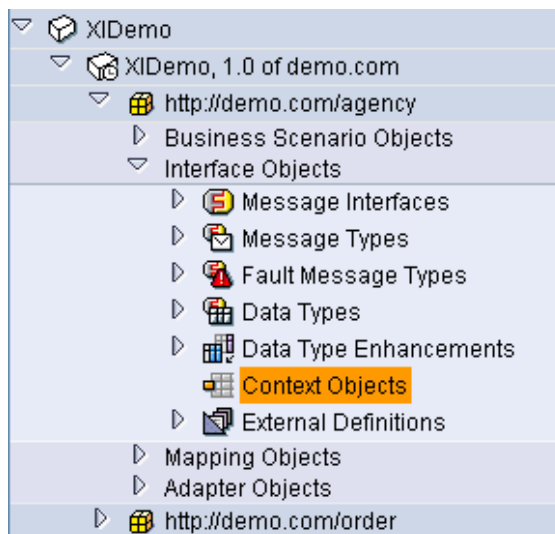
- Receive messages and trigger process
- Send, transform (merge or split) messages
- Collect messages
- Multicast
- Serialization
- Process control elements such as switch, assign, fork, wait, block, loop, control
- Deadlines, exception handling, conditions

New: Context Objects

Context Objects

- Encapsulate access to data that is contained in the payload or in the header (technical context objects) of a message
- Increase the readability of routing rules because no XPath knowledge is needed at design or configuration time to formulate a condition
- Can be used in logical routing and in business process models to access and evaluate message data

1. Create Context Object



2. Use Context Object in Message Interface

Context Objects of Input Message

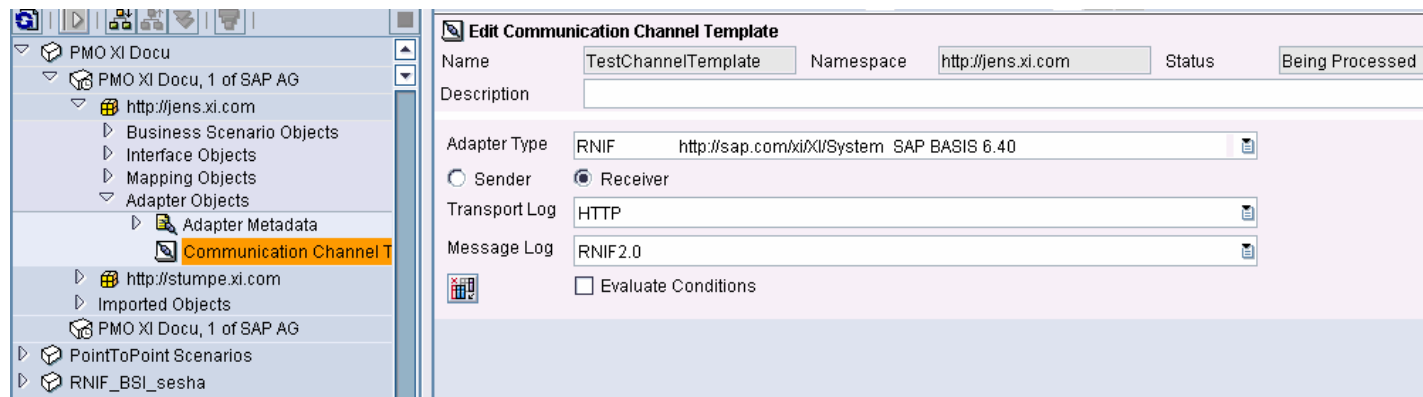
Structure	Category	Type	Context Object	Description
BookingOrderConfirmati...	Element	p0:BookingOrder...		
AgencyID	Element	xsd:string		Agency ID
OrderNumber	Element	xsd:string		Order number
BookingStatus	Element	xsd:string		Booking status
BookingID	Element	p0:BookingID		Booking ID
AirlineID	Element	xsd:string		Airline ID
BookingNumber	Element	xsd:string		Booking number

New Object: Channel Template

Designers can define **channel templates** in the Integration Repository and deliver them with the software component version.

