XOTclIDE

Artur Trzewik
mail@xdobry.de

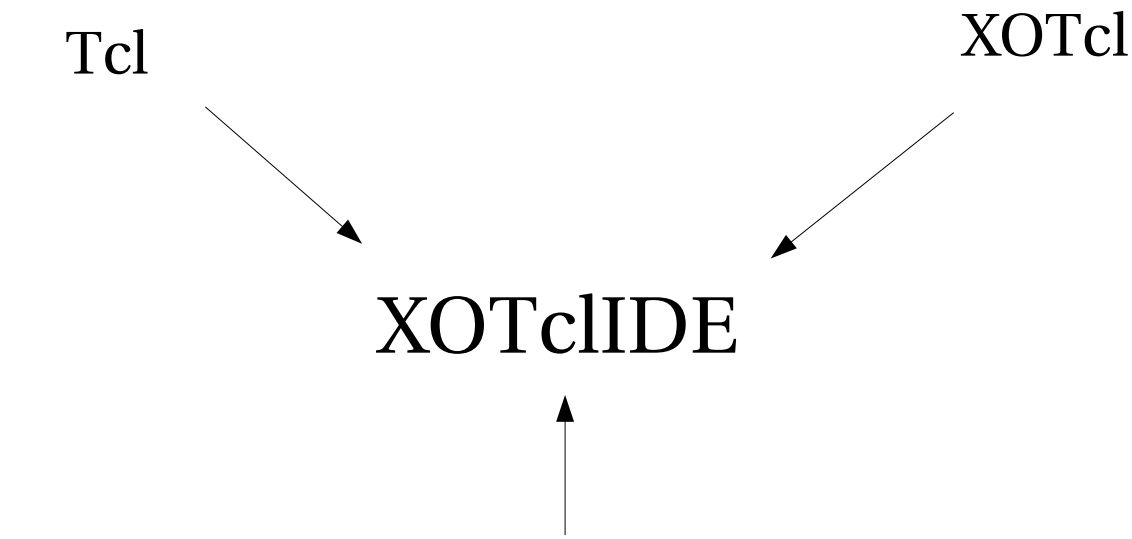
http://www.xdobry.de/xotclIDE

4. European Tcl Workshop 2003 - Nürnberg

XOTclIDE

eXtended
Object oriented
Tool
Command
Language
Integrated Interactive
Development
Environment

Background – (1)



Background – XOTcl (2.1)

- Hight flexible fast object oriented Tcl-Extension
- Based on OTcl from MIT Institute
- extended by Gustaf Neumann and Uwe Zdun
- Stable version 1.02.
- Open source project available from http://www.xotcl.org

Background – XOTcl (2.2)

Highlights:

- Multiple inheritance (method chaining)
- Three level class system
- read/write introspection
- Without protection mechanisms
- Object aggregation / Nested classes
- Per-Class and Per-Object Mixins / Filters
- Meta-data
- Assertions

Background – Smalltalk (3)

- Mature pure object system
- Simple syntax
- Mature libraries (collections)
- Implicit typed (dynamic typed)
- Reflection (Introspection)
- Block concept (similar to tcl uplevel)
- First IDE System (for example Squeak)

XOTclIDE Targets

- Suitable for big object oriented projects (more than 30.000 lines of code)
- Team-Programming
- Extreme Programming
- Integrated version control
- Additional tools

XOTclIDE is interactive (1)

Interactive mode in tcl shell

```
$ tclsh
% proc foo {} { puts "foo" }
% info body foo
 puts "foo"
% proc foo {} { puts "foo2" }
% rename foo {}
```

This IDE is not:

- Manager for script files
- Tcl Code-Editor

This IDE serve GUI for interactive work with XOTcl and Tcl

XOTclIDE is interactive (2)

- No classic development cycles“Edit, Compile, Debug”
- Classes and object will be build iterative method on method. Method definition are compiled (evaluated) at once.
- Debugger is important developing-tool

Structure Programs

XOTcl

Packages$\in$ Classes$\in$ Methods

XOTclIDE

Components$\in$ Classes$\in$ Categories$\in$ Methods

Additional comments for every level

Components

- Based on Tcl packages
- Introspection
- recreation
- unload
- Controlled initialization (special initialization methods)

Tcl-Editor

- Syntax highlighting
- Code completion (context sensitive)
- Integrated syntax checking
- Parenthesis highlighting
- Evaluation of Tcl scripts
- Wizards

Source Navigation

- Search after Method-Sender
- Search after Class-Definitions
- Search after Method-Implementors
- Search after Text

In Global, Component or Class context.

