CRAY The Supercomputer Company

3rd Party Software

Gail Alverson
August 5, 2005

CRAY The Supercomputer Company

Outline

- Part 1 – Case Study: Story of an engineer
  - What's my software engineering history?
  - Example of Cray software stack/software types
- Part 2 – 3rd party software
  - What is it?
  - Why use it – what are the benefits?
  - And what are the costs?
- References

CRAY The Supercomputer Company

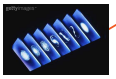
My SW Engineering roles 1991 - present

- **Research engineer, libraries and debugger**
  - Designed and implemented components
  - Worked with a small team
- **Project lead/Software engineer, libraries and tools**
  - Added responsibility for broader direction of the products
  - And project assignments/schedule
- **Manager, programming environments**
  - Stepped back from coding to focus on design and project planning
  - More management of engineers
  - More interaction with vendors, customers, and corporate roadmap
- **Senior Manager, OS and PE components**
  - Project direction, estimation, planning and execution
  - Integration work and release work
  - Engineer management, including their career growth

CRAY The Supercomputer Company

Cray Inc.

Focus on systems to solve huge computationally intense problems


Government-Classified
Government-Research
Weather/Environmental
Automotive
Life Sciences
Aerospace
Academic Research


Cray Solutions for:
Security
Design Engineering
Environmental
Life Sciences
Data Management/SAN

CRAY The Supercomputer Company

Boeing has a Cray X1E

- X1E is a vector supercomputer, 18GF/s vector nodes
- Boeing uses it on an in-house aerospace code - Computational Fluid Dynamics
- The X1E allows Boeing to accomplish work that used to take a week, in a single day

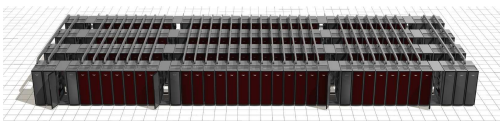
BOEING

CRAY The Supercomputer Company

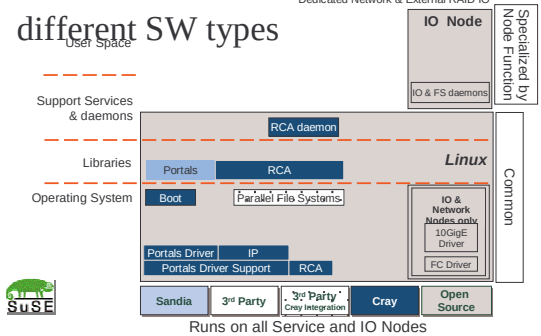
I worked most recently on the Red Storm system

Massively parallel processing supercomputer system used for analysis and stewardship of nuclear weapons

- $93M contracted with Sandia National Labs
- Key system characteristics
  - 10,000 AMD 2 GHz processors
  - High bandwidth mesh based custom interconnect
  - High performance I/O subsystem
  - Fault tolerant



Red Storm Software Stack illustrates different SW types



Runs on all Service and IO Nodes

What is 3rd party software?

- Software produced by someone/organization outside your company, for a broad set of consumers
- *Brainstorm examples*:
  - Linux OS
  - PGI compilers
  - PBS batch submission system
  - ACML scientific libraries
  - MPICH MPI library
  - JUnit
  - Eclipse
  -




The Portland Group™

Why use 3rd party software?

- *Brainstorm motivations:*
  - Function exists, why reinvent the wheel
  - Less development time
  - Less development cost
  - Requested by customers
  - Others have expertise
  - Others contribute to the support and evolution
  -

How can we use it?

1. As a tool in the product development process:
   - *Brainstorm examples:*

2. As part of a product being shipped:
   - *Brainstorm examples:*

We can try to classify 3rd party software

- **Commercial software**
  - Developed, distributed and maintained under a licensing agreement with a vendor, generally for a fee
- **Free software/Open source software**
  - Software that one may use, copy, modify, redistribute for free, although certain conditions/licenses may apply
  - Copyright is retained by the author
- **Public domain software**
  - Total absence of copyright protection
  - Anyone can copy, modify or use it in any way they wish
- Others: **freeware, shareware**, … [tools]

The hidden costs of using 3rd party software [in your product]

1. **Be very aware of the intellectual property (IP) rights associated with the software, such as:**
   - ***Copyright*** – protect an expression of an idea on a medium
   - ***Patent*** – protect an innovation
   - ***Trade secret*** – protect a concept/idea/innovation that is not generally known and has value
   - **License** – allow access to otherwise protected software

Ask:

- Do you have the right to use the software?
- Is it important to protect some proprietary code (IP) you're adding?
- Do the IP rights allow you to do this?

Case Study: License

- Require a strcpy() function for your product's proprietary libc
- Consider a BSD libc routine and a glibc routine
  - BSD license – *generous*
    - *No requirement to distribute your modified source code*
    - Source redistribution requires copyright notices and disclaimers
    - Binary distribution requires copyright notice and disclaimers in the documentation.
  - GPL license – *infectious*
    - *Source must be made available to customers who request it*
    - *Your modifications (derivative work) must be included with the source*
    - LGPL, license for libraries is a middle ground

The hidden costs of 3rd party software (2)

2. **Take into account integration costs**
   - Rarely does the software do exactly what you need it to
   - Typically there is some integration that must occur, and some expertise that must be built inhouse
3. **Consider whether additional testing and/or robustification is needed**
   - You're ultimately responsible
4. **Don't forget to plan for updates**
   - How will you incorporate updates of the 3rd party software into your version of it?

Case Study: PathScale

PathScale

Compiler product is based on SGI open source code-GPL

Ported the code to a new architecture and enhanced its performance

- PathScale has built a strong knowledge of the Open Source code and maintains it in-house
- They distribute the Open Source components
- And engineered the PathScale proprietary code to avoid the GPL code (independent and separate works) and thus does not distribute it

My questions to you

- What 3rd party software did you use in your product?
- Are there any requirements of your company [you] by doing this?

Summary

- 3rd party software is good!
- 3rd party software, in particular those used in your company products, has costs and/or restrictions that must be understood before committing to their use

References

- http://uwsg.ucs.indiana.edu/usail/software/third-party.html/
- http://www.opensource.org/
- http://www.gnu.org
- http://www.pathscale.com/