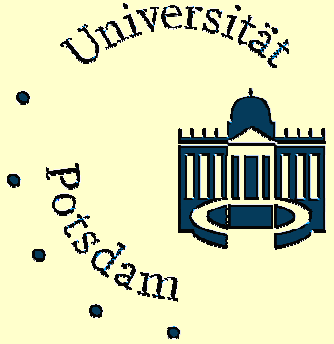


NaXDB – Pipelined XQuery Processing in a Native XML Database System

Jens Hündling

Mathias Weske, Jan Sievers

Hasso-Plattner-Institute for IT-Systems Engineering

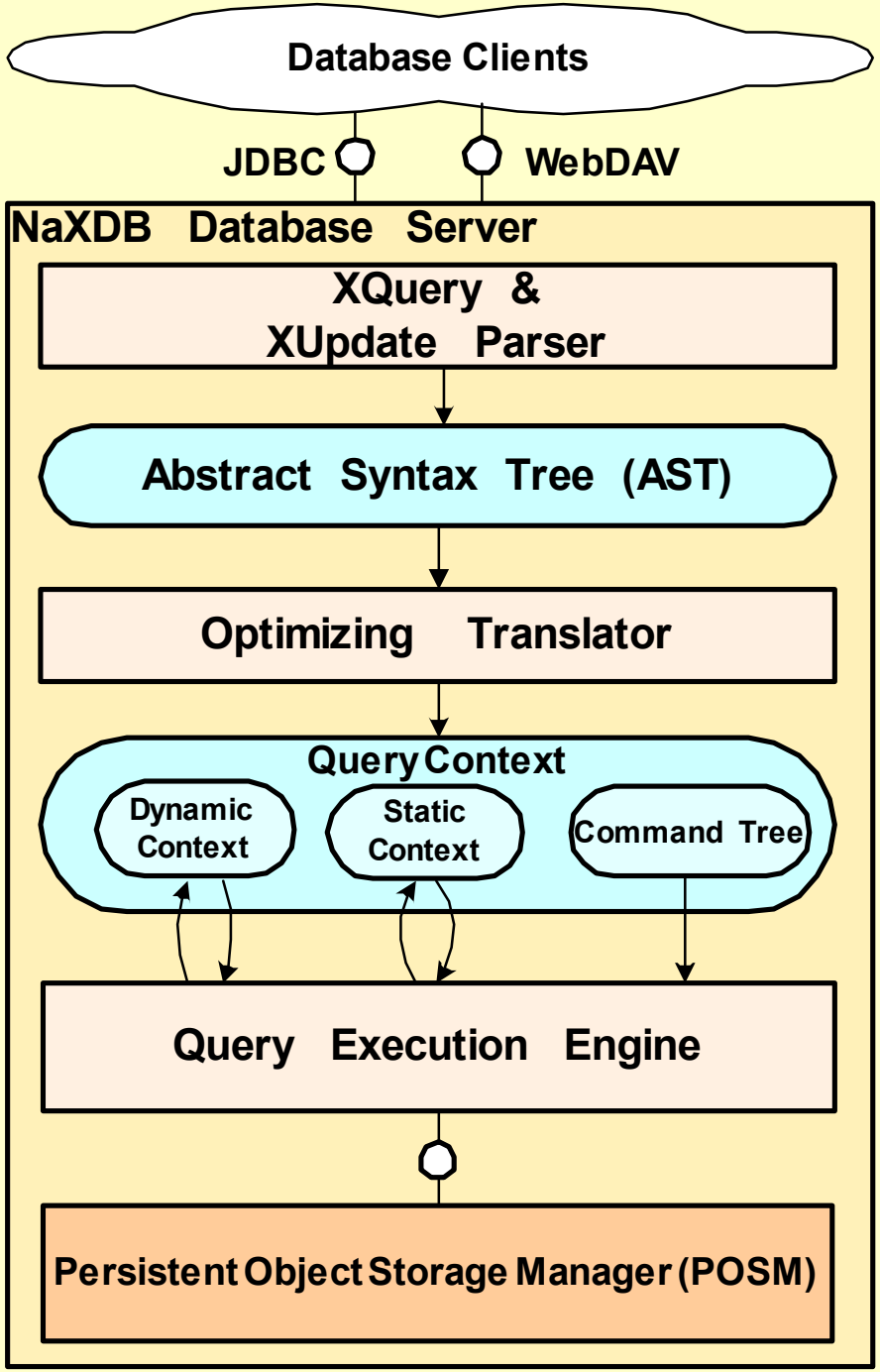
at the University of Potsdam, Germany

Second International Workshop on XQuery Implementation, Experience and Perspectives (XIME-P 2005) June 16-17, 2005, Baltimore, Maryland