Debugging

- Two realizations
  - pure Tcl/XOTcl
  - Based on atkdebugger Tcl-Extension
- Call-Stack display
- Read and write introspection of local variables

XOTclIDE implements so called dynamic debugger

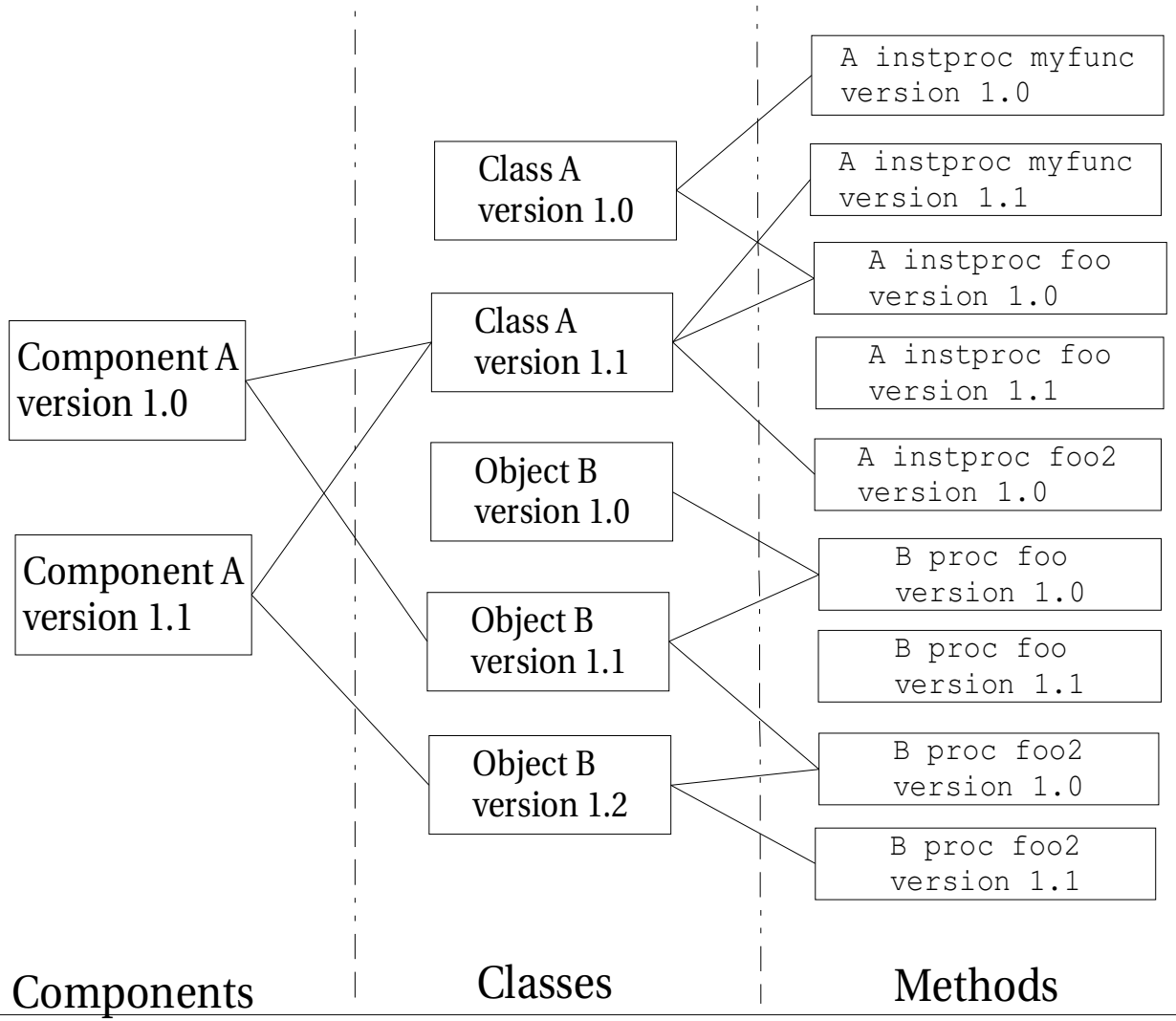
Extended Debugger

- Based on atkdebugger (C extension, need Tcl-Patch for 8.4.1)
- Breakpoints (integrated into IDE). conditional Breakpoints
- Step into / Step Over
- Program flow highlighting
- No code instrumenting (it is very fast)

Version Control (1)

- Equivalent to structure of XOTclIDE components
- Works in background. No explicit Check-Out and Check-Int
- Integrated into environment
- Persistence in relational DB (mysql, postgres, sqlite, odbc). Sophisticated SQL-Queries are possible
- supports Team-Programming

Version Control (2)



Configuration-Management

- Building groups of components (Solutions, Products)
- Releases-Versioning
- Deployment
- Changes tracking

Additional Futures

- Unit Test Framework (based on Sunit)
- HTML-Documentation Generator
- Tclkit – available
- Project importing
- Pure Tcl support
- Methodcalls-Tracker
- Coverage Analysis Browser

Outlook

- Full dynamic debugger
- Transformator ITcl -> XOTcl
- Generator for C, C++ wrappers (GUI for SWIG)
- GUI-Builder

That's all!

Any Questions ?

http://www.xdobry.de/xotclIDE