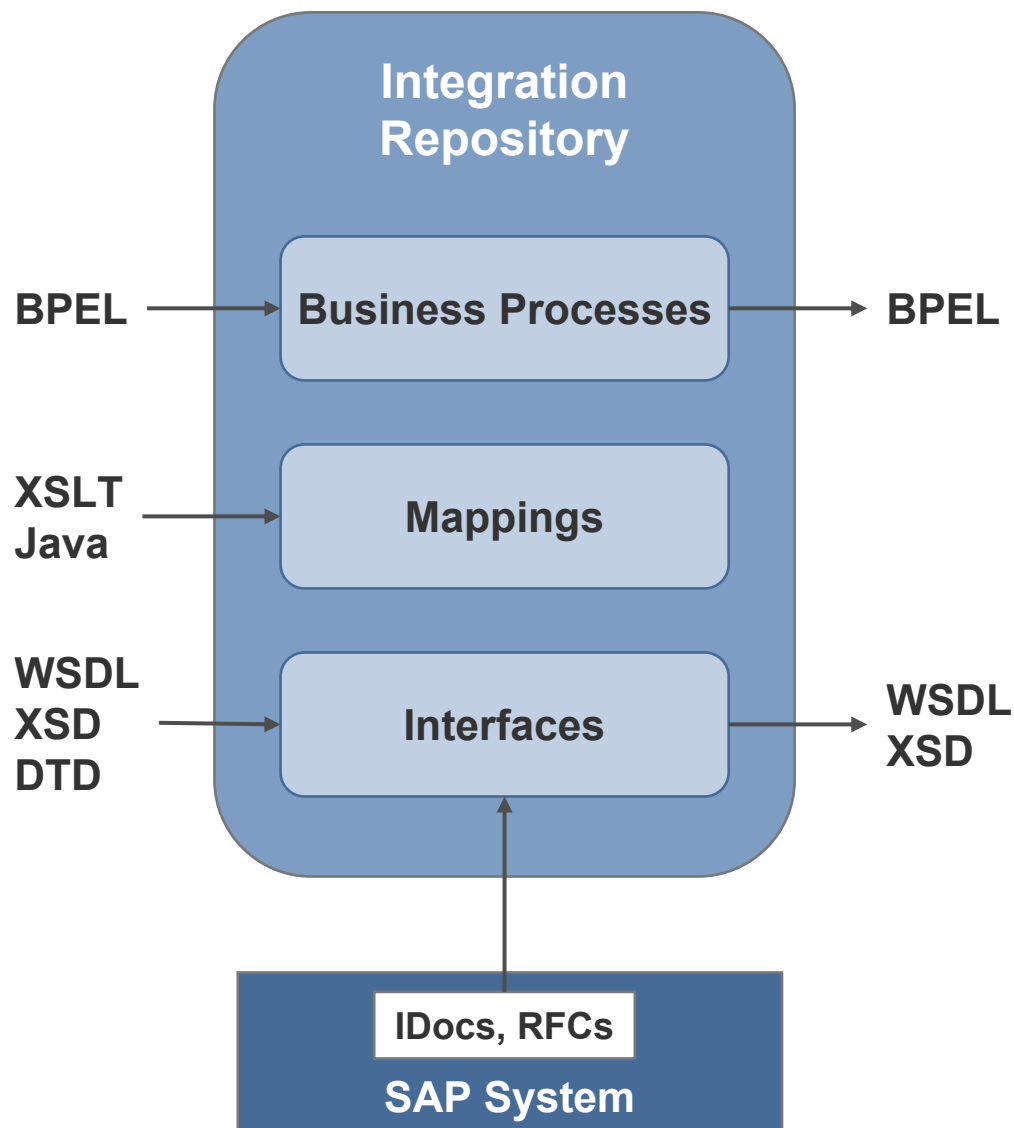
Configurators can use these templates to configure channels (that is, information on message protocol, transport protocol, URL and logon data) in the Integration Directory.

The instantiation of a repository template, which copies the template to the directory, is supported by a wizard.



Main usage of this feature in XI 3.0: RosettaNet PIPs

Import / Export Function



■ The import function for WSDL, XSD and DTD files enables you to upload message definitions from external sources.

■ After conversion to WSDL they can be used to define message mappings and message interfaces.

■ The new object type **External Definition** is a container which is used to make external message definitions available in the Integration Repository.

Message Mapping

Enhancements to **message mapping** functions:

- You can split and merge messages for business processes (multi-mapping).
- You can define mapping templates at data type level.
Advantage: Reuse of mappings in new message mappings.
- You can use Document Type Definitions (DTDs) in message mappings by importing them as an external definition.

The screenshot displays the SAP Message Mapping Designer interface. On the left, a tree view shows the project structure under 'PMO XI Docu', including 'Mapping Objects' and 'Mapping Templates'. The 'myFirstMappingTemplate' is selected. The main area shows the 'Display Mapping Template' dialog with the following details:

- Name: myFirstMappingTemplate
- Namespace: http://jens.xi.com
- Status: Active

The 'Design' tab is active, showing two data type editors side-by-side:

- Data Type: Person**

	type	Occurrences	De
Person	p0:Person	1..1	
gender	xsd:decimal	1..1	
surname	xsd:string	1..1	
- Data Type: Passenger**

	type	Occurrences	Descr
Passenger	p0:Passenger	1..1	
title	xsd:string	1..1	
name	xsd:string	1..1	

There are several new functions which make it **easier to navigate**:

- Where-used list for functions and source fields
- All fields can be searched for by their name
- XPath rules can be stored in the clipboard

Enhanced test facilities:

