Design and Academic Use of the Haiku Operating System

François Revol
LCIS (CTSYS Team), LIG (ADELE Team)
François.Revol@imag.fr

1. Introduction
- Operating System
  - OpenSource
  - MIT licence
- Desktop-focused
- Designed from scratch
  - But leverages more than 40 existing OpenSource projects
  - Coherency from kernel to GUI
  - KiSS: Keep It Smart & Simple
- Started 2001 as OpenBeOS
- First alpha release sept. 2009
  - 10400 downloads
- 45 committers, >90 contributors

2. Missing & Reused parts
- Still in development
  - Live sample of full OS evolution
  - But still human size
- More appealing practical work
  - Google Summer of Code
- PPC & ARM port, Gallium3D…
- Missing parts
  - Leaves room for experimentation
  - IPv6, Java, power management, full multimedia
- Reused software projects
  - Demonstrates integration of foreign code
  - FFmpeg, FreeType, MESA…

3. Object-Oriented C++ API
- Built on top of POSIX and native C calls
- Coherent hierarchy of C++ classes
- Organized in Kits
  - Support, Storage,
  - Network, Mail,
  - App, Interface,
  - Media
- Translation (format conversion)
  - Add-ons (png, svg, SANE, gocr…)
  - Demonstrates OO concepts
    - multiple inheritance, FBC…

4. Clean and Coherent Design
- Multithreaded Applications
  - BApplication, BWindow are BLooper
  - Smoother experience
- Ported software can serialize
- POSIX layer in libroot.so
- C++ API (Kits) in libbe.so…
- Multithreaded Servers implement Kits (App, Media…)
  - Based on Add-ons
  - Graphics "Accelerators"
  - MediaNodes
  - input_server Devices, Filters and Methods…

5. Kernel & Drivers
- VFS & Caches
  - Axel Dörfler, Entwicklung eines Dateisystemcaches mit Unterstützung von Methoden aus der künstlichen Intelligenz, Master Thesis at Universität Osnabrück
  - FreeBSD abstraction layer helps network drivers ports
  - WiFi stack prototype
  - Colin Günther (Ongoing Master Thesis)

6. Prototyping with the GUI
- University of Auckland
  - AIMHelp: generating help for GUI applications automatically
  - Multi-Platform Document-Oriented GUIs
  - Auckland Layout Model
    - Uses Linear Programming Solver
    - C++ (Haiku), Java, .NET
    - http://aucklandlayout.sourceforge.net/
  - A More Manageable Multi-Window Interface
    - Stack & Tile
  - Google Summer of Code
    - Sub-pixel antialiased rendering

7. Filesystem
- OpenBFS
  - Classical UNIX constructs (modules) with new techniques
  - B-Trees, block_tuns (= extents)
  - Typed & Indexable Extended Attributes
    - Fast Search
    - Line Query
    - Base for Semantic Desktop
- VFS supports other types
  - ISO9660, NTFS, ext3, NFS…
  - Write and xaotr overlays

8. Extended Attributes: Need for Interoperability
- Many OSES use Extended Attributes
  - But semantics and API differ
- Existing mapping schemes do not cooperate
- Need for Idempotent Scheme
- Proposal: A Universal Extended Attribute Namespace
  - Map global namespace in native namespace and vice-versa

9. Conclusion
- Clean design makes it easier to study
- Classical APIs allows code reuse
- Native API allows new designs
- Already used for research
- Haiku is a nice testbed for research and teaching

Research and teaching needs, Clean design and simple means,
Haiku fulfills.

Project Links
- http://www.haiku-os.org/
- http://dev.haiku-os.org/ (bugs & svn)
- irc://irc.freenode.org/haiku
- http://ports.haiku-files.org/
- http://www.osdrawer.net/ (native applications)