What's Next for the U.S. Department of Defense (DoD) and Open Source Software (OSS)?

David A. Wheeler

July 23, 2007

This presentation contains the views of the author and does not indicate endorsement by IDA, the U.S. government, or the U.S. Department of Defense.

What is Open Source Software (OSS)?

- **OSS: software licensed to users with these freedoms:**
  - **to run the program for any purpose,**
  - **to study and modify the program, and**
  - **to freely redistribute copies of either the original or modified program (without royalties, etc.)**
- **Synonyms: libre software, Free software*, FOSS, FLOSS**
- **Antonyms: proprietary software, closed software**
- **Widely used; OSS #1 or #2 in many markets**
  - **"… plays a more critical role in the DoD than has generally been recognized." [MITRE 2003]**
- ***Not* non-commercial**

* The term "Free software" sometimes means OSS, and sometimes instead means "no charge"

Warning: Fuzzy Crystal Ball

- **“It’s hard to make predictions, especially about the future” - Niels Bohr ((reportedly) Yogi Berra)**
  - **Some of this presentation is very speculative**
- **“The best way to predict the future is to invent it.” - Alan Kay**
  - **By thinking about the future, maybe we can speed the good & avoid the worst**

What's Next?

1. Existing OSS Trends Accelerate
2. Increased OSS support business
3. DoD will slowly start more OSS projects
4. General-purpose High Assurance (HA) Components Developed as OSS
5. More OSS-like gated communities
6. More guidance
7. Net-centricity/Service-Oriented Architecture (SOA) =>OSS on Servers
8. Clients move to Web applications & virtualization
9. Speculation: Risk of International Standards Process Meltdown

Selected History of OSS in DoD

- 1981-1983: DARPA funds BSD TCP/IP implementation
  - Mar 1982: TCP/IP host-to-host DoD standard by Jan 1983
- 1990/1992: AF funds GNAT Ada (validated 1995)
- 2000: NSA releases SELinux
- 2002: MITRE survey; OSS already common in DoD
- 2003: "DoD in OSS" policy memo: OSS fine!
- 2004: U.S. Federal policy memo (same)
- 2006: Open Technology Development (OTD) Roadmap:
  - Open Standards & Interfaces
  - Open Source Software & Designs
  - Collaborative & distributed online tools
  - Technological agility
- 2007: Navy memo: *OSS is commercial item/COTS*

1. Existing OSS Trends Accelerate

- Increased use of existing OSS (components & systems)
  - More flexibility (modifiable) & often lowers TCO
    - Historically flexibility is a *key* military advantage
  - Successes are lowering perceived risk, incr. knowledge
  - Contractors unwilling to ever use OSS become unable to compete against smarter competitors
- Legal/GPL allergies slowly subside
  - As technologists & managers train their lawyers (!)
  - Proprietary vendors' efforts to forbid competition through fear losing effectiveness – sky isn't falling
  - Examine when appropriate & when not, instead of a visceral fear of the different
- Increasingly realize: OSS *is* commercial item, COTS
  - Per U.S. Law & FAR; ramifications getting realized

See "Commercial is not the opposite of FLOSS" www.dwheeler.comb

Nearly all OSS are Commercial items / COTS

- ***Nearly all OSS are commercial items, & if extant, COTS***
- **U.S. Law (41 USC 403) & Federal Acquisition Regulation (FAR) prefer commercial items (inc. COTS) and NDI:**
  - **Agencies must "(a) Conduct market research to determine [if] commercial items or nondevelopmental items are available … (b) Acquire [them] when… available … (c) Require prime contractors and subcontractors at all tiers to incorporate, to the maximum extent practicable, [them] as components..."**
  - **Commercial item is "(1) Any item, other than real property, that is of a type customarily used by the general public or by non-governmental entities for purposes [not unique to a government], and (i) Has been sold, leased, or licensed to the general public; or (ii) Has been offered for sale, lease, or license to the general public... (3) [Above with] (i) Modifications of a type customarily available in the commercial marketplace; or (ii) Minor modifications… made to meet Federal Government requirements..."**
  - **True for nearly all off-the-shelf (OTS) OSS; OSS is commercial item/COTS**
- **OSS projects usually seek improvements = financial gain**
  - **U.S. Code Title 17, section 101 defines "financial gain" as including "receipt, or expectation of receipt, of anything of value, including the receipt of other copyrighted works."**
- **Many OSS projects supported by commercial companies**
  - **IBM, Sun, Red Hat, Novell, Microsoft (WiX, IronPython, SFU, Codeplex site)**
- **Often developers paid (2004: 37K/38K Linux changes)**
- **OSS licenses and projects approve of commercial support**
- **Use COTS/NDI because *users share costs* – OSS does!**