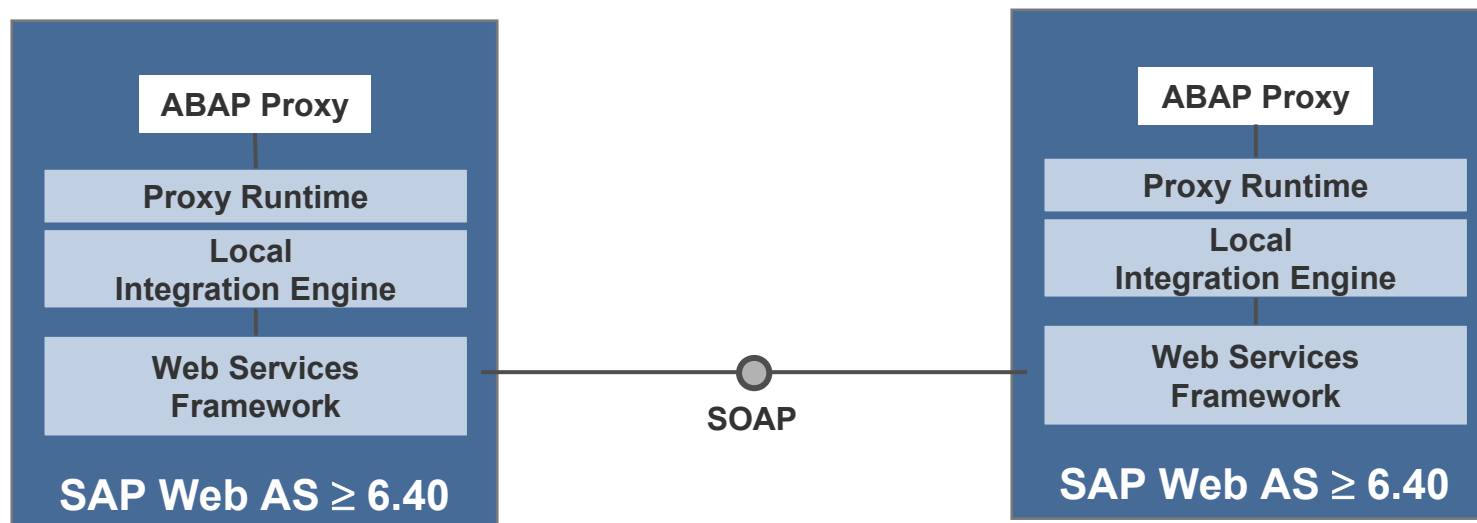
- XML instances which serve as test cases can be generated out of source message structures
- Instances can now be depicted and changed in an structural editor
- Instances can be stored and are transported
- For the textual representation of an instance there is a pretty-print function

P2P Optimization

Message Exchange Between ABAP Proxies (synchronous only)

If no services of the Integration Server are required (mapping, receiver and interface determination), the generated ABAP proxies can communicate point-to-point by using the Web Service Infrastructure of the SAP Web AS.

Advantage: Improved performance, while knowledge about interfaces is still located centrally in the Integration Builder.



Up to now, the Integration Repository supported versioning of all repository objects at software component level. **Support Packages of a software component version** are now also supported.

