Ateneo de Manila University

Software Development Perestroika

William Emmanuel S. Yu
December 2001
http://sysads.ateneo.net/wyu/
wyy@admu.edu.ph
Open Source Software (OSS) Development

★ a new cooperative model for software development
  – benefits both the software developers who aim to capitalize on their software development efforts and other developers
  – the open source development model improves the quality of these development tools
  – spend time developing the software instead of spending time developing the tools
★ two roads to take:
  – develop potentially profit generating close source proprietary software
  – develop potentially humankind benefiting open source public domain software
★ place no restrictions on the end products generated by these tools
★ it is the choice of the developer
Open Source Software (OSS) Development

★ provides commercial grade and standards compliant tools
  ★ being used by software development houses, companies, schools and universities around the world
  ★ commercial products exist that are developed with OSS tools
    ★ CISCO IOS (developed with GCC)
    ★ Stronghold (Apache), Sendmail and MySQL
    ★ Netscreen and some other commercial firewall appliances
★ provides greater flexibility and customizability
★ enables cross platform development
★ and many more....

December 2001
OSS Development Tools

- Compilers and Interpreters
- Libraries and Toolkits
- Editors and Integrated Development Environments (IDE)
- Debuggers
- Source Code Management Tools
- Collaborative Development Tools
Compilers and Interpreters

- artificial intelligence
  - lisp, haskell

- web applications
  - perl, python
  - PHP Hypertext Preprocessor (PHP)
  - Active Server Pages (ASP)

- embedded applications
  - C, assembly

- large scale and complex applications
  - C++, Java, Delphi

- systems/network glue coding
  - perl, shell scripting, C
Libraries and Toolkits

- Standard System Libraries
  - GNU C Library
  - C++ Standard Template Library (STL)
  - Math, Graphics and others
- Graphical User Interface (GUI) Toolkits
  - Gimp Toolkit (GTK)/GNOME
  - Qt/KDE
  - X/Motif
  - Kylix/Delphi
- Application Specific Libraries
  - Lightweight Directory Access Protocol (LDAP)
  - Pluggable Authentication Modules (PAM)
- Common Object Request Broker Architecture (CORBA)

★ Embedded Systems Development Kits
  - GNU Compiler Collection/Cygnus Developer Tools
  - Qt Palmtop Environment (QPE)
  - M-lib, newlib, ...

★ and definitely more ...
Integrated Development Environments (IDE)

- syntax highlighting
- ability to integrate build tools
- integrated debugger
- hot key and shortcuts

- rhide - is a turbo/borland like IDE
- kdevelop - IDE from the KDE development team
- emacs - everything but the kitchen sink
- glade - RAD GUI builder for gtk/gnome
- anjuta - IDE specifically for writing C/C++ gtk/gnome, bonobo and console applications
- Kylix/Delphi - a RAD tool for developing cross platform delphi applications
Debuggers

★ GNU Debugger (gdb, xxgdb) - command line and X windows debugger from the GNU Free Software Foundation

★ Java Debugger (jdb) - bundled command line debugger for Java

★ most interpreters come with built-in debuggers

★ most IDEs come with built-in debugger or simple make calls to a command line debugger
Source Code Management Tools

- GNU make - a tool which controls the generation of executables and other non-source files of a program from the program’s source files
- GNU autoconf - is an extensible package of m4 macros that produce shell scripts to automatically configure software source code package
- GNU automake - is a tool for automatically generating ‘Makefile.in’ files compliant with the GNU Coding Standards which is required by autoconf
- Concurrent Versions System (CVS) - a system used to maintain and manage multiple versions of a software source tree
Collaborative Development Tools

- Sourceforge - is an online Web application to assist in the development of Open Source Software by providing Bug tracking, Project Management, Forums, Mailing Lists, Surveys, Software Map, File Releases, Top Projects, Site Docs, Project News, Site News, Snippet Library, Registration System, and Individual Account Maintenance

