Melting Pot XML

Bringing File Systems and Databases One Step Closer

Christian Grün   Alexander Holupirek   Marc H. Scholl
DBIS Group, U Konstanz

BTW2007, Aachen, March 2007

Long term perspective

Find synergies between semi-structured database and file system techniques

Database guy’s dream

Query the file system

(like a database)

File Systems

- Fast and reliable storage ✔
- Proven and stable interface (VFS) ✔

☞ Therefore FS have not fundamentally changed in years

Increase of personal data

- convenient access ✘
- information retrieval ✘
- query capabilities ✘

☞ ... but FS have not fundamentally changed in years

The right mixture

- Journaling, recovery already ported to FS
- Jim Gray speaking of a FS/DBMS détente*
- Pat Selinger demands to join forces

* détente (french): *release from tension* (USENIX FAST 05)

Semi-structured data

- Tree-aware databases
- Hierarchical file systems
- Information contained in files and file systems can be expressed in XML

```
/
|-- bin
|-- etc
|   `-- services
|-- usr
`-- var
```

```
<dir name="/">
    <dir name="etc">
        <file name="services"/>
    </dir>
</dir>
```

```
/
|-- bin
|-- etc
|   `-- services
|-- usr
`-- var
```

```
<dir name="/">
    <dir name="etc">
        <file name="services">
            #
            # Network services, Internet style
            #
            # Note that it is ...
        </file>
    </dir>
</dir>
```

```
<file fs:name="Contrapunctus 9 a 4 alla Duodecima.mp3" ...
      fs:type="audio/mpeg">
  <mp3:content mp3:track="9/11" mp3:version="id3v2"
               xmlns:mp3="urn:fsxml:content:mpeg7:id3v2:simplified">
    <mp3:title>Contrapunctus 9 a 4 alla Duodecima</mp3:title>
    <mp3:albumtitle>Die Kunst der Fuge</mp3:albumtitle>
    <mp3:comment>BWV 182</mp3:comment>
    <mp3:creator>
      <mp3:role mp3:type="artist">
        <mp3:name>Robert Hill</mp3:name>
      </mp3:role>
      <mp3:role mp3=type="composer">
        <mp3:name>Johann Sebastian Bach</mp3:name>
      </mp3:role>
    </mp3:creator>
    <mp3:recordingyear>1970</mp3:recordingyear>
    <mp3:genre>Classical</mp3:genre>
  </mp3:content>
</file>
```

[MPEG7]

Punch line

- Map FS into (internal) XML representation
- Map FS operations to XPath/XQuery
- Feed into an XML-aware database
- Get *a feeling* regarding performance

Ad-hoc evaluation

Is it possible to achieve interactive response time by implementing/simulating a file system using a general-purpose XML-aware DB?

mappedfs docs

	Number of elements			
filename	<dir>	<file>	<txt:content>	<mp3:content>
mappedfs.struct.xml	1.445	17.040	—	—
mappedfs.xml	1.445	17.040	6.128	1.422
phobos04.xml	32.819	244.065	81.999	1.592

filename	**attributes**	**incl. contents**	**file size**
mappedfs.struct.xml	314.906	—	7M
mappedfs.xml	319.172	6.128	230M
phobos04.xml	3.664.208	81.999	8.6G