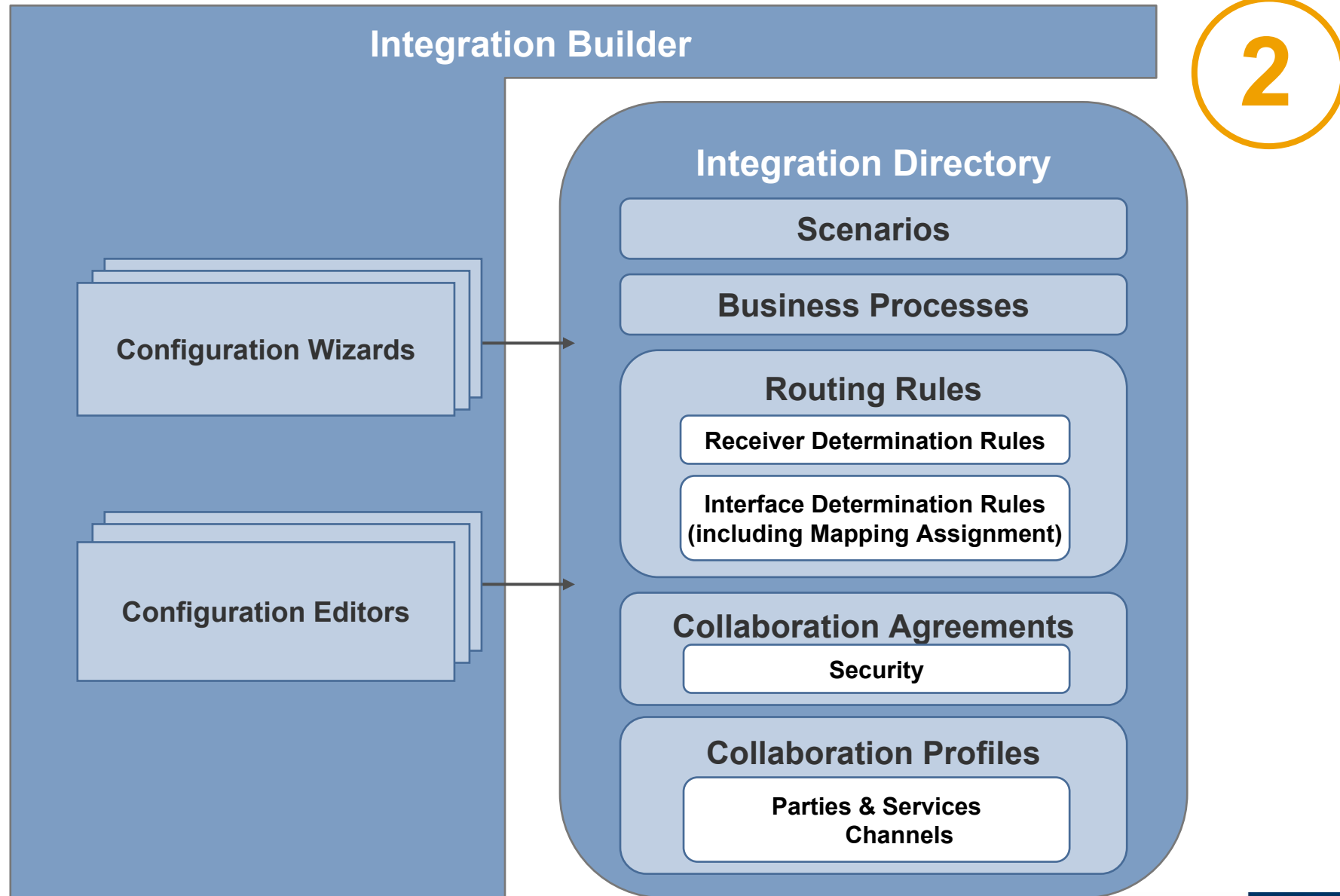
Message interfaces can be marked as “abstract“ (instead of “inbound“ or “outbound“). This applies to interfaces which are used in BPM or which are handled by adapters.

Advantage: There is no need to define two interfaces (inbound and outbound) if their structure is identical.

It is possible to enhance **data types** shipped by SAP with customer-specific fields. These enhancements will not be lost in the event of a release upgrade.

The information modeled in **business scenarios** can be used to describe product release combinations. A separate “component view“ can be defined for each release.

Integration Builder – Configuration Time



New Object: Party

The new object **party** facilitates the B2B functions of SAP Exchange Infrastructure and contains the following information:

- Name, description, and additional identifiers (DUNS, DUNS+4, GLN) of the party
- Assigned Services (explained on the following slide)
- Assigned Channels (message protocol, transport protocol, URL, logon data, adapter configuration)

The screenshot displays the SAP Exchange Infrastructure configuration interface. On the left, a tree view shows the hierarchy: RosettaNet Location > Party and Services > Party > RNIFTestPartyBSI. The 'RNIFTestPartyBSI' node is highlighted with a yellow circle, and an orange arrow points to it from the left. The right pane shows the 'Display Party' details for 'RNIFTestPartyBSI'. The 'Party' field is 'RNIFTestPartyBSI' and the 'Description' is 'BSI Party'. Below this, the 'Alternative Identifiers' table is shown:

Assigning Agency	Identification Schema	Identifier
016	DUNS	123456789

This enables **adapters** to be **configured centrally** in the Integration Directory

New Object: Service

The new object **service** can represent:

- A business system (in A2A scenarios)
- Semantical units (grouping of interfaces), which are used mainly for routing purposes in the communication between multiple B2B parties. An example would be a specific PIP (RosettaNet).
- A business process

The screenshot displays the SAP Business Suite interface for configuring a service object. On the left, a tree view shows the hierarchy: RosettaNetLocal0C1 > Party and Services > Party > RNIFTestPartyBSI > Service > PIP0C1_R0102_Initiator. An orange circle highlights the 'PIP0C1_R0102_Initiator' node, with an orange arrow pointing to it from the left. The main area shows the 'Display Service' configuration for this object. The 'Service' field is 'PIP0C1_R0102_Initiator', the 'Party' is 'RNIFTestPartyBSI', and the 'Description' is 'Partner initiates PIP0C1'. The 'Status' is 'Active'. Below this, the 'Business Service' section is visible, with tabs for 'Receiver' and 'Sender'. The 'Inbound Interfaces' section is currently empty, showing a table with columns for 'Name' and 'Namespace'.

Collaboration Profile

The new party, service and channel objects represent the capabilities of communication partners and can be subsumed under the term “**collaboration profiles**”.

