

# Reference Guide

The xUML Services Reference Guide is a comprehensive guide for building reusable xUML services with the **Scheer PAS BRIDGE®**, in support of Advanced SOA initiatives that combine Business Process Management (BPM), Service Oriented Architecture (SOA), and Event Driven Architecture (EDA) concepts.

Model Driven Integration (MDI) is a groundbreaking approach to integrate, migrate, and consolidate heterogeneous distributed software systems. It raises the level of abstraction from non-transparent, mostly undocumented code to unambiguous executable models, which serve simultaneously as documentation and production runtime. With MDI, executable models in form of UML®, BPMN® and ARIS EPC are used as the exclusive source to build complex business application and integration scenarios.

By reading this document, you will learn how to create xUML services, which integrate lower level backend access functions to communicate with existing backend applications and services. The xUML Runtime infrastructure houses the various interface bridging adapters for synchronous and asynchronous application connectivity. Out-of-the-box, the E2E Bridge supports a broad selection of public and proprietary interface standards, covering access to databases, systems, message protocols, and platform-specific connectivity, including Java EE and the proprietary tRFC protocols from SAP.

The notion of a **service** is used in the SOA request/response pattern: a service is a function that is well defined, self-contained, and does not depend on the state of other services. However, most frequently it depends on interfaces of other services.

Nowadays, the term **service** is often used synonymously with Web service. Although Web services are an important service type, by reading this document you will learn not only how to model Web services but also other kinds of services such as SAP RFC services or timer services used for EDA patterns.

## Topic Overview

Section [Bridge Architecture Overview](#) introduces the basic Bridge concepts and the general workflow of building Bridge-compliant and executable UML models.

Section [Basic Structural Modeling](#) explains structural modeling elements like classes, relationships, and attributes.

Section [Basic Behavioral Modeling](#) describes how the behavior of classes and operations can be defined by using activity diagrams.

Section [xUML Action Language](#) references all operations, macros, and functions provided by the E2E Action Language, which implements parts of the Action Semantics UML Extensions.

Section [Advanced Behavioral Modeling](#) addresses advanced topics like the concept and implementation of persistent states and the security concepts that can be realized with the Bridge using interceptors and proxies. Other topics describe how to use E2E Libraries for well-defined interfaces to encapsulated functionality of your service, monitoring capabilities, and the object-oriented concept of polymorphism.

Section [xUML Service Adapters](#) lists all adapters. An adapter is a component having a special behavior, like for example a file adapter that can be used to read files from within a Bridge service. For each component, we describe how class-, activity-, and component diagrams are used. Furthermore, we show for each adapter how to define the deployment diagram. Each adapter introduces its own profile that is a set of stereotypes and tagged values used to specify adapter properties. This profile is also described for each adapter.

Section [Service Implementations](#) lists all E2E services. An E2E service is a service having a special behavior. The SAP service for example can be used to design a Bridge service that acts like a SAP system. For each service, we describe how class-, activity-, and component diagrams are used. Furthermore, we show for each service how to define the deployment diagram. Each E2E service introduces its own profile that is a set of stereotypes and tagged values used to specify service properties. This profile is also described for each E2E service.

Section [xUML Service Model Debugging, Testing and Quality Assurance \(QA\) Concepts](#) explains by which concepts and tools the Bridge supports testing and debugging.

Section [Import - Export Mechanisms](#) contains import rules. The point of using UML for integration modeling is to unify the view on software artifacts. This includes also a unification of various data models as class structures. This chapter shows how different data models are mapped to UML classes by E2E Importer modules. For example, the Bridge maps WSDL definitions, XML Schemas, or SAP IDoc grammars to UML class models. The semantic differences between these models and UML classes are handled by stereotypes and tagged values.

Section [Appendix](#) contains a list of all Bridge error codes, the E2E Profile used in E2E Builder, regular expressions, char set definitions, database-specific notes, database-specific mappings, and ABAP types mappings (for SAP RFC).

### On this Page:

- [Topic Overview](#)
- [Preconditions](#)
- [Conventions Used in this Reference Guide](#)
- [Example References](#)

# Preconditions

An understanding of object-oriented concepts is useful. Basic knowledge of UML and usage of a UML tool (e.g. MagicDraw™) will simplify the learning process.

# Conventions Used in this Reference Guide

The syntax of operations and statements of the xUML Action Language is described using the following Backus Naur Form (BNF) style conventions:

- {x} denotes zero or more occurrences of x.
- x | y means one of either x or y.

The following typographical conventions for explaining syntax are used in the reference guide:

E2E Action Language statements	Action Language statements are always formatted using a monospaced font.
<i>substitutable</i>	Substitutable information, which represents a name, a value, an expression, etc. and needs to be provided by you, is always formatted using the monospaced font in italic style - example:  <code>set <i>booleanVariable</i> = true   false;</code>  In this example, a name for <i>booleanVariable</i> must be provided by you. You have the choice to assign either true or false to it. A resulting line of script could look like this:  <code>set bl = true;</code>

# Example References

Several Scheer E2E sample UML models are provided with the Bridge installer. You can unpack them to a folder of your choice as described in [Unzipping the Builder Examples](#). If provided for a discussed topic, you will find example references as shown below at the beginning of a chapter.

In the example reference below, the array sort example is part of the E2E Builder project **E2E Action Language**. Open this Builder project and choose the UML model **arrayConcat.xml**.

**Example File (Builder project E2E Action Language/Array):**

	<your example path>\E2E Action Language\Array\uml\arrayConcat.xml
---	---

The <your example path> is the path where you installed the examples to. Default path is **C:\E2E Documentation**.

Clicking the download icon in the example reference allows you to download the latest version of the example project from the documentation pages.