The NaXDB Project

Overview

- Joint project in Germany
  - HPI, University of Potsdam
  - SAP Labs, Berlin
- Goals
  - **Na**tive **X**ML **D**ata**B**ase using MaxDB by MySQL
  - Multi-user XQuery and XUpdate
  - User Interfaces for developer
    - Eclipse Plugins, debug features, WebDAV



NaXDB Architecture

Overview

- Parsers generate
  - AST for XQuery & XUpdate
- Translator optimizes
  - Command tree
- Pipelined query processing
  - Command tree traversal
- POSM
  - Based on MaxDB by MySQL
  - Stores versioned objects

NaXDB Features

Features

- Storing native XML tree
  - Two object types
    - Structure Tree: small, linked objects (ca. 60 bytes)
    - Content: fixed or variable sized objects
  - OID's and object versions handled by POSM
- XQuery and XUpdate support
  - Path-navigation on structure objects
  - Structural changes
    - New versions of structure objects, i.e.
    - Modified links of "surrounding" objects

NaXDBs Structure Objects

Features

- Structure object for nodes
  - QName OID
  - Links (OIDs)
    - Parent, first and last child
    - Preceding and following sibling
  - Content object OID
  - Based on OIDs, managed by POSM
- Sorting
  - (if necessary) uses structure
  - Structure tree in memory

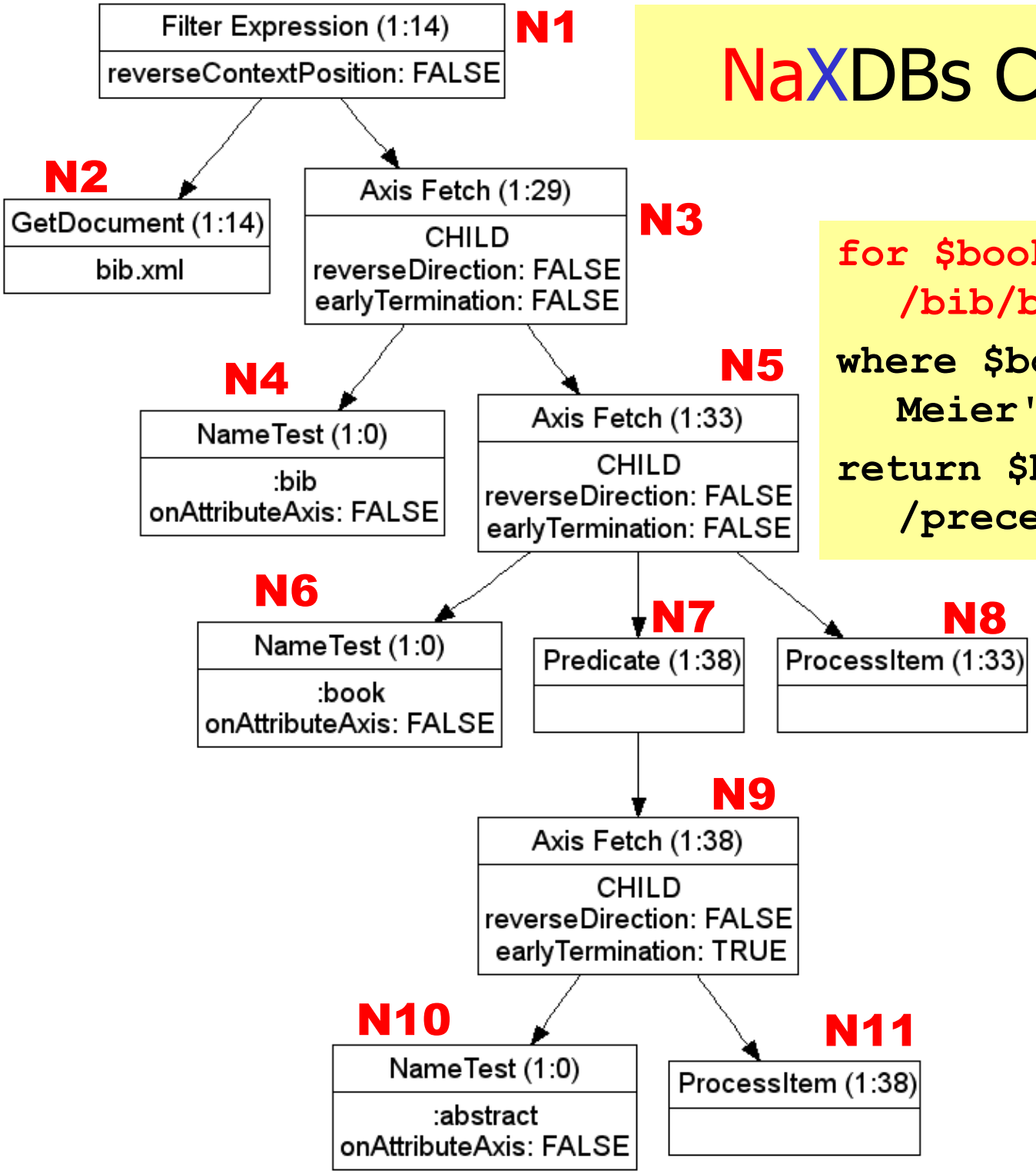
Lessons learned from NaXDB

- Advantages of nativeness
  - Sound concept
    - Navigable structure
    - Command tree traversal
    - Pipelined Query processing
  - → Debug-Feature
    - Graphical command trees, demo available (ask me ☺)
    - Valuable for developers and tuning
- Modifications
  - Versioned object allow snapshot-isolation
  - Concurrent Read/Write access