Cardinality:

- Parties can offer any number of services, but have to offer at least one (1:n).
- A service can be assigned to a party (n:1) but can also be used without such an assignment. This makes sense in A2A scenarios (business system as service) or when a service represents a business process.
- Channels are assigned to a service (n:1).

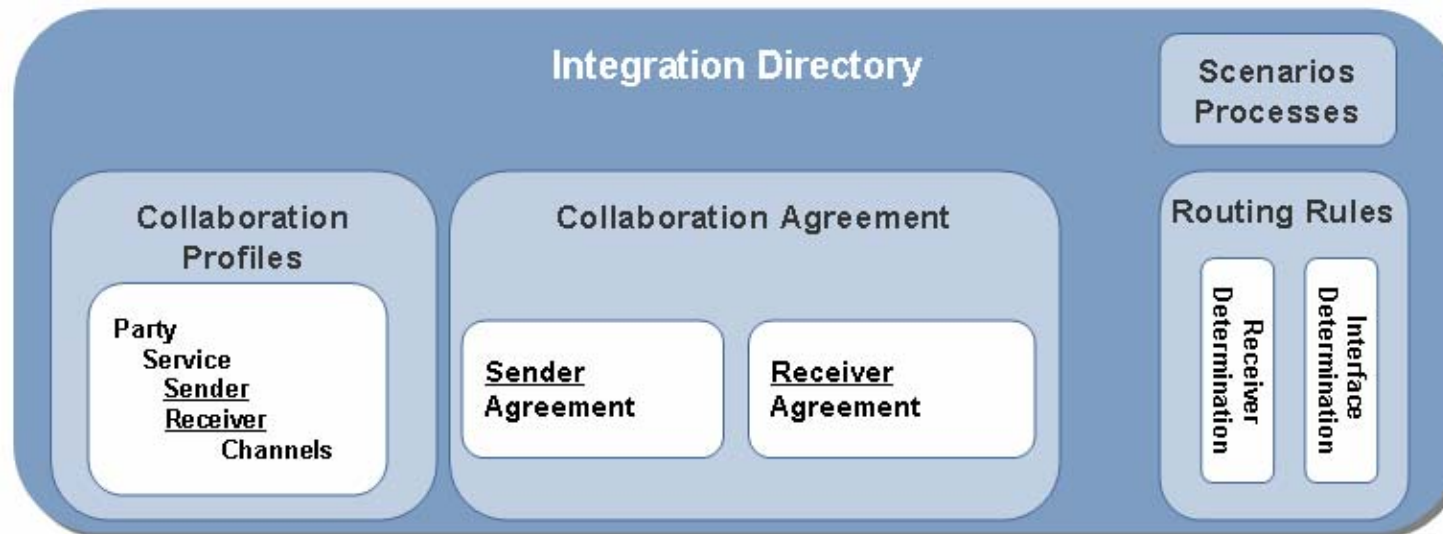
Party and service objects can act as senders and receivers, depending on the scenario in which they are used.

Collaboration Agreement

The new objects **sender agreement** and **receiver agreement** describe which of the various possibilities (message protocol, transport protocol and so on) are actually used at runtime.

The sender agreement is used on the Integration Server for inbound processing, the receiver agreement for outbound processing.

Sender agreement and receiver agreement can be subsumed under the term “**collaboration agreement**”.



Receiver Determination Rules

Receiver determination rules determine which receiver a certain message has to be sent to at runtime.

In XI 2.0 you can define content-based routing rules manually by using XPath in the condition editor.

In XI 3.0 you can now also reference context objects, which have been defined beforehand in the Integration Repository. These context objects improve usability as they hide the XPath language from the user.

The **business process object** in the Integration Directory contains a read-only link to its originating repository process and is used to define the required receiver determination rule. Thus, business processes may appear as receivers or senders of messages in receiver determination rules.

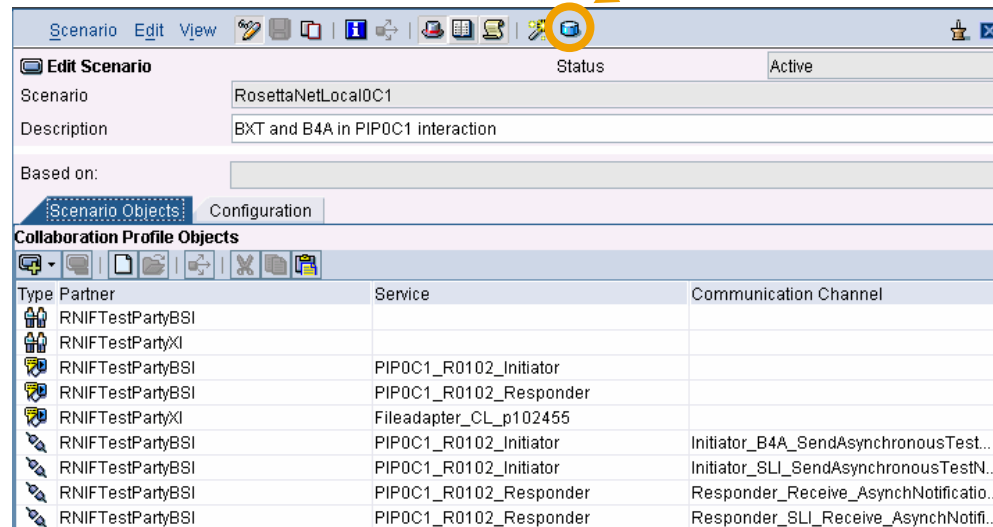
Business process objects do not contain a process definition and can only be deployed, not created in the Integration Directory.