DoD already has guidance on commercial items; just apply to OSS

- **DoD AT&L's "Commercial Item Handbook"**
  - **"the Government should be able to choose from a range of supplies or services providing the best value"**
  - **"a product does not have to be developed at private expense to be commercial. Even if the Government has paid for its development, or... has a military origin... *who paid for development... [and] the offered price is not part of the commercial item determination.*"**
  - **"An item is commercial because of the supply or service itself, not because of who provides the item... a Government source... can be governed by a FAR Part 12... even if it does not provide those services in a commercial market"**
  - **"Buyers increase their leverage when... unique requirements or specifications are minimized... existing commercial distribution systems are used, and so forth."**

2. Increased OSS support business (not necessarily more businesses)

- Major DoD integrators often integrate components ***others*** support/indemnify; will often do same with OSS
  - Easier than coordinating integrators; "honest broker"
- Already: Red Hat, IBM, devIS, MySQL, AdaCore, ...
- More OSS start-ups followed by consolidations
  - Surviving long delays in acquisition process is key
  - Focus & first-mover advantages
  - OSS advantages to developer companies: Lower entry & sustainment costs, product cost advantages vs. proprietary, more flexible product
  - OSS challenges to companies: Smaller margins, easier customer & developer mobility=>lost trust disastrous
  - Mistakes quickly fatal to the undercapitalized, gobbled up by another

3. DoD will slowly start more OSS projects (inc. major additions)

- U.S. Gov't has already: GNAT, SELinux, Expect, EZRO, VistA
- Potential: Commercial capabilities/maintenance, flexibility, GOTS-like fit, low cost->affordable wide deployment
- Seed standard: *successful stnds are OSS*; "executable spec"
- Some areas more likely that others
  - Earlier: research, existing GOTS, security, collaboration mechanisms (e.g., CMS), specialized interface standards
  - Later: Larger support tasks (e.g., accounting, tracking)
- Suggestion: Research "defaults to FLOSS" (MIT/X11?)
- Slowed for a variety of reasons
  - Difficulty in getting parties together, who funds what
  - Opposition from organizations who could extract more $ from the DoD; causes slowness & "going under ground"
- DoD necessary distinctives will *not* be OSS... but much isn't

Interesting Potential OSS areas

- **Modify SCM systems so can record/prove provenance**
  - **Some OSS projects alrady use crypto keys (e.g., Debian)**
  - **But often can't record/prove far back into supply chain**
  - **Supports distributed SCM, handles subverted repository**
  - **DoD needs internally, *and* desirable for suppliers**
- **High assurance infrastructure components as OSS...**

See “High Assurance and FLOSS” www.dwheeler.com

4. General-purpose High Assurance (HA) Components Developed as OSS

- Many tools for developing HA already OSS
- You'd think most HA developed as OSS
  - Mathematicians require peer review of proof, proofs often initially wrong [De Millo,Lipton,Perlis]
  - Many voting machine proposals require source code
  - OSS often *very* good in medium assurance vs. proprietary
  - How can you write HA, without reading lots HA sw first?
- Currently HA often not OSS – expect slow transition
  - Military-unique/classified – fine, can't reveal/be OSS
  - Infrastructure HA – often a *good* case for OSS (wide use)
  - HA developers/customers very conservative, so rarely apply “new” approaches like OSS... yet
  - Funding OSS may be resisted by current suppliers
- Lots of potential for HA OSS; will be a learning process

