

# XmlBeansSampleSuggestions

## Suggestions for XMLBeans Samples

Here's a partial list of suggestions for XMLBeans samples. This page is provided so that you can get some inspiration for writing your own samples, and so that you can add ideas for samples that you think are needed.

This list is divided primarily into the technology areas that the XMLBeans API is typically divided into. Of course, some of the most interesting and useful samples will show how to do something that requires (or is easiest with) multiple parts of the API. For those samples, add your suggestions to the Combinations group at the bottom of this list. The ideas here are expressed as questions about how things work (with the sample being an answer), but feel free to express your suggestion in another way.

### Schema

XMLBeans provides full schema support. One goal for schema-related samples is to have a sample set that illustrates XMLBeans' handling of the range of things that can be expressed in schema. In other words, were there's a question about how aspects of schema are manifested in XMLBeans, we should have a sample.

- How are enumerations handled?
- How do schema date and time types show up?
- How does one handle elements defined as xs:any? How does the value for the processContents attribute affect XMLBeans' validation of instances?
- How can I control the way type and package names are generated from schema during compilation?
- How are substitution groups handled?
- Basic methods to create a simple XML document from given schema
- Basic ways to read the fields of an XML document

[Add your ideas.]

### Cursor

The XML cursor provides low-level access to XML, complementing XMLBeans' schema support.

- How does one move chunks of XML from one document to another (or to another part of the document)?
- How can one use bookmarks to keep track of specific locations in XML?
- How do the push and pop methods work? What are some example algorithms that use them?
- How can a cursor be used to manage mixed content XML?

[Add your ideas.]

### Schema Type System

The schema type system is XMLBeans' view of the types defined in schema – in other words, schema metadata. You can use this API to find out about the schema at run time, in Java code.

- What kinds of things are knowable via the schema type system? How are these represented by the API?
- How can one retrieve schema specifics for a given element?
- Can a database schema model of tables and columns be created from an xml schema?

[Add your ideas.]

### DOM (beginning with v2)

DOM support is planned for XMLBeans version 2. Whereas in version 1 you can obtain a DOM-oriented copy of XML that is represented by an XMLBeans schema type, in version 2 you will be able to manipulate the same XML in the same code through the cursor, schema types, and DOM API.

[Do you have DOM experience through other technologies? Add your ideas for XMLBeans-specific samples.]

### Combinations

Interesting capabilities (and sometimes limitations) arise when you combine the multiple ways that XMLBeans provides for handling XML.

- Is there some way to retrieve and store XMLBeans directly on XML Databases instead of XML files? Some developers store XML documents in XML Databases and would be nice to have a [DataSourceFactory](#) to connect to this database to make some queries on XML data returning XMLBeans.

[Add your ideas.]

### Pre and Post event subscription

Instead of having to create and compile a static handler for pre and post event notifications, what if I could subscribe to a node for pre and post events on a node, and that subscription (optionally?) propagates to all parent nodes.

This way I can handle changes to documents in-context instead of globally.

For example, I am implementing a forms-based editor for my document, and would like to be notified whenever fields change so I can log changes and notice when there is anything new that needs to be saved by the user.

## XSD enums as Enum objects

It would be great to generate XSD enums as java 1.5 enums.