Summary & Outlook

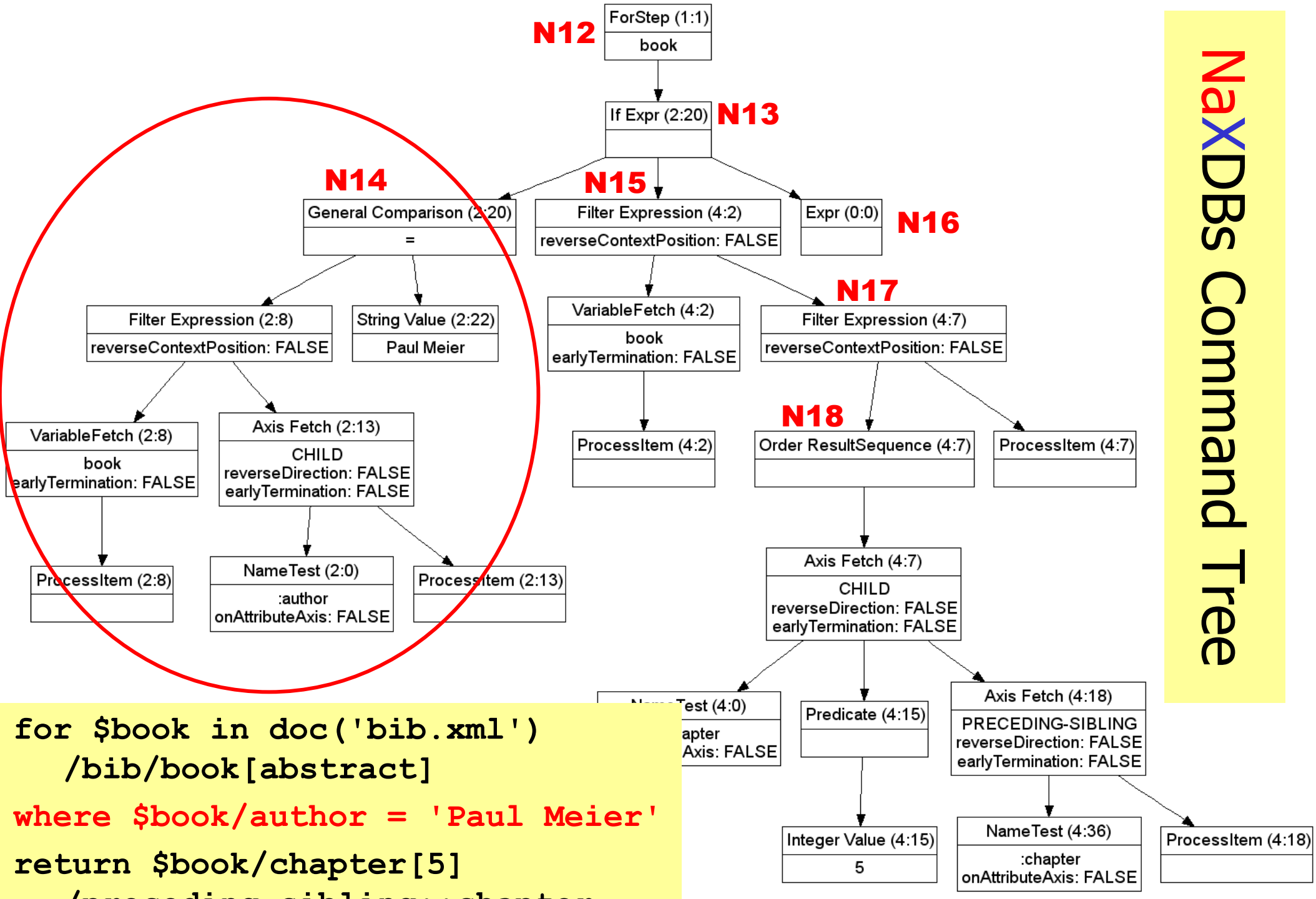
- NaXDB
  - Native XML Database System
  - Persistent storage and Multi-User
- Project Status
  - Currently on hold
  - Benchmarking is next step
- Open Issues
  - slow initial XML document load
    - Complex structure and great documents
  - XML Schema support

