Stephen L Arnold

Systems Engineering/Architecture and FOSS Developer

Education

1995-Present:	Graduate/short courses in Toxic Risk Assessment, C, Ada, Zope, Doppler and Polarimetric Weather Radar, 595th TEG Test and Evaluation Course, AMS Faculty Workshop, AHC Professional Development courses.
1990:	MS Degree in Geophysics, San Diego State University.
1986:	BS Degree in Geophysics, San Diego State University.

Technical skills

CS/SE:	Expert knowledge of programming languages and development tools, system performance, design, testing, and administration, as well as the software lifecycle, CI, software processes, requirements engineering, system architecture, and CyberSecurity
OS:	Unix/Linux/Embedded (Gentoo, OE, RHEL, Debian/Ubuntu), Android
Proj Mgmt:	SE DIDs, Planner, OpenAdams, doorstop, Make/Autotools, trac/git, doxygen, docutils
DataBase:	sqlite, redis/nosql, Postgres/spatial
Software:	Libre/Open office tools, Graphviz, Dia, Inkscape, Maxima, Octave
Architectures:	x86/x86_64, ARM/AVR, MIPS, Sparc, PowerPC
Embedded Systems:	Gentoo, OpenEmbedded, debian/Ubuntu, design/build/deployment of applications for ARM and other embedded systems, Android
Sciences:	Geophysical fluid dynamics, sensors and data acquisition, data analysis & modeling, geology/geography, geophysical/multi-spectral remote sensing and geospatial analysis, environmental/health risk assessment, applied math modeling

Languages

English:	Native
Spanish:	Conversational(-ish)
Programming:	Python, Bash/POSIX Shell, C, Ada, Java, C++, Perl, js, AWK, FORTRAN
Markup:	reStructuredText, HTML, DTML, XML, Markdown
Engineering:	IV&V, OOD/P, UML, DoDAF, RMF/DIACAP, 2167/498/12227, RCC/AF/NASA Range Safety Stds toolchains/SDKs, CI/Agile, jenkins/apache/trac/git workflows, open document production
Applied Math:	Numerical analysis, linear systems, differential equations, functional analysis, non-linear dynamics, math modeling, statistical methods, risk models

Relevant Work History

2014 - Present: Principal Scientist, Systems Architect, Business & Community Development - Vanguard Computer Technology Labs, Inc - Goleta, CA. VCT Product/Project management IV&V, Range Meteorology and Hazard/Risk Modeling Subject Matter Expert (SME), education & training instructor, conferences & expos, open source outreach & education. Systems Architecture and CyberSecurity, specializing in Gentoo Hardened, OpenEmbedded, RHEL, and Debian/Ubuntu, Linux development, build, and deployment testing. Linux kernel/u-boot hacking and software testing on various ARM devices (Gentoo Linux, OE). Business/community development (event support, outreach, presentations, leads & proposals).

2011 - 2014: Senior Software Engineer - SynergyHD3 - Arthrex California Technology, Inc. - Goleta, CA. DevOps, software/systems engineering, CM/QA, test & integration. Agile infrastructure & CI process implementation, Jenkins build/test/deployment automation. Linux OS/Distribution engineering, test automation, process validation.

2009 - 2011: Staff Scientist, IV&V / Systems Engineering and Architecture Support - ENSCO, Inc. - VAFB, CA. Subject Matter Expert – Range Meteorological Systems, Flight Safety, Instrumentation. Launch Range Enterprise Architecture analysis (supersystem, subsystem, product). Safety-critical systems IV&V (Flight Termination, Range Timing). Range Architecture Model VV&A, model evaluation, code analysis, and test engineering. Software process, internal SCM and IV&V tool engineering.

2006 - 2009: Staff Scientist, System Performance Evaluation and Validation - ENSCO, Inc. - VAFB, CA. Performance analysis and verification & validation of Weather, Flight Operations, TEMS/Timing and Infrastructure products for new Mission Flight Control Center. Responsible for Weather, Infrastructure, and Data Handling subsystems; technical IA and System Security/Accreditation support.

2002 - 2006: Senior Engineer, MARSS Project, Range Instrumentation and Infrastructure - ENSCO, Inc. - Santa Maria, VAFB, CCAFS. RSA-MARSS and INF system design and requirements engineering, user training. Model evaluation and localization, integration with MM5/AWIPS, range instrumentation integration and test support.

1997 - 2002: Senior Scientist - Independent Verification & Validation - ENSCO, Inc. - VAFB, CA. IV&V of Launch & Test Range Meteorological and Flight Safety Systems, Range meteorologist and engineering test support to AFOTEC and 17th Test Squadron.

1991 - 1997: Senior Systems Engineer, IV&V Task Lead on Meteorological Range Safety Systems - Geodynamics Corp - VAFB, CA. Requirements/code analysis, windfield, dispersion, & hazard/risk model evaluation, dense gas hazard analysis, developed new statistical estimation algorithm for processing time-averaged wind data without access to the raw samples. Also performed real-time range safety algorithm & performance analysis, instrumentation and data quality analysis.

1990 - 1991: Applied earth science & software consultant, San Diego, CA. Air toxics risk assessment modeling studies, control-tradeoff & sensitivity studies, environmental assessments, cost-benefit analyses, bug-fixes and model enhancements (IWG Corp), geophysical data acquisition (JR & Assoc), database design/implementation (SDSU Foundation).

1984 - 1990: Research Assistant, Coastal and Boundary Layer Meteorology - SDSU and Scripps Center For Coastal Studies, San Diego, CA. Field observations, data collection, analysis, post-processing for 3 major field programs, scientific programming, custom instrumentation development (tethered sounding gear, kite and balloon platforms).

FOSS Experience

2014 - Present: Founding member Central Coast Open Source Solutions Exchange, an open source technology-focused meetup. Technology and Open Source adviser, Santa Maria Startup Weekend. Open source presentations, technology training, demos.

2012 - Present: Co-maintainer of meta-printing layer, - OpenEmbedded and Yocto contributor (oe-core, meta-openembedded, meta-python).

2003 - Present: Senior Developer - Gentoo Linux. Previuously maintainer of several development tools, GIS/scientific libraries, mentor of new developers; currently primary maintainer of Gentoo ARM overlay and personal testing overlay.

2000 - Present: Upstream developer and/or maintainer of several tools and utility libraries for source code metrics, graphics, science, and education. See the maintenance release page and the individual github project sites for more information.

Education Experience

2014 - Present: Volunteer Instructor/Mentor - Google Summer of Code, Computer Science Education Week, and Open Document Foundation's Document Freedom Day.

- GSoC mentor for BeagleBoard.Org: BeagleSat & anemometer projects.
- Computer Science Education Week: Python Intro to CS course.
- ODF contributor, Document Freedom Day: Open Document Workshop.

1999 - 2009: Associate Faculty - Allan Hancock College (senior geography and meteorology instructor). Taught Physical and Human Geography courses and occasional technology courses, updated official geography course outlines, created new introductory meteorology course.

Selected Projects

- GNU Winds On Critical Streamline Surfaces (GWOCSS) diagnostic windfield model
- af_alg - an openssl engine for the kernel user-config crypto interface
- CCCC & Cyclo tools for source code complexity and structural metrics
- URT - Updated version of the Utah Raster Toolkit and RLE graphics library
- Open Source Weather Station - Raspberry Pi and weather data ingest and display software, arduino lightning detector (AMS WeatherFest demo)
- Various embedded Linux tools (build manifests, kernel/u-boot patches and rootfs tools, etc)

  - vct-socfpga-bsp-platform - Manifest tool for Poky/OpenEmbedded on Altera socfpga boards
  - vct-beagleboard-bsp-platform - Manifest tool for BeagleBoard.Org and/or TI builds of Poky and OpenEmbedded
  - vct-edgerouter-bsp-platform - Manifest tool for Poky/OpenEmbedded on MIPS Edgerouter Lite
  - DE1_SOC_Linux_FB - FPGA demo project, fixed and documented for DE1- SoC board
  - mips3-octeon-devel - Build script for mips3 n64/n32 linux-stable patches
  - gnu-arm-toolchain - Modified build scripts and readme for building on ARMv7a host
  - meta-small-arm-extra - Custom ARM device layer for RPi, BBB, imx2x, Neo and similar
  - meta-alt-desktop-extras - Usability-focused Poky/OE desktop layer for constrained devices
  - multi_arch-mainline-linux - Kernel build and deb packaging scripts modified for native x86

Selected Publications

Towards an Open Instrumentaion Platform: Getting the Most From MAVLink, ArduPilot, and BeagleBone (2016) S. L. Arnold, presented at the Special Symposium on Meteorological Observations and Instrumentation, Special Session: Unmanned Aerial Vehicles for Atmospheric Research and Field Programs (97th Annual AMS Meeting, 2017).

Open Data Standards and Open Source Modeling Tools: The GPL'd Release of Winds On Critical Streamline Surfaces (GWOCSS) (2015) S. L. Arnold, presented at the 31st Conference on Environmental Information Processing Technologies, Open Data Standards and Sharing track (95th Annual AMS Meeting, 2015).

Incorporating the AMS Online Weather Studies Resources In the Design of a New Meteorology Course (2007) S. L. Arnold, presented at the Educational Initiatives Poster Session, 17th Symposium on Education (88th Annual AMS Meeting, 2008).

Open Source Technologies in Science Education: What's Your Geek IQ? (2004) S. L. Arnold, presented at the Joint Session on Cyberinfrastructure to Support Atmospheric and Oceanic Education: Examples and Strategies, 14th Symp. on Education (85th Annual AMS Meeting, 2005).

The Meteorological And Range Safety Support (MARSS) System: a GIS-based Tool for Launch Area Hazard Prediction and Visualization (2004) S. L. Arnold, A. Dianic, and E. Magnuson, presented at the 21st Int Conf on Interactive Information Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology (85th Annual AMS Meeting, 2005).

An Operational System for Real-time Lightning Display and Resource Protection (2004) E. Magnuson, S. L. Arnold, and A. Dianic, presented at the Conference on Meteorological Applications of Lightning Data (85th Annual AMS Meeting, 2005).

A Minimum Variance Approach to Estimating Wind Direction Statistics (2001) S. L. Arnold, presented at the 18th Int Conf on Interactive Information Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology (82nd Annual AMS Meeting, 2002).

Physical & Thermodynamic Properties of Hypergolic Propellants: A Review and Update. (1999) S. L. Arnold, presented at the 1999 JANNAF Inter-agency Propulsion Committee, PD&CS, and S&EPS Joint Meeting, San Diego, CA.

Professional

Affiliations:	American Meteorological Society, American Geophysical Union, Assoc. for Computing Machinery, Computer Science Teachers Assoc, American Federation of Teachers
Interests:	Earth Science Education, Computer Literacy/Fluency, Privacy and Security, Linux and Open Source, Environmental Risks and Hazards, Space Exploration and Technology

Personal

Home sites:	http://www.gentoogeek.org and http://dev.gentoo.org/~nerdboy/
Repositories:	https://github.com/sarnold and https://github.com/VCTLabs
Publications:	http://www.researchgate.net/profile/Stephen_Arnold4
Interests:	Guitar/Bass/Percussion, Science Fiction, Open Source

Appendix A

Example of open source use in engineering; graphviz diagrams and IV&V.

Overall IV&V Engineering Process for Tracking Anomaly Reports

- Revise: Problem described in AR was partially fixed but needs additional work; a new revision of an AR has been written.
- Validate: Developer/customer validates ARs in the Pending state.
- Prioritize: Developer/customer prioritizes ARs and assign to project(s).
- Verify: IV&V verifies ARs declared fixed by the developer. Partial fixes generate a revision of the original AR (the latter is closed based on the partial fix).
- Report: IV&V reports new ARs, AR State changes, and status of open ARs.

AR States and Status

- Draft State: Anomaly Reports begin in a draft status for IV&V peer review; refined drafts may be circulated outside IV&V if warranted.

  - Draft Status: IV&V peer review of potential anomalies results in publication of draft AR.
- Open State: Open ARs begin with a status of "Pending" when an approved draft AR is published. Engineering review leads to the next status change, typically "Valid". Valid ARs are prioritized and assigned to an appropriate project; when engineering considers the problem fixed, the AR status is changed to "Fixed". Fixed ARs are verified by IV&V, and their status changed to "Verified" if the problem was fixed. In the case of a "partial" fix, a the original AR is declared "Verified" and closed based on the fix, and a new revision of the AR is written to describe any remaining issues.

  - Pending Status: An original or revised AR is generated, entered in the database, and delivered to project distribution list.
  - Valid Status: Developer (or customer) validates that AR identifies an error or problem condition that must be fixed.
  - Fixed Status: Problem described in AR is considered fixed by developer but has not been verified by IV&V.
  - Verified Status: Problem has been fixed by developer and IV&V has verified the fix.
- Closed state: Typically closed ARs have the status "Verified" (verified by IV&V), however, there are several other potential status flags for closed ARs, depending on the circumstances (see below).

  - Invalid Status: AR is considered technically inaccurate and does not describe an error.
  - Rejected Status: AR is technically accurate but the problem will not be fixed due to non-technical reasons.
  - Accepted Risk Status: Cost/benefit ratio does not justify fixing the problem.
  - Unverifiable Status: Original problem cannot be recreated in order to verify fix, and there is no other recourse.
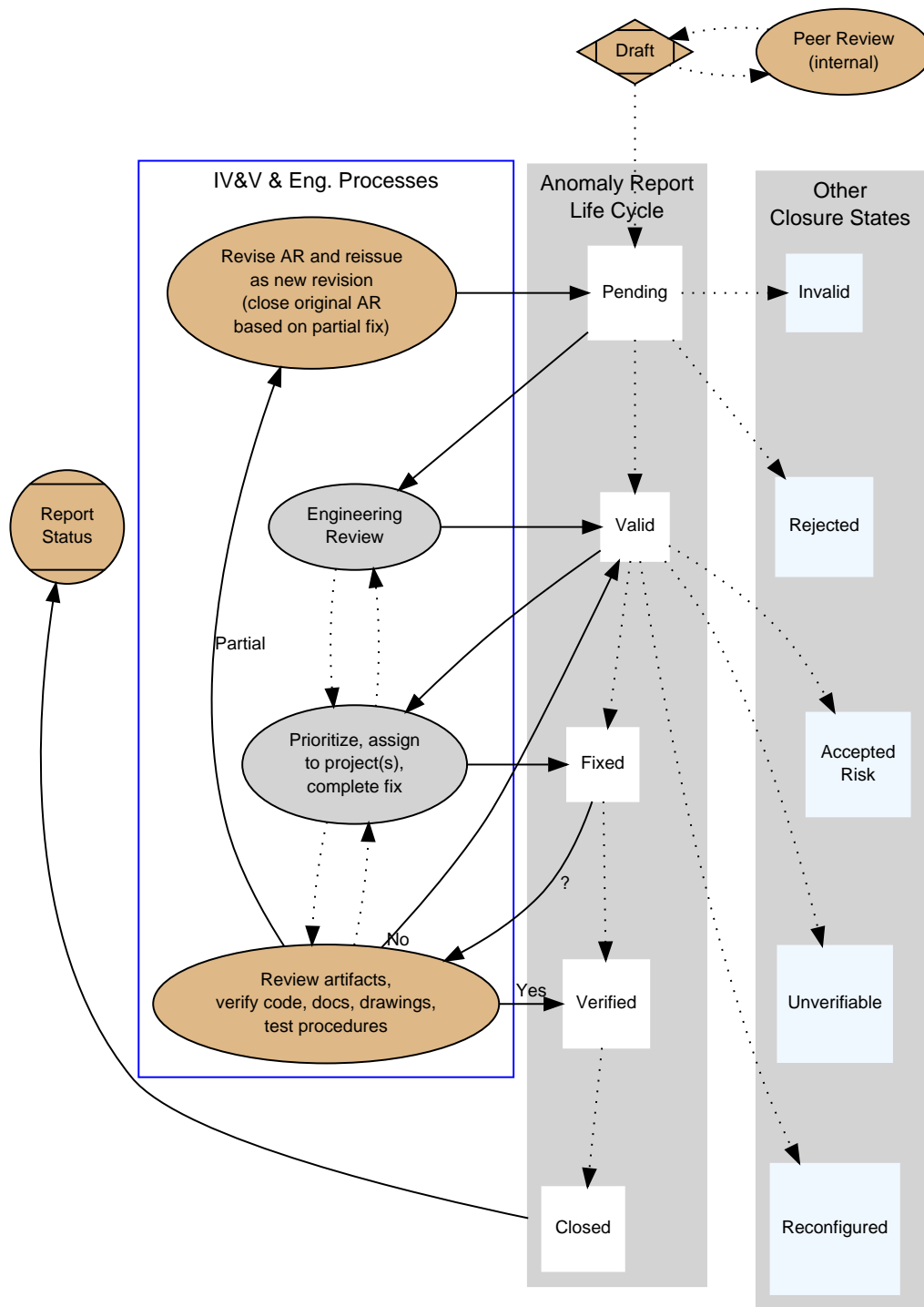 - Reconfigured Status: System has changed such that the original problem no longer applies.

Fig A. Anomaly Report (AR) State Diagram

The Dot source code for the AR diagram is included below.

```
1 digraph G {
2     size="6,8";
3 //    ratio=fill;
4     compound=true;
5     labelfontname=Arial;
6     fontname=Arial;
7     fontsize=10;
8     subgraph cluster0 {
9         node [style=filled,color=white,shape=square,fontname=Arial,fontsize=10];
10        style=filled;
11        color=lightgrey;
12        fontname=Arial;
13        fontsize=12;
14        label = "Anomaly Report\nLife Cycle";
15
16        Pending -> Valid -> Fixed -> Verified -> Closed [style=dotted];
17    }
18    subgraph cluster1 {
19        node [style=filled,fontname=Arial,fontsize=10];
20        fontname=Arial;
21        fontsize=12;
22        label = "IV&V & Eng. Processes";
23        color=blue
24
25        revise [label="Revise AR and reissue\nas new revision\n(close original AR\nbased on partial fix)",fillcolor=burlywood]
26        validate [label="Engineering\nReview"];
27        prioritze [label="Prioritize, assign\nto project(s),\ncomplete fix"];
28        verify [label="Review artifacts,\nverify code, docs, drawings,\ntest procedures",fillcolor=burlywood];
29
30        revise -> validate -> prioritze -> verify [style=invis];
31    }
32    subgraph cluster2 {
33        node [style=filled,color=powderblue,shape=square,fontname=Arial,fontsize=10];
34        style=filled;
35        color=lightgrey;
36        fontname=Arial;
37        fontsize=12;
38        label = "Additional\nClosure Status";
39
40        Invalid -> Rejected -> Accepted -> Unverifiable -> Reconfigured [style=invis];
41    }
42    ranksep=.75;
43
44    { rank = same; "Draft"; "peer"; };
45    { rank = same; "revise"; "Pending"; };
46    { rank = same; "Pending"; "Invalid"; };
47 //    { rank = same; "Closed"; "report"; };
48 //    { rank = same; ""; ""; };
49
50    Draft [style=filled,shape=Mdiamond,fillcolor=burlywood,fontname=Arial,fontsize=10];
51    peer [style=filled,label="Peer Review\n(internal)",fillcolor=burlywood,fontname=Arial,fontsize=10];
52    report [style=filled,shape=Mcircle,label="Report\nStatus",fillcolor=burlywood,fontname=Arial,fontsize=10];
53    Accepted [label="Accepted\nRisk"]
54
55    Draft -> peer [style=dotted];
56    peer -> Draft [style=dotted];
57    Draft -> Pending [style=dotted];
58    Pending -> validate;
59    Pending -> Invalid [style=dotted];
60    Pending -> Rejected [style=dotted];
61    validate -> Valid;
62    prioritze -> validate [style=dotted];
63    validate -> prioritze [style=dotted];
64    Valid -> prioritze;
65    Valid -> Accepted [style=dotted];
66    Valid -> Unverifiable [style=dotted];
67    Valid -> Reconfigured [style=dotted];
68    prioritze -> verify [style=dotted];
69    verify -> prioritze [style=dotted];
70    prioritze -> Fixed;
71    Fixed -> verify [label="?",fontname=Arial,fontsize=10];
72    verify -> Verified [label="Yes",fontname=Arial,fontsize=10];
73    verify -> revise [label="Partial",fontname=Arial,fontsize=10];
74    verify -> Valid [taillabel="No",fontname=Arial,fontsize=10];
75    verify -> report [style=invis];
76    revise -> Pending;
77    Closed -> report [weight=10,ltail=cluster0];
78    Reconfigured -> report [weight=10,ltail=cluster2];
79 }
80
```

Appendix B

Experiments in git commit graphs using Graphviz (approx. 10 years of data).

The full graphs are too large and/or not "compatible" enough to embed in a simple document; the full size .svg graphs are hosted on dev.gentoo.org.
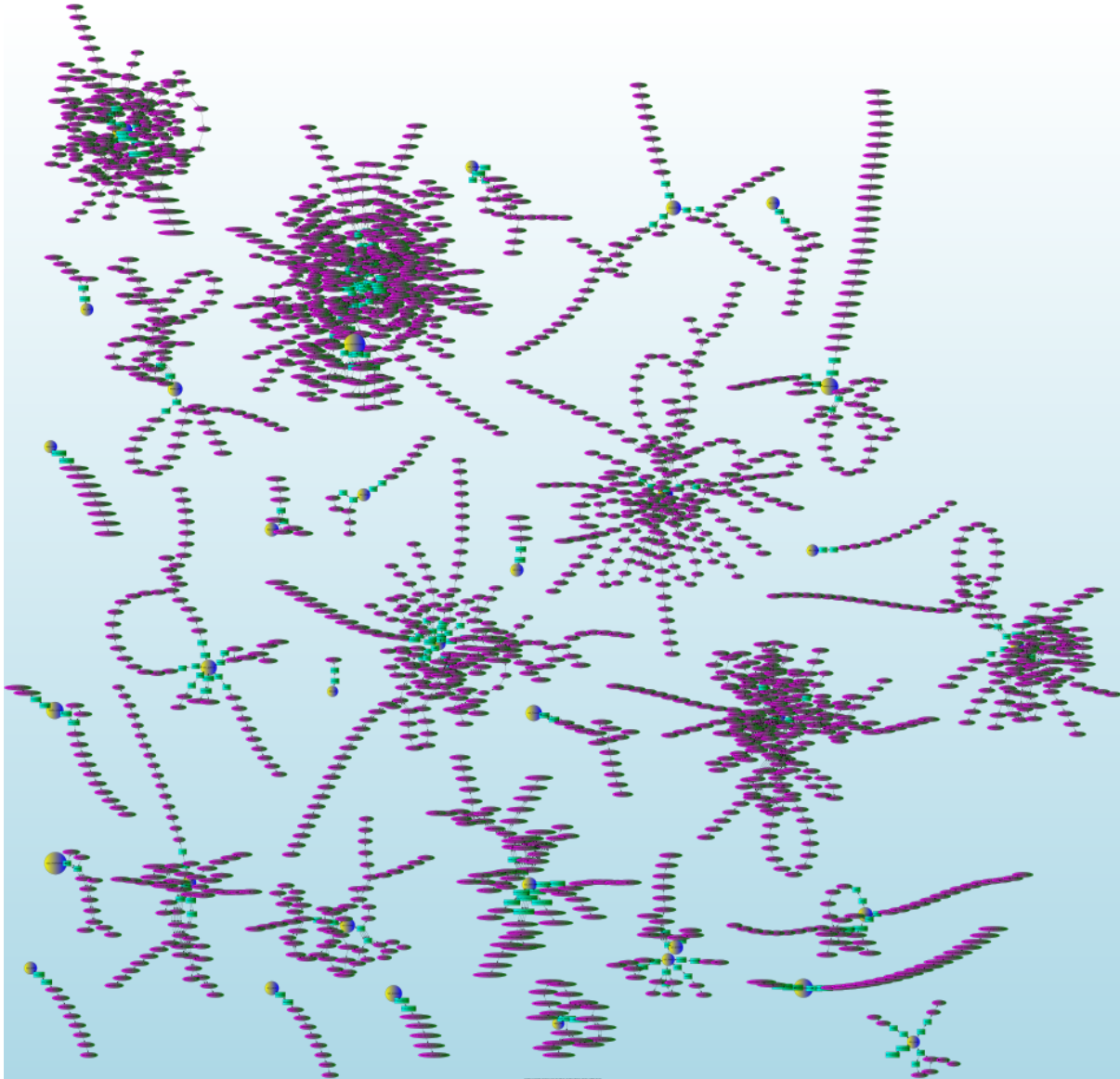

Fig B. Category / Package Cluster Graph, Gentoo Linux commits by nerdboy, 2003 - 2013