Many OSS tools support high assurance development

- Formal methods: Community Z tools (CZT) , ZETA, ProofPower, Overture, ACL2, PVS, Coq, E, Otter/MACE, PTTP, Isabelle, HOL4, HOL Light, Gandalf, Maude Sufficient Completeness Checker, KeY, RODIN, Hybrid Logics Model Checker, Spin, NuSMV 2, BLAST, Java PathFinder, SATABS, DiVinE, Splint (as LCLint), ...
- Analysis implementation: Common LISP (GNU Common LISP (GCL), CMUCL, GNU CLISP), Scheme, Prolog (GNU Prolog, SWI-Prolog, Ciao Prolog, YAP), Maude, Standard ML, Haskell (GHC, Hugs), ...
- Code implementation: C (gcc), Ada (gcc GNAT), ...
  - Java/C#: FLOSS implementations (gcj/Mono) maturing; gc issue
- Configuration Management: CVS, Subversion (SVN), GNU Arch, git/Cogito, Bazaar, Bazaar-NG, Monotone, Mercurial, Darcs, svk, Aegis, CVSNT, FastCST, OpenCM, Vesta, Superversion, Arx, Codeville...
- Testing: opensourcetesting.org lists 275 tools Apr 2006, inc. Bugzilla (tracking), DejaGnu (framework), gcov (coverage), ...

See “Open Technology Development Roadmap”

5. More OSS-like gated communities

- **Sometimes don’t want to release as OSS, but want share internally in DoD community**
- **Increased attempts to form “gated communities”**
  - **Contracts give government unlimited rights for self**
  - **Microsoft trying to implement gated communities**
  - **Problems: Smaller userbase -> less cost-sharing, less innovation, fewer developers; culture change; Non-commercialization; contractors don't perceive a WIIFM**
    - **Result: Less likely to succeed than OSS**
  - **Sometimes provides some of benefits of OSS**
- **SourceForge-like repositories for government**
  - **Existing ones grow, more will start**
  - **Will use OSS development tools (SCM, mailing lists, etc.)**
  - **May also be useful for special “vetted” OSS versions**

It only makes sense!

- **“In cases where the military pays to develop software for its own use... DoD needs to assert its legally established government rights to view, access and modify code, and leverage it across the department”**
  - **Sue C. Payton, Deputy Undersecretary of Defense (Advanced Systems and Concepts)**

See "OSS in Government Acquisition" & "How to Evaluate OSS/FS Programs" www.dwheeler.com

6. More guidance

- **How to judge OSS vs. other approaches**
  - **Calculating real TCO (nothing free; id inputs, rules of thumb)**
  - **How to evaluate existing OSS (inc. for security)**
- **Before including *any* software, examine license**
- **How to make major mods & new OSS components**
  - **How to enable culture change; how to set up, run, etc.**
  - **Maximize number of potential users**
- **License selection**
  - **Modify existing component & release: Use their license**
  - **New: Use one of most common commercial licenses that are mutually compatible with other most common**
    - **MIT/X11, BSD-new, LGPL, GPL; & explain *why***
- **Legal clarifications (esp. LGPL/GPL, classified)**
- **Management how-to, causing culture change**

7. Net-centricity/Service-Oriented Architecture (SOA)=>OSS on Servers

- Increasingly services accessed through standard interfaces on TCP/IP
- Services can run on anything, use locally whatever's sensible
- Issue: Current SOA standards very complex
  - Need to simplify use, or will be replaced
  - *Idea* of SOA right, but potentially simpler tech like REST, XML-RPC, etc. could bury WSDL etc.
- No barrier to OSS use on servers leads to rapid increase in OSS use on servers
  - Accelerated in addition by a slow shift to more server-centric applications, next...

8. Clients move to Web applications & virtualization

- Current desktop environment increasingly problematic
  - High admin cost (install/upgrade doesn't scale)
  - Hard to secure, inadequate data isolation on failure
  - Applications sometimes require incompatible platforms
  - "Can't install until approved" inflexible, not safe enough
- Increased movement to web apps & virtualization
  - HTML+CSS; Javascript->AJAX; Java; VNC and even X
  - Technologies like XUL enable disconnected use
  - Virtualization->separation; remotable; run untrusted apps
  - We'll still have desktops, but apps migrate to network
- OSS (Linux) desktops may increase, no "Linux year"
- Eases switch to OSS applications (most run on server)
- Exception: jamming env. (tactical)->Limited bandwidth

See "Open Standards and Security" www.dwheeler.com

Open Standards are *Important*; see My Parables about Standards...