NaXDBs Command Tree




```
for $book in doc('bib.xml')
   /bib/book[abstract]
where $book/author = 'Paul
   Meier'
return $book/chapter[5]
   /preceding-sibling::chapter
```

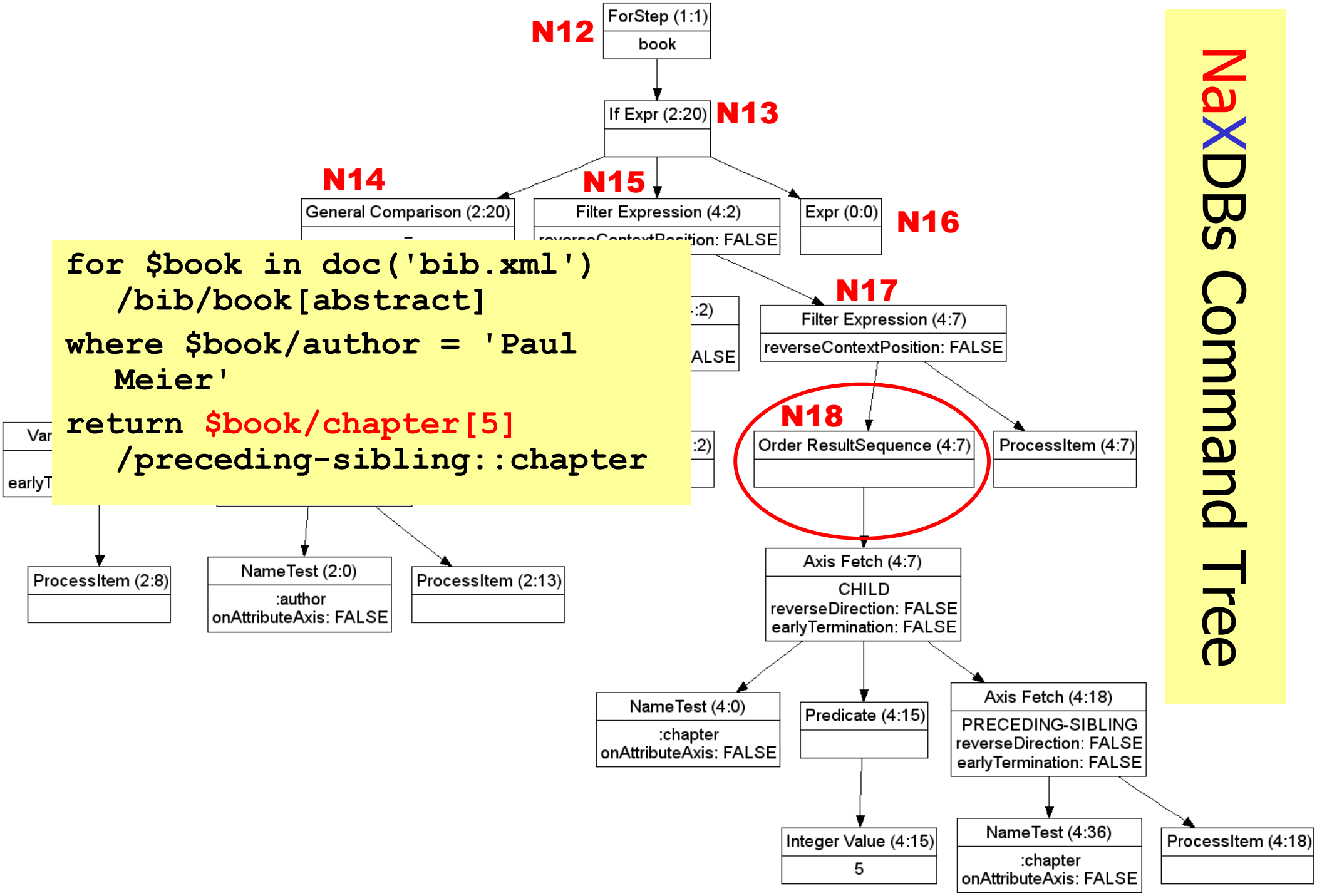
NaXDBs Command Tree




```
for $book in doc('bib.xml')
   /bib/book[abstract]
where $book/author = 'Paul Meier'
return $book/chapter[5]
   /preceding-sibling::chapter
```

NaXDBs Command Tree

```
for $book in doc('bib.xml')
   /bib/book[abstract]
where $book/author = 'Paul
   Meier'
return $book/chapter[5]
   /preceding-sibling::chapter
```



Thanks for your attention!

http://bpt.hpi.uni-potsdam.de

Business Process Technology Research Group

Prof. Dr. Mathias Weske

Email: Jens.Huendling@hpi.uni-potsdam.de