



Thomas Zastrow, Groningen May 2006

Processing dialect data using an XML database



The Data

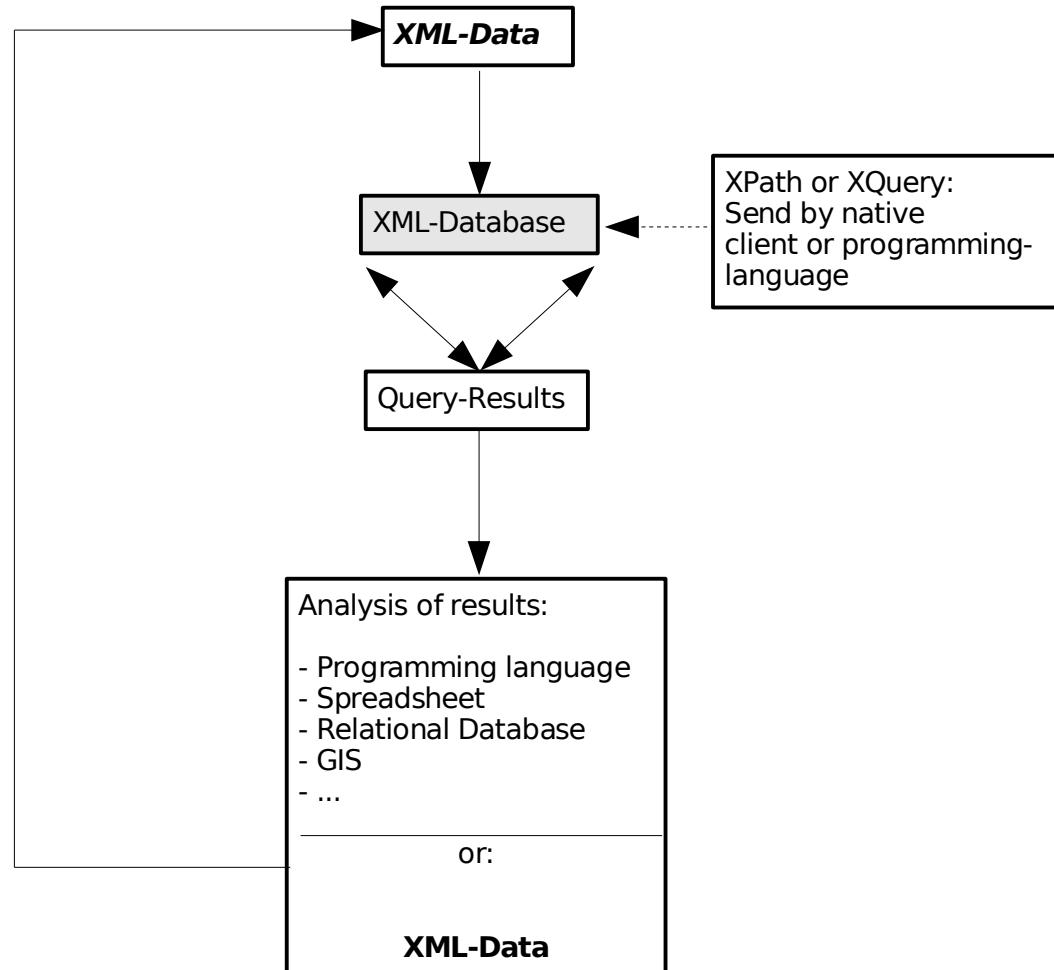
- ◆ **Ultimately, we will have 200 sites**
- ◆ **At the moment: 66 + 14 without coordinates**
- ◆ **Annotated in XML, using the CLaRK-system**
- ◆ **Ca. 155 words per site**

In Tübingen:

- ◆ **Integration into the XML database eXist**
- ◆ **Generation of wordlists in L04-format via XSLT**
- ◆ **Querying the database with the native client and from Java**



The XML Workflow





The eXist-database

- ◆ **OpenSource, written in Java**
- ◆ **Runs on Windows, Linux, ...**
- ◆ **A lot of interfaces: XML-RPC, SOAP, HTTP, ...**
- ◆ **Programing languages: Java, Perl, PHP, ...**
- ◆ **Integrated webserver**
- ◆ **Native client**

- ◆ **XPath and XQuery for querying the database**
- ◆ **XSL transformations included**
- ◆ **XML validation against DTDs**

- ◆ **In Tübingen:**
- ◆ **Most programs are written in Java**
- ◆ **XML-RPC is used as interface from within the Java programs**
- ◆ **Based on Java 5, developed in NetBeans IDE**



XPATH / XQUERY

- ◆ **XPath: Getting all sampa/variant-tags which contain a blank**

```
collection( "/db/nbd" )//sampa/variant[contains(., " ") ]
```

- ◆ **XQuery: Building a list of all entries which have an english-tag “lamb”**

```
let $nbd := collection("db/nbd")  
  
for $entry in $nbd//entry where  
    string($entry/english)="lamb"  
return <site>{$entry/..../name}{$entry/..../num}{$entry}</site>
```



The eXist database

The screenshot shows the eXist Admin Client interface with three main windows:

- File Browser:** A window on the left showing the directory structure of the database. It displays files like `PODiad.dtd`, `beli.dat.xml`, `beryaha.dat....`, `byala.dat.xml`, `bychva.dat.xml`, `chasha.dat.xml`, `cheresha.dat....`, `dadoh.dat.xml`, `den.dat.xml`, `djob.dat.xml`, `dobyat.dat.xml`, `doshyl.dat.xml`, `dva.dat.xml`, `ezik.dat.xml`, and `greshka.dat`. The file `beli.dat.xml` is currently selected.
- Query Editor:** A window in the center containing an XQuery editor. The history pane shows the query `1. //variant[contains(., "t_s")]`. Below it, the query input pane contains the XQuery `//variant[contains(., "t_s")]`. The results pane shows the output of the query, which consists of many URLs starting with `s:exist="http://exist.sourceforge.net/NS/exist"`.
- Command Line:** A window at the bottom showing a terminal-like interface. It displays the command history:

```
type help or ? for help.  
exist:/db> cd "bulidialect"  
exist:/db/bulidialect> cd "data_petya"  
exist:/db/bulidialect/data_petya> cd "54-words"  
exist:/db/bulidialect/data_petya/54-words
```

The status bar at the bottom indicates: "Found 19 items. Compilation: 3ms, Execution: 2503ms".



The EXIST Database

The screenshot shows the XQuery Sandbox interface running in Mozilla Firefox. The URL in the address bar is `http://localhost:8080/exist/sandbox/sandbox.xql`. The main area contains the XQuery code:

```
document("./db/nbd/bansko.txt.xml")//site
```

Below the code, there are buttons for **Send**, **Clear**, and **Check**. To the right, there are buttons for **Display: 20** and **More Options**. The results section displays the following XML output:

```
Found 1 in 0.027 seconds.
Showing Items 1 to 1
1 <site>bansko, mzl
<num>4082</num>
<name>bansko</name>
<entry>
  <key>bane-agne</key>
  <english>bane</english>
  <clform ana="A_ncnf">jarne</clform>
  <nform>agne</nform>
  <variant ana="Nncnf">jagne</variant>
  <sampa>
    <info>agne</info>
    <variant ana="Nncnf">"jagne</variant>
  </sampa>
</entry>
<entry>
  <key>aaz</key>
  <english>aaz</english>
  <clform ana="Ppe-ost">aaz</clform>
  <nform>aaz</nform>
  <variant ana="Ppe-ost">ja</variant>
  <sampa>
    <info>aaz</info>
    <variant ana="Ppe-ost">"ja</variant>
  </sampa>
</entry>
<entry>
  <key>bene-nr</key>
  <english>white-plastic</english>
  <clform ana="A_p13dje">b3lie</clform>
  <nform>b3lie</nform>
  <variant ana="A_p13">b3lie</variant>
  <sampa>
    <info>b3lie</info>
    <variant ana="A_p13">"b3lie</variant>
  </sampa>
</entry>
```

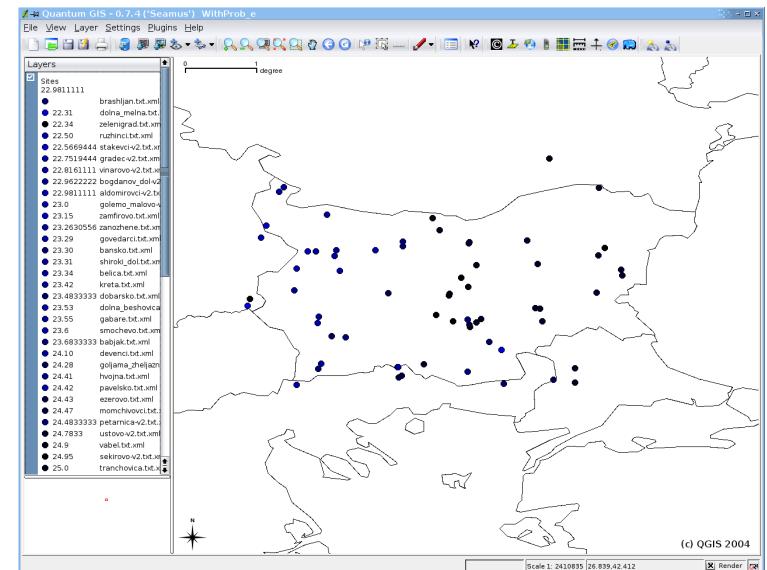
The results are displayed in a scrollable window with a yellow header.



Visualization with a GIS

- ◆ **GIS: Geographical Information System**
- ◆ **A geographical coordinate system**
- ◆ **Visual data is arranged in layers**
- ◆ **Layers can contain vector- or raster-data**
- ◆ **Combination of different layers**

- ◆ **QGis, an OpenSource-GIS is used here**
- ◆ **QGis uses XML as data format**





References

- ◆ **eXist XML database:** <http://exist.sourceforge.net>
- ◆ **XQuery standard:** <http://www.w3.org/XML/Query>
- ◆ **XQuery tutorial:** <http://www.w3schools.com/xquery/default.asp>