Scenarios

In general, **scenarios** in the Integration Directory serve as (optional) groupings for all kinds of configuration objects.

Scenarios thus allow configurators to define task-specific views on directory objects.

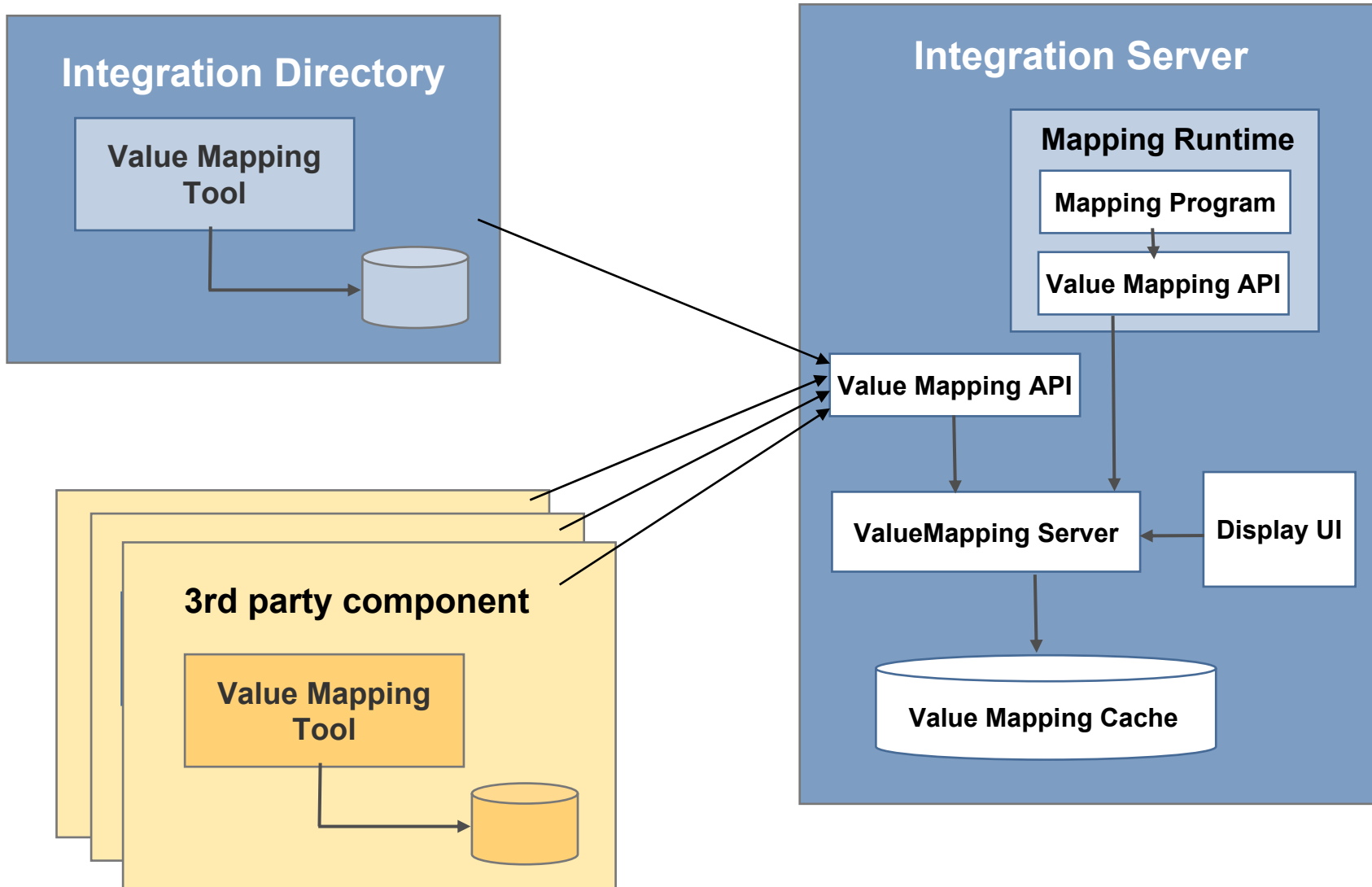
These scenarios may refer to a repository scenario. In this case you can generate receiver and interface determinations and adapt these according to customer needs (Business Scenario Configurator).



