XML Serialization

Some Designer components and operations are using a built-in serialization of internal data model to XML and vice versa. These are:

- Components:
  - REST Adapter

    The serialization procedures are used, when serializing Designer classes to a REST call and serializing back the response.
- Operations:
  - xmlToClass() Operation for Strings
  - xmlToClass() Operation for Blobs
  - classToXML() Operation
  - classToXMLFragment() Operation
  - classToJSON() Operation
  - jsonToClass() Operation

To the serialization in general, the rules listed on this page further below apply. You can control these serializations from and to XML documents by assigning stereotypes to data model elements that define an XML schema.

XML_Serialization_Example

Click the icon to download a simple example model that shows how to control XML mappings with stereotypes in **Scheer PAS** *Designer*.

Related Pages:

- REST Adapter

Related Pages:

- xmlToClass() Operation for Strings
- xmlToClass() Operation for Blobs
- classToXML()
- classToJSON() Operation
- jsonToClass() Operation

General Serialization Rules

The following general serialization rules apply to class to XML serialization:

Level	Rule
general	The XML root element is named after the name of the object that contains the data to be mapped.
class	Each class who's type is not derived from a built-in simple type (e.g. **String**) is serialized to an XML element.
attribute	Each attribute of simple type (or derived from a simple type) is serialized to an XML attribute of the XML element defined by the containing class.
	Each attribute of complex type is serialized to an XML element.

The following basic XML Schema content models are supported by the Designer:

Content Model	Description
all	Serializes all elements of the data model to the XML target document. All names of sub-elements must be unique, the order irrelevant.
sequence	The elements of the data model are serialized by their **order** definition.