Prof. Dr. Stefan Brass
Institut für Informatik
MLU Halle-Wittenberg

October 27, 2017

XML and Databases
— Exercise Sheet 2 —

Part a) to d) will be solved in class, you only have to submit Part e). But please think about the questions in Part a) before the meeting! Send your solutions for the homework e) to the instructor via EMail: `brass@informatik.uni-halle.de` (with “`xml17`” in the subject line). The official deadline is November 2, 10:00 (before the lecture time).

Not all submitted homeworks will be corrected, but all homework exercises will be discussed in class. If you should have questions about your homework, please ask! A precondition for getting credit for this course is that you submit solutions to two thirds of the homework sheets. Obviously wrong or very incomplete submissions do not count.

If you should miss the tutorial session (Thursday, 12-14), it is recommended that you solve the “in-class exercises” yourself.

In-Class Exercises

a) Answer the following questions (similar questions might be asked in an oral exam):

- What is the difference between well-formed XML and valid XML?
- How can one include texts in XML that may contain “`<`” and “`&`”? What would you do if these characters are seldom? What would you do if there is a longer piece of text in which these special characters are quite common (e.g. Java source code)?
- What are entities (more precisely: general, parsed, internal entities)? What would be a typical application? Explain the syntax of the declaration of such entities and the syntax of entity references.
- Please enumerate the five entities that are predefined in XML.
- What are parameter entities? How do declaration and usage differ from the normal “general” entities? Why does it make sense to distinguish the two types of entities?
- How does a processing instruction look like? What is the purpose of processing instructions? Why is a processing instruction very similiar to an XML declaration?
- Which syntactical possibilities are there for a `DOCTYPE` declaration?

- Explain the overall structure of an XML file.
- How can a table from a relational database be translated to XML? Discuss different alternatives.
- Elements in XML have always a defined order (an XML file is a document), whereas table rows are stored in no specific order in a relational database. Explain how this might influence the translation between relational data and XML (in bothe directions).
- Compare keys and foreign keys in a relational database with `ID` and `IDREF` in XML.
- How can one translate a one-to-many relationship to XML? Name at least two different possibilities. Discuss restrictions and possible problems.
- How can one translate a many-to-many relationship to XML?
- Explain how an arbitrary ER-diagram with one-to-many and many-to-many relationships can be translated to XML.

b) Please look at the XML 1.0 specification:

[https://www.w3.org/TR/REC-xml/]

Alternatively, you can use “The Annotated XML Specification” by Tim Bray:

[http://www.xml.com/axml/testaxml.htm]

Please find out which characters are allowed as first characters of a name in XML (e.g. element names). (Hint: Read Section 2.3 “Common Syntactic Constructs”.) It suffices if you concentrate on ASCII characters. However, you might be interested also in German national characters (“Umlaute”). The official source for the codes of these letters is:

[http://unicode.org/charts/PDF/U0080.pdf]

However, any ISO Latin 1 table suffices (the character codes are identical to Unicode, only the encoding as UTF-8 is different).

c) Please have a look at the XHTML 1.0 Strict DTD:

[https://www.w3.org/TR/xhtml1/dtds.html#a_dtd_XHTML-1.0-Strict]

What is the content model of the element “`table`”? Please name a permitted sequence of child elements (only the names of the elements, not the full XML/XHTML text). Which data type has the attribute `lang` (search also for `i18n`)?

d) Consider the following relational database and develop a DTD for an XML representation of the data. It might be useful to draw a directed graph with the relations as nodes and the foreign keys as edges. Remember that nesting can only represent trees, so you have to cut some edges and represent these edges e.g. with `ID`/`IDREF`.

The database is part of a data dictionary of a relational database. It consists of the following tables:

- `USERS(USER, FULLNAME°)`

USERS	
USER	FULLNAME
SYS	Data Dictionary Owner
SB	Stefan Brass

- `TABS(OID, OWNER→USERS, TABNAME)`
  The second table contains a list of all tables stored in the system. Tables are identifed by an "object ID", and alternatively by table owner and table name.

TABS		
OID	OWNER	TABNAME
1	SYS	USERS
2	SYS	TABS
3	SYS	COLS
4	SYS	GRANTS
5	SB	STUDENTS

- `COLS(OID→TABS, COLNO, COLNAME)`
  This table has one row per table column. The entries are identified by the `OID` of the table and the number of the column within the table.

COLS		
OID	COLNO	COLNAME
1	1	USER
1	2	FULLNAME
⋮	⋮	⋮

- `GRANTS(OID→TABS, GRANTOR→USERS, GRANTEE→USERS, PRIV, GRANTABLE)`
  This table contains the information who (`GRANTOR`) has given whom (`GRANTEE`) an access right (`PRIV`) for a table (`OID`), and whether this was "`WITH GRANT OPTION`".

GRANTS				
OID	GRANTOR	GRANTEE	PRIV	GRANTABLE
1	SYS	BRASS	SELECT	N
⋮	⋮	⋮	⋮	⋮

Homework Exercises

e) Please develop a DTD for your XML encoding of the nutrition supplements database from Homework 1. If you want, you can change the structure of your data file. Please check that your data file really validates with respect to your DTD. Submit the data file and the DTD (possibly also in one file) and name the tool that you used for the validation.

- `Supplement(Name, Producer)`
  The underlining indicates primary key attributes.

Supplement	
Name	Producer
Centrum	Lederle
One A Day	Bayer

- `Vitamin(Substance, Daily_Value, Unit)`
  The second column contains the recommended daily value, and the third column the unit used for amounts of that vitamin.

Vitamin		
Substance	Daily_Value	Unit
Vitamin A	5000	IU
Vitamin C	60	mg
Biotin	300	mcg
Iron	18	mg

- `Contents(Name → Supplement, Substance → Vitamin, Quantity)`
  This table corresponds to a many-to-many-relationship between "Supplement" and "Vitamin" (the arrows indicate foreign keys). "Quantity" is the contents in one tablet, with the same units as in table "Vitamin".

Contents		
Name	Substance	Quantity
Centrum	Vitamin A	5000
One A Day	Vitamin A	5000
Centrum	Vitamin C	60
One A Day	Vitamin C	60
Centrum	Biotin	30
Centrum	Iron	18
One A Day	Iron	27

Of course, the concrete data listed here are incomplete and probably outdated. I used this example in a database exam in Pittsburgh in 1999.