The screenshot shows the SAP Business Scenario Configurator interface. The 'Scenario Objects' tab is active, displaying a table of Collaboration Profile Objects. A yellow arrow points to the 'Scenario Objects' tab icon in the top toolbar.

Type	Partner	Service	Communication Channel
	RNIFTestPartyBSI		
	RNIFTestPartyXI		
	RNIFTestPartyBSI	PIP0C1_R0102_Initiator	
	RNIFTestPartyBSI	PIP0C1_R0102_Responder	
	RNIFTestPartyXI	Fileadapter_CL_p102455	
	RNIFTestPartyBSI	PIP0C1_R0102_Initiator	Initiator_B4A_SendAsynchronousTest...
	RNIFTestPartyBSI	PIP0C1_R0102_Initiator	Initiator_SLI_SendAsynchronousTestN...
	RNIFTestPartyBSI	PIP0C1_R0102_Responder	Responder_Receive_AsynchNotificatio...
	RNIFTestPartyBSI	PIP0C1_R0102_Responder	Responder_SLI_Receive_AsynchNotifi...

Value Mapping



Migration XI 2.0 – XI 3.0

Where do I find my objects after an upgrade?

Business systems migrate to “partyless” services.

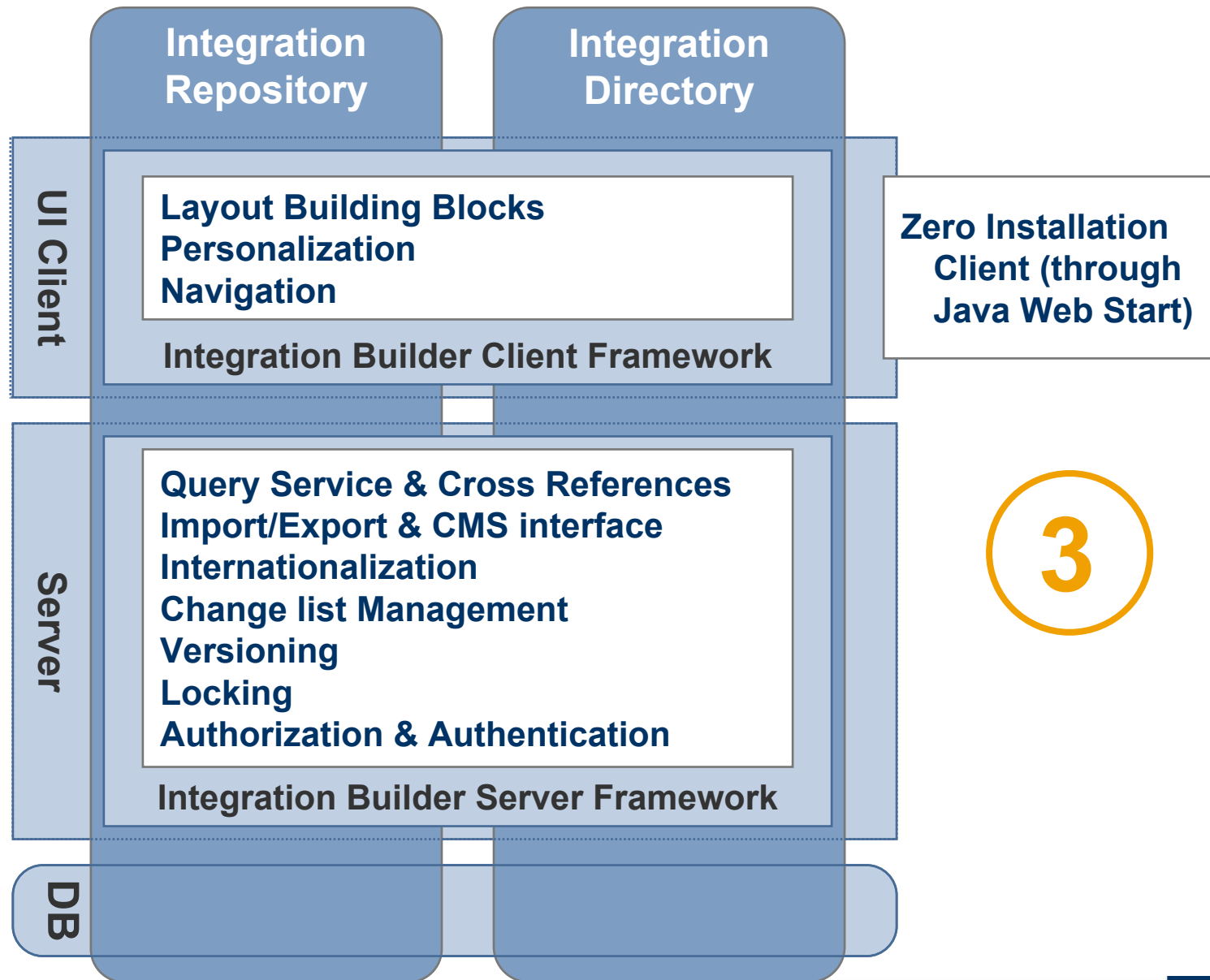
Receiver determinations of a sender business system and interface belonging to different XI 2.0 scenarios will be grouped in one receiver determination.

