

# The JaxMe parser for XML Schema

## 1. The JaxMe parser for XML Schema

This is JaxMeXS, a parser for XML schema. Yet another one, to be precise. As there are a lot of other parsers around, the question arises: What makes it different? What advantages does it have?

Let's begin with a disadvantage: This one is not a strict parser. In other words, it was not written to detect any possible invalid schema. Strictness is a secondary goal.

The main purpose of JaxMeXS is being extensible. It is written with XML languages in mind, that extend XML schema. The best example is [JAXB](#), the XML binding specification for Java. It extends XML schema by adding some more tags, for example like this:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://ws.apache.org/jaxme/examples/misc/jaxb"
  xmlns:jaxb="http://java.sun.com/xml/ns/jaxb">
  <xs:annotation>
    <xs:appinfo>
      <jaxb:schemaBindings>
        <jaxb:package name="com.mycompany.xml" />
      </jaxb:schemaBindings>
    </xs:appinfo>
  </xs:annotation>
  ...
```

JAXB is a Java source generator, which converts an XML schema into Java classes. These Java classes have the ability to read an XML document conforming to the original schema and return its information. As usual for Java classes, they have to be located in some package, `com.mycompany.xml` in our case. The above example demonstrates how JAXB uses an additional tag `jaxb:package` to specify it.

This is exactly where JaxMeXS fits in. Quite unsurprising, because it was written as the XML schema parser for [JaxMe](#), which aims to be a JAXB implementation and extends the JAXB language by even more tags. The main idea of JaxMeXS is:

- Write a parser for XML schema.
- Extend the parser with the JAXB tags. This is simple: Any additional element is implemented by a standard Java bean. The attributes are mapped to properties and the property setters are called almost automatically.

- Extend the JAXB parser with the JaxMe tags by adding more beans.