- Bugzilla - is a bug ticketing system

- Mailing lists - is the traditional way of keeping correspondence and distributing software amongst developers in a OSS project
Frequently Asked Questions

★ If I use OSS tools do I have to release my code as open source?
★ Do commercial software vendors support Linux? or do they have Linux support for their software?
★ Can I still use conventional software engineering methodologies?
★ Can I use OSS tools to develop applications for the Windows platform? or any other platform?
★ Do OSS solutions scale?
★ Can Linux be used in the heterogenous environment?
★ Is WYSIWYG development or rapid applications development (RAD) possible with OSS tools?
★ What is a good introductory programming language to learn?
If I use OSS tools do I have to release my code as open source?

★ technically NO, but we would like you to!

★ most OSS development tools are release under a not so restrictive licenses such as:

- Library GPL (GCC and other libraries)
- Artistic License (Perl and friends)

★ dependent on the licensing restrictions of the development tool
Do commercial software vendors support Linux? or do they have Linux support for their software?

★ Most major commercial software vendors do have Linux support for their products and their technologies:

– Oracle - have Linux versions for their popular database application suites. Oracle 8 until Oracle 9i have Linux versions available

– IBM - has ported most of the applications to Linux and has added Linux support to most of their technologies including the S/390 series servers. IBM has ported DB2, Websphere and many other technologies to Linux.

– SUN - has helped promote Java support on the Linux platform. Linux is now included as one of the Java’s major platforms. SUN has also released Star Office under the GNU GPL. It has also released its Solaris operating system binaries for free.

★ these companies also provide support solutions for Linux-based products
Can I still use conventional software engineering methodologies?

★ Of course you can!

★ OSS software tools tend to be:
  – free
  – well tested and typically well documented
  – upgrades, patches and optimizations are quickly made available
  – large pool of developers

★ commercial software tools tend to be:
  – system tailored and optimized
  – come bundled in a single simple to install package
  – commercial support is available (?)
Can I use OSS tools to develop applications for the Windows platform? or any other platform?

★ Of course you can!

★ OSS developer tools for the Microsoft platforms are also available:
  – djgpp - a complete 32-bit C/C++ development system for Intel x86 PCs for MS-DOS
  – cygwin - is a UNIX environment for Windows that enables a developer to write Win32 console or GUI applications that make use of the standard Microsoft Win32 API and/or the Cygwin API
  – GNU Compiler Collection - a suite of compilers for C, C++, Java and others
Do OSS solutions scale?

★ dependent on the solution’s implementation

★ but, it can!

★ OSS tools for high availability and load balancing:
  – beowulf clustering - for scientific and compute intensive applications
  – Linux Virtual Server (LVS) - load balancing and failover support
  – FailSafe - SGI architecture for high availability systems
  – Heartbeat/Fake - OSS combination for detecting a downed server and replacing it in the network
Can Linux be used in the heterogenous environments?

⋆ YES! ang kulit!

⋆ complaint to standards
  – POSIX compliant OS with a Unix-like API
  – GCC tools are standards compliant such as ANSI C and ISO C++
  – uses TCP/IP as it’s main mode of communications
Is WYSIWYG development or rapid applications development (RAD) possible with OSS tools?

★ and another big YES!

★ there are open source products that support WYSIWYG and RAD software development:
  – Kylix - source compatible delphi RAD tool from Borland
  – Glade - GTK RAD tool from the GNOME project
  – Kdevelop - Qt RAD tool from the KDE project
What is a good introductory programming language to learn?

☆ Java
- early object oriented programming (OOP) training
- strongly typed
- self-contained
- standard
- associations can be made with real objects

☆ Pascal
- clean structured programming language
- strongly typed
- simple
Copyright © 2000-2001 by William Emmanuel S. Yu. This material may be distributed only subject to the terms and conditions set forth in the Open Content License, v1.0 or later (the latest version is presently available at http://opencontent.org/opl.shtml).