No further changes to interface determinations.

End points and logon data will create channels to the business systems and receiver agreements – There may be a n:m relation because of the combinatorics.

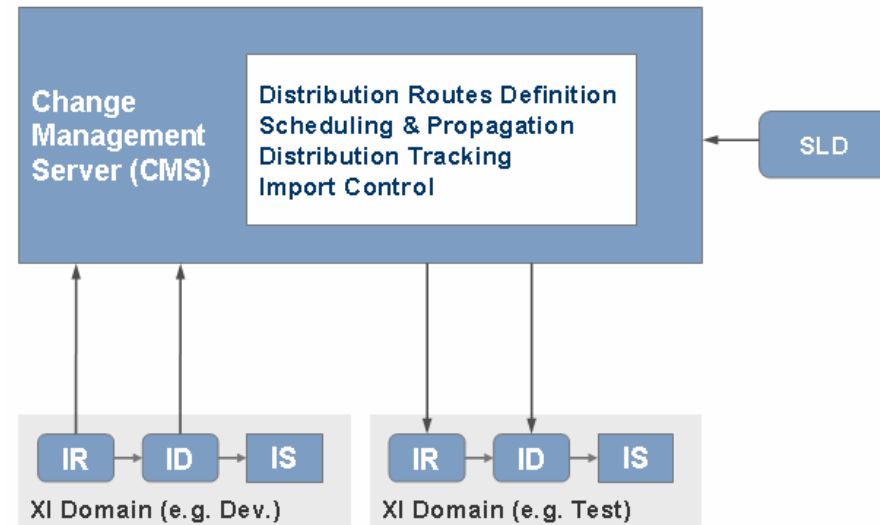
Scenarios will migrate to scenarios, which group receiver determinations.

Integration Builder – Generic Functions



Transport Management

The Integration Builder uses the **change and transport management** functions provided by the **Change Management Server (CMS)**.



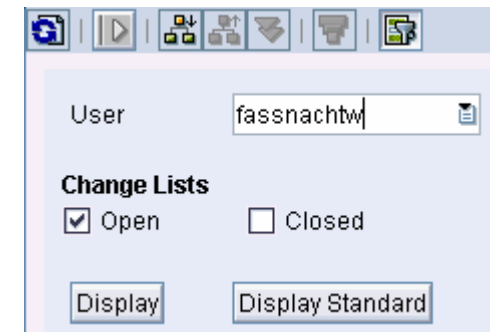
All objects of the repository and directory can be transported.
You can package objects in different ways:

- All objects of a specific change list (design and configuration time)
- All or some objects of a software component version or of a namespace (Integration Repository)
- A freely defined selection of directory objects
- All objects of a scenario (Integration Directory)

Development Environment and Change Management

Functional enhancements in the development environment and change management:

- Where-used lists make dependencies between objects more transparent
- Each user can create an infinite number of change lists (for managing changes)
- You can display error messages in an amodal message box and simultaneously work on the corresponding error
- Interface objects in the Integration Repository can be copied including all dependent objects



Authorization, Personalization and Accessibility

Functional enhancements regarding **authorization**, **personalization** and **accessibility**:

- Authorizations for actions such as display, change, or delete can be assigned to individual objects and groups of objects.
- The user interface of the Integration Builder can be personalized by each user to fit individual requirements.
- Users with disabilities can access the most important functions and editors.
- Keyboard shortcut keys are supported.

The **new features of the Integration Builder** add business value because they

- Enable you to model real world, stateful cross-system scenarios, increase flexibility when implementing a shipped scenario by using customer enhancements, handle the import of external message definitions and make mappings easier to design (**Integration Repository**)
- Introduce the configuration of B2B and BPM functionalities, a centralized adapter maintenance and optimizations like the enhanced value mapping function (**Integration Directory**)
- Support multiple users simultaneously and with increased data security when working together on integration projects by means of a flexible change management system and finely tunable authorization concept (**Generic Functions**)

- 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®, NT®, EXCEL®, Word®, PowerPoint® and SQL Server® 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 and Informix® Dynamic Server™ are trademarks of IBM Corporation in USA 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®, the Citrix logo, ICA®, Program Neighborhood®, MetaFrame®, WinFrame®, VideoFrame®, MultiWin® and other Citrix product names referenced herein are trademarks of Citrix Systems, Inc.
- HTML, DHTML, XML, XHTML 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.
- MarketSet and Enterprise Buyer are jointly owned trademarks of SAP AG and Commerce One.
- SAP, SAP Logo, R/2, R/3, mySAP, mySAP.com 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 trademarks of their respective companies.