Software construction tools for KDE

Thomas Nagy
tnagy256@yahoo.fr

August 29, 2005
1. Introduction

2. Software for building KDE applications
   - Overview of the KDE build system
   - Main families
   - KDE4 expected changes

3. The Bksys framework
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4. Conclusion

5. Questions
Introduction

About the author
- Software Engineer, School of the Mines of Nantes, graduated in 2004
- Main area of interest:
  - KDissert, a mind-mapping tool for document creation
  - BKsys, a replacement for the KDE build system

Acknowledgements
- The Kdissert users
- The BKsys early adopters (Rosegarden, kio-locate, ..)
- The Akademy team
1. **Introduction**

2. **Software for building KDE applications**
   - Overview of the KDE build system
   - Main families
   - KDE4 expected changes

3. **The Bksys framework**
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4. **Conclusion**

5. **Questions**
Introduction KDE application writing

- **Tools required**
  - `make -f Makefile.cvs; configure; make; make install`
  - `Autoconf, Aclocal, Autoheader, Automake, Libtool, GNU/Make`

![Diagram of tool organization](image)

**Figure:** Organization of the tools

- **Installation:** copy files at specific locations and change permissions

Thomas Nagy  tnagy256@yahoo.fr  Software construction tools for KDE
Rules are joined in cascade

Figure: Source code compilation

Figure: Documentation compilation
Shortcomings of the current build system

- **Technical features**
  - Size of the scripts, tools needed for the compilation and overall speed
  - Addition of new compilation rules is uneasy

- **User-friendliness**
  - Reliability of the builds
  - Hard to make mistakes, easy to spot them
1 Introduction

2 Software for building KDE applications
   - Overview of the KDE build system
   - Main families
     - KDE4 expected changes

3 The Bksys framework
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4 Conclusion

5 Questions
Main families

- **Make based:** GNU/Make, NMake - unreliable, complicated and unportable
  - reliability and usability
  - portability and maintainability
- **Makefile generators:** QMake, Autotools, CMake - use macros as a language
  - still questionable reliability
  - more commands are required for compiling
  - errors are harder to debug
- **Script-based:** Unsermake, SCons, Cons - slow and memory-consuming
  - Powerful languages (python, perl) are used
  - No separation between data and logic
  - Less adapted for incremental builds (but better at distributed ones)
1 Introduction

2 **Software for building KDE applications**
   - Overview of the KDE build system
   - Main families
   - KDE4 expected changes

3 The Bksys framework
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4 Conclusion

5 Questions
KDE4 expected changes

- Changes are difficult to enforce
  - The codebase prevents a quick transition
  - The developers are not willing to change their tools

- State of the main candidates
  - Unsermake is meant to provide a smooth transition to developers hence the preferred solution for KDE4
  - QMake does not have the necessary features but CMake was proved to work and scale
  - BKsys provides a complete replacement for the KDE3 build system
1. Introduction

2. Software for building KDE applications
   - Overview of the KDE build system
   - Main families
   - KDE4 expected changes

3. The Bksys framework
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4. Conclusion

5. Questions
A full-featured replacement for building KDE applications

Target audience
- New projects
- Casual programmers

SCons-based
- Scons can be included: 57kb
- The admin/ directory weights only about 150kb

Build KDE and other kinds of programs: Wxwidgets, Qt4, ..

Main features: small, fast, reliable, and separation of the build dir
1 Introduction

2 Software for building KDE applications
   - Overview of the KDE build system
   - Main families
   - KDE4 expected changes

3 The Bksys framework
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4 Conclusion

5 Questions
Organization of the framework

- Tell the tool what to build
  - Build things from sources
  - Install files in locations
  - Give explicit dependencies
- Give rules for building
  - Tells the tool how to produce the files
  - Scan files for implicit dependencies
- Detect the system configuration
  - Find headers, libraries and configuration checks
  - Make components reusable among projects
Project setup

- Declaring targets
  - The scripting interface
    - Really complicated builds are possible
    - Brute-force parsing is made difficult
  - Pmanager, the GUI for BKsys
    - Easy to parse and to create check tools
    - Gui is provided for user-friendliness, Kdevelop integration is easy
- Adding new rules and configurations
  - Modules are python files
  - Add configuration checks
1 Introduction

2 Software for building KDE applications
   - Overview of the KDE build system
   - Main families
   - KDE4 expected changes

3 The Bksys framework
   - A full-featured replacement for building KDE applications
   - Organization of the framework
   - Further improvements

4 Conclusion

5 Questions

Thomas Nagy tnagy256@yahoo.fr

Software construction tools for KDE
Further improvements

- Overcome the autotools incompatibilities to facilitate the migration
  - The command-line handling is slightly different
  - Wrappers are provided

- More detection modules are needed
  - Creation of a library of modules
  - More configuration helpers

- Finish the GUI
  - Add and remove targets and sources
  - Use of drag and drop
BKsys targets a particular kind of application development
It needs a stronger userbase to evolve faster
Questions ?
References

- **Build systems reviews**
  - Article on build systems:
    http://freshmeat.net/articles/view/1715/
  - Another review:
    http://www.a-a-p.org/tools_build.html

- **Software construction tools**
  - CMake page http://www.cmake.org/
  - SCons, the foundation of BKsys
    http://www.scons.org/
  - BKsys page
    http://freehackers.org/~tnagy/bksys.html
References (2)

Figure: A book on Autotools