Table 1. Numbers about XML documents containing mapped file systems

Evaluated queries

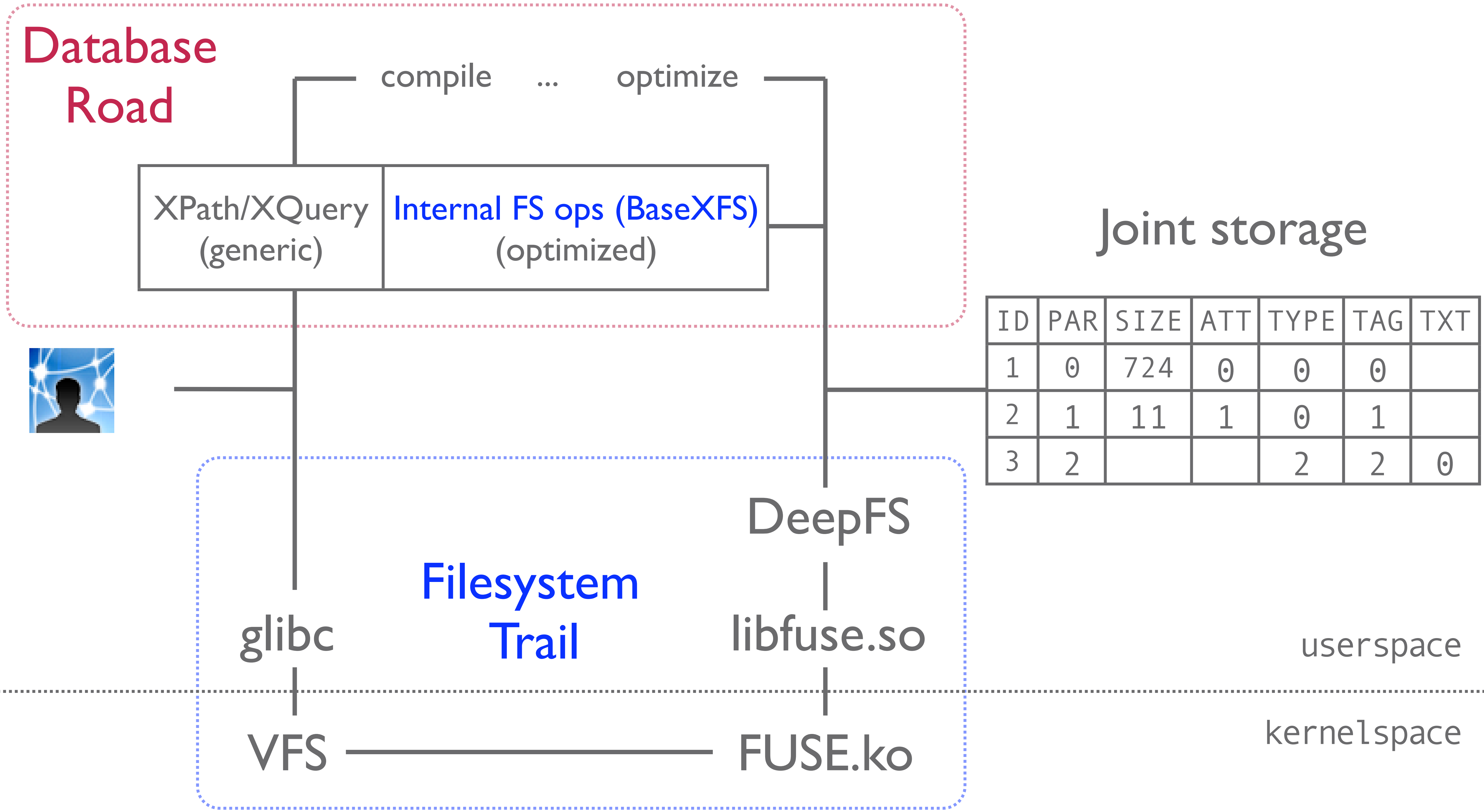
- Navigation along directory hierarchy and into files
- Modifications (mkdir, ls, rm ...)
- Search for file names & partial strings in content
- ... just a first proof-of-concept

☞ interactive response time ✔

Project stack

General purpose XML-aware DB ✔
Userlevel FS (DeepFS) + DB-embedded FS ops (BaseXFS)
Stackable File System Module
File System

Joint storage for FS and DBMS



Joint storage

ID	PAR	SIZE	ATT	TYPE	TAG	TXT
1	0	724	0	0	0	
2	1	11	1	0	1	
3	2			2	2	0

Summary

- Joint storage is key
- Simplicity is key for kernel integration
- Synergies between semi-structured database and file system techniques
- Perspectives:
  - VFS+, a generic (query) interface to data

Thank you !

Melting Pot XML

Bringing File Systems and Databases One Step Closer

Christian Grün Alexander Holupirek Marc H. Scholl
DBIS Group, U Konstanz

BTW2007, Aachen, March 2007