BRL-CAD Users Group Meeting 2002

Friday, October 25, 2002

BRL-CAD & Mac OS X

Christopher Sean Morrison
morrison@arl.army.mil

Quantum Research International, Inc.
BRL-CAD & Mac OS X

Topics:

• Apple Computer, Inc’s new platform
• Architecture considerations
• BRL-CAD and UNIX
• The actual port to Mac OS X
• Performance issues
• Future direction
Apple Computer, Inc.

“Mac OS X is a modern operating system that combines the power and stability of UNIX with the simplicity and elegance of the Macintosh.”

Apple is the largest UNIX-based platform vendor on the planet.
Mac OS X

- New BSD-based operating system
- Open-source kernel (Darwin), some libraries and APIs as well
- Standards compliance
  - OpenGL, POSIX, MPEG4, Java2, IPv6, LDAPv3, IPSec, SSL, SSH2, …
- Interoperability with Windows and UNIX
- Popular commercial software available
  - Word, Excel, PowerPoint, Internet Explorer, Photoshop, Mathematica, Pagemaker, InDesign, …
Multiprocessing & More

• Symmetric multiprocessing architecture available
• Altivec vector pipeline (aka Velocity Engine™)
Darwin

• Core open-source operating system
• X Server support
  – Xdarwin and XonX
• POSIX standard compliance supported
  – Threading (POSIX and Mach threads available)
  – Shell
  – sysctl interface
• OpenGL support
  – hardware accelerated, double-buffered windows, per-pixel alpha channel & fade control, and more…
  – Quartz extreme
# Darwin

<table>
<thead>
<tr>
<th>Classic</th>
<th>Carbon</th>
<th>Cocoa</th>
<th>Command Line</th>
<th>GTK &amp; KDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platinum</td>
<td>Aqua</td>
<td>Quartz</td>
<td>Terminal Window</td>
<td>Window Manager</td>
</tr>
<tr>
<td>QuickDraw</td>
<td>Quartz</td>
<td></td>
<td>Shell (csh, bash, etc.)</td>
<td>XDarwin</td>
</tr>
</tbody>
</table>

**Darwin (Core Operating System)**

Based on BSD 4.4 - Mach 3.0 Micro Kernel
BRL-CAD & UNIX

• BRL-CAD is designed to work best in a UNIX-based environment
• Many compact and well-defined tools that perform particular tasks
• Presently 397 tools, utilities, commands, …
• Commands may be chained together for flexible usage
Why Mac OS X?

- Open standards compliance
- It is UNIX based
  - It has a POSIX command line interface and tools
- Apple has a respectable history of reliability and “good design” in both software and hardware

- It runs Microsoft Office, Photoshop, Mathematica, …

- It’s the largest UNIX-based platform
The Port

• Bulk of work (90%) was done in less than half an hour
• Iterative and incremental approach
• Files modified:
  – sh/machinetype.sh
  – Cakefile.defs
  – h/conf.h
  – h/machine.h
  – libbu/parallel.c
  – libfb/…
  – libdm/…
Iterative and Incremental Process

- Get `sh/machinetype.sh` to work
- Run `setup.sh` successfully
  - Get cake to work
- Stub a base configuration into `Cakefile.defs`
- Add architecture details to `h/machine.h`
- Compile, modify `h/conf.h`, and repeat …
  until all errors and (most) warnings are eliminated
Problems Encountered

- Adding the architecture piece-wise unveiled assumptions and dependencies that were not expected
  - E.g. X Windows support on UNIX
- Subtle bugs (some nasty) that were not evident on other platforms emerged
- Those problems have been fixed
Extra Functionality Needed

- **Multiple processor support (SMP) working**
  - Modified `bu_avail_cpus()` in `libbu/parallel.c`
- **Added something more functional than the debug framebuffer**
  - Added X Windows support configuration options
  - Add OpenGL support configuration options
- **Other ideas for later …**
Performance

• Running the BRL-CAD benchmark
  – bench/run.sh actually runs the test
    • results stored in the file named “summary”
  – bench/try.sh invokes the benchmark run interactively, opening up a window per image
    • For the time being, you will need X Windows to be installed

• Interpreting the results
  – Apple has done impressive optimizations
    • Uses the open source GNU Compiler Collection (GCC 3.1)
  – The hardware takes significant advantage of L2 cache available
    • G4 500’s have 1MB unified L2 cache per chip
    • G4 800 and 1000’s have 256KB L2 cache and 1MB L3 cache
Benchmark Results

average rays/second (rtfm)

<table>
<thead>
<tr>
<th></th>
<th>G4 500</th>
<th>Dual G4 500</th>
<th>G4 800</th>
</tr>
</thead>
<tbody>
<tr>
<td>20000</td>
<td>36354</td>
<td>83633</td>
<td>40722</td>
</tr>
<tr>
<td>40000</td>
<td>36177</td>
<td>83100</td>
<td>40430</td>
</tr>
<tr>
<td>60000</td>
<td>36495</td>
<td>82675</td>
<td>40490</td>
</tr>
<tr>
<td>80000</td>
<td>36631</td>
<td>82905</td>
<td>40456</td>
</tr>
<tr>
<td>100000</td>
<td>36525</td>
<td>83169</td>
<td>40440</td>
</tr>
</tbody>
</table>
Demo
Comparison to Linux on PPC

- Yellow Dog Linux 2.3 with default install gave approximately 25% slower runtime performance.
- Compiles significantly faster than Darwin:
  - 25 minutes as opposed to 1.5 hours.
- Hints that Apple has better compiler optimizations and perhaps better run-time libraries.
Future work

- Use Project Builder compilation environment coupled with GNU’s autotools
- Integrate Altivec vector pipeline support into math operations

- Relinquish dependancy on X
  - Create libfb and libdm interface for Aqua
  - Use AquaTk
Use Aqua Interface

<table>
<thead>
<tr>
<th>Classic</th>
<th>Carbon</th>
<th>Cocoa</th>
<th>Command Line</th>
<th>GTK &amp; KDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platinum</td>
<td>Aqua</td>
<td>Quartz</td>
<td>Terminal Window</td>
<td>Window Manager</td>
</tr>
<tr>
<td>QuickDraw</td>
<td>Quartz</td>
<td></td>
<td>Shell (csh, bash, etc.)</td>
<td>XDarwin</td>
</tr>
</tbody>
</table>

**Darwin (Core Operating System)**
Based on BSD 4.4 - Mach 3.0 Micro Kernel