COMP624121 Querying Data on the Web

Lab Work XQ

Consider again the **Mondial** database but now with the data represented in XML. We provide you with three versions:

I. the original one (which we'll refer to as **the single-file version**) from the Institute for Informatics in the Georg-August-Universität Göttingen, in which there is a single XML file (for which both .xsd and .dtd files are available), called **mondial.xml**;
II. another version (which we'll refer to as **the values-in-attributes version**) which was obtained by conversion from the relational version in which for each table T, each tuple t in T generates an attribute-only XML element, leading therefore to as many XML files as there are relational tables, each called **<T>-a.xml** for a table T;
III. finally, a second version (which we'll refer to as **the values-in-elements version**) which was obtained by conversion from the relational version in which each tuple in a table T generates an XML element with the attribute values as elements too, giving rise again to as many XML files as there are relational tables, each called **<T>-e.xml** for a table T.

For this lab work you must use the **BaseX** native XML database[1]. For Task 1, you will find it most convenient to use **basexgui** as this provides a more convenient interface for you to compose and debug your queries.

Task 1: Consider again the following English-language specifications (re-used from Lab Work RQ above):

(1) Return the names of countries that are not landlocked (i.e., have a sea coast).
(2) Return the names of all lakes, rivers and seas.
(3) Return the average length over all the rivers in the database.
(4) Return the name of countries that have more than 10 islands.
(5) Return, for every river in Great Britain, the length of that river.
(6) Return the name of the countries that have the 10 longest total length of rivers.
(7) Return all the information available about cities whose name is Manchester.
(8) Return the name of cities whose name starts with the substring 'Man'.
(9) Return the name of the country and the name of the organization of countries with Buddhist populations that are members of organizations established after 1st December 1994.
(10) Return the name of each country with the number of islands in it.

In Lab Work RQ you wrote RA expressions for (1)-(6) above and SQL expressions for (7)-(10) above. You will re-use your solutions here. Firstly, translate (1)-(6) to SQL. Secondly, make a script of your translation of (1)-(6) adding to these your original solution to (7)-(10). Now, run the script in **sqlite3** and note of the response times for running those queries.

Task 2: Write XQuery expressions *against the original **mondial.xml** file* that, upon evaluation, return the data characterised by the above English-language specifications. If you find that there is a mismatch between what the SQL query and the XQuery query can return, make a note of that. Run the script in **BaseX** and note of the response times for running those queries. Note also the size of the file.

Task 3: Write XQuery expressions *against the **<T>-a.xml** files* that, upon evaluation, return the data characterised by by the above English-language specifications. If you find that there is a mismatch between what the SQL query and the XQuery query can return, make a note of that. Run the script in **BaseX** and note of the response times for running those queries. Note also the size of the files involved/invoked in each query.

Task 4: Write XQuery expressions *against the **<T>-e.xml** files* that, upon evaluation, return the data characterised by by the above English-language specifications. If you find that there is a mismatch between what the SQL query and the XQuery query can return, make a note of that. Run the script in **BaseX** and note

[1] http://basex.org/

of the response times for running those queries. Note also the size of the files involved/invoked in each query.

Task 5: Summarize your investigation with a plot, accompanied by interpretation and comment, that compares the use of a relational DBMS on relational data and the use of a native XML database on the three different XML approaches to modelling the same data.

Marking

- Each query in Task 1 is worth up to 1 mark, for a total of up to 10 marks.
- Each query in Task 2 is worth up to 2 marks, for a total of up to 20 marks.
- Each query in Task 3 is worth up to 2 marks, for a total of up to 20 marks.
- Each query in Task 4 is worth up to 2 marks, for a total of up to 20 marks.
- Task 5 is worth up to 30 marks.
- The whole lab is worth 100 marks and contributes up to 10 marks to the final mark for the course unit.

Software/Data

Versions of Mondial in XML

You need the database to be local to you, so that you have write permissions on it. Data is always under:

> /opt/info/courses/COMP62421/data

For this lab work, you will use the three XML versions of the **Mondial** database, which need to be copied into your private file space from

> **the single-file version**
> /opt/info/courses/COMP62421/data/Mondial/xml/Mondial.xml
>
> **the values-in-attributes version**
> /opt/info/courses/COMP62421/data/Mondial/xml/converted/a.tar.zip
>
> **the values-in-elements version**
> /opt/info/courses/COMP62421/data/Mondial/xml/converted/e.tar.zip

Note that for the latter two versions, you will need to unzip the tar file and untar it to generate the folder where the many files actually lie. If you do something like:

```
unzip a.tar.zip
tar xf a.tar
unzip e.tar.zip
tar xf e.tar
```

you should end with two folders (called 'a' and 'e') and inside each of them one XML file per table in the relational version that you used in Lab Work RQ (Week 1).

BaseX

BaseX is already installed in the lab machines (but not on kilburn.cs.man.ac.uk, so beware).

Since the system is very easy to install when you have privileges for doing so, you may want to install it in a personal computer too, but, note, we must assume that you are working from the lab, i.e., we cannot control/compensate for issues arising from installation in personal machines.

You will not use the client-server model, but rather a local one. So, beware making things unnecessarily complicated for you. **BaseX** has a homepage at http://basex.org/

The documentation is available in http://docs.basex.org/wiki/Main_Page (but note that the online documentation is for v. 8.3, and we'll be using v. 8.1, though nothing of great substance should have changed to the point of affecting your work).

The two relevant commands are

```
/usr/bin/basex
/usr/bin/basexgui
```

The former is a command-line interface as well as accepting input from the command line. Use

```
man basex
```

to learn more and, in particular, for later, how to pass queries and scripts using standard input.

The later is a graphical user interface and should be the one you explore first. Use

```
man basexgui
```

to learn more. In the case of the GUI version, if you want to ssh into a lab machine, remember to tunnel it through X and, plus or minus latency problems, you should be able to work on it.