- **Magic food (independence from supplier)**
  - **Only need food 1/year, all vitamins & minerals, first 1 $1**
  - **... but you can eat ONLY it from now on (others poison), and there's ONLY ONE manufacturer. Think the prices will go up? What's social cost of crack? Dependence is a security problem!**
  - **Not attacking MS/RH/etc. Need suppliers; not dependence on 1**
  - **Two IT independence strategies: Open standards & OSS (differ!)**
- **Firehose couplings (so defenders can cooperate)**
  - **1904 Baltimore fire: cities' couplings differ, 2,500 buildings lost**
  - **Multiple "standards" NOT good; multiple *implementations***
- **Railroad gauge – Contributed to Confederacy's loss**
  - **Eliminate unnecessary costs/time, freeing up money/time**
  - **Plug&play (cars/engines with tracks) allows innovation & improvement (steam→diesel). *No one organization* does all innovation. See also audio equipment**

9. Speculation: Risk of International Standards Process Meltdown

- Roman Republic (pre-Empire) lasted for centuries
  - Depended on consensus process (vetoing partner)
  - Subverted consensus process==>no more republic
- International standards process risking consensus loss
  - China rejecting if encumbered by non-Chinese patents
  - OSS projects cannot use patent-encumbered standards unless royalty-free, and often shunted away from standards process ($$ for consortia)... yet increasingly #1 or #2 market position
  - Microsoft claims it's great to have multiple incompatible international standards/area, each controlled by a vendor (OOXML vs. OpenDocument, XPS vs. PDF, ...)
  - If others follow, dozens of incompatible standards/area
  - Standards bodies get richer by allowing encumbered standards, many standards, charge $$$/document (didn't write, limits use)
  - If everything is a standard, nothing is
    - Baltimore pre-1904, Confederacy railway: lose a city/country

How to subvert the standards process

- Create a specification that incompatibly duplicates existing international standards' functionality (the more the better)
- Create consortia working group you control
  - Rules forbid any substantive changes from your base document, flood membership with pet companies
- Slip in discriminatory usage limitations
  - "Reasonable and Non-Discriminatory" (RAND) of essential claims in software is Newspeak
- Standardize only a useless/ambiguous subset
- Require implementation of bugs so competitors can't do better (e.g., 1900 wasn't leap year; require it)
- Use misleading name
- Fast track, so issues can't be discovered in time
- Flood government voting committees

Possible standards process outcomes

- No change – resisted, or only happens a very few times
- Or Floodgates – every vendor owns its ISO standard
  - Soon international standards meaningless (all have one)
  - People will fall back to products-as-standards (ugh)
  - OSS projects could become vendor-neutral lifeboat
    - Only two supplier-independence tactics: OSS & standards
    - Since anyone can use OSS result, and/or fork a new project, less economic incentive to subvert OSS
- OSS already flexing its muscles
  - Apache/AOL 1996 disagree on HTTP spec: Apache wins
  - GNOME & KDE wiped out CDE/Motif; replacing CORBA with OSS community standard, D-Bus (FreeDesktop.org)
- World needs/users want consensus-based standards
  - Trust, once lost, hard to regain. Collapse not necessary; risk

