XML-Constituents:

elements

- tag name (required)
- attributes (optional), there can be an ID attribute, attributes can be ID references
- content (optional)

content can be...

- text (CDATA, PCDATA)
- CDATA is only text (character data)
- PCDATA can be text + entities + elements (parsed character data)
- nested elements

document consists of

- XML header with version and encoding
- optionally PIs (Processing Instructions)
- optionally a schema reference
- optionally an inline-DTD
- exactly ONE root element
- the root element with its children forms a tree

Object networks, i.e. instances of classes and their links, in general form an arbitrary structured graph (or even many graphs)

Discussion.

Among others, the following rules were discussed. One option was to use nesting to represent some of the associations of the domain model. Though possible, the following rules don't use nesting. The resulting document tree is rather "flat".

Rules:

- Objects are represented as elements with an ID attribute, tag name = class name
- Elements are not nested, but all objects are kept in a list
- Attributes are represented as sub-elements, tag name = attribute name
- We need an "artificial" root element, tag name = something unique
- Edges can be represented by IDs and IDREF, IDREFS
- Edges are contained in element that represents source object and refer to ID of target object, tag name = relation name

Example XML document (without schema/DTD):

```
<?xml version="1.1" encoding="utf-8"?>
<University-Data>
      <Universtity id="uko">
            <name>Universität Koblenz-Landau</name>
            <HAS_DIVISION target="fb4"/>
            <HAS_DIVISION target="fb1"/>
      </University>
      <Department id="fb4">
            <name>Faculty 4</name>
            <description>Computer Science</description>
      </Department>
      <Department id="fb1">
            <name>Faculty 1</name>
            <description>Humanities</description>
      </Department>
</University-Data>
```