



**“The GNU C Library” or
“That *thing* between you and
your goal...”**

2013-04-16

Carlos O'Donnell
Red Hat

That *thing* between you and your goal...

- Linked against by virtually all applications...
 - Includes: libc, libm, libpthread
- Consequently the subject of a lot of complaints...
 - Too slow (Performance)
 - Incorrect behaviour (Conformance)
 - Missing features (New APIs, New machine port