Conclusions

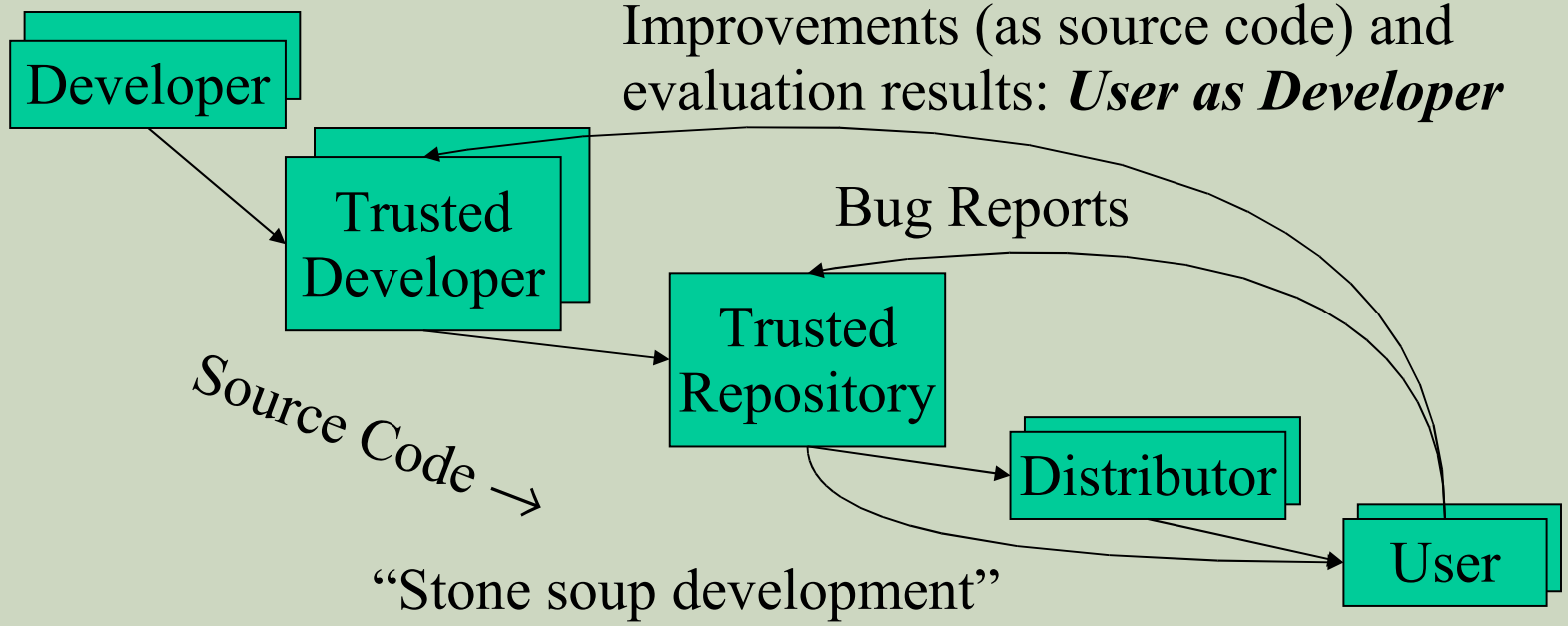
1. Existing OSS Trends Accelerate
2. Increased OSS support business
3. DoD will slowly start more OSS projects
4. General-purpose High Assurance (HA) Components Developed as OSS
5. More OSS-like gated communities
6. More guidance
7. Net-centricity/Service-Oriented Architecture (SOA) => OSS on Servers
8. Clients move to Web applications & virtualization
9. Speculation: Risk of International Standards Process Meltdown

Increasing use in DoD of OSS & OSS-like approaches

Won't eliminate proprietary software; balance will shift

Backup slides

OSS Development Model



- OSS users typically use software without paying licensing fees
- OSS users typically pay for training & support (competed)
- OSS users are responsible for developing new improvements & any evaluations that they need; often cooperate/pay others to do so

Why would governments use or create OSS (value for government)?

- **Can evaluate in detail, lowering risk**
  - **Can see if meets needs (security, etc.)**
  - **Mass peer review typically greatly increases quality/security**
  - **Aids longevity of records, government transparency**
- **Can copy repeatedly at no additional charge (lower TCO)**
  - **Support may have per-use charges (compete-able)**
- **Can share development costs with other users**
- **Can modify for special needs & to counter attack**
  - **Even if you're the only one who needs the modification**
- ***Control own destiny*: Freedom from vendor lock-in, vendor abandonment, conflicting vendor goals, etc.**

In many cases, OSS approaches have the *potential* to increase functionality, quality, and flexibility, while lowering cost and development time

Acronyms

- COTS: Commercial Off-the-Shelf (either proprietary or FLOSS)
- DoD: Department of Defense
- HP: Hewlitt-Packard Corporation
- JTA: Joint Technical Architecture (list of standards for the DoD); being renamed to DISR
- OSDL: Open Source Development Labs
- FLOSS: Open Source Software
- RFP: Request for Proposal
- RH: Red Hat, Inc.
- U.S.: United States

Interesting Documents/Sites

- "Why OSS/FS? Look at the Numbers!" http://www.dwheeler.com/oss_fs_why.html
- "Use of Free and Open Source Software in the US Dept. of Defense" (MITRE, sponsored by DISA)
- President's Information Technology Advisory Committee (PITAC) -- Panel on Open Source Software for High End Computing, October 2000
- "Open Source Software (OSS) in the DoD," DoD memo signed by John P. Stenbit (DoD CIO), May 28, 2003
- Center of Open Source and Government (EgovOS) http://www.egovos.org/
- OpenSector.org http://opensector.org
- Open Source and Industry Alliance http://www.osaia.org
- Open Source Initiative http://www.opensource.org
- Free Software Foundation http://www.fsf.org
- OSS/FS References http://www.dwheeler.com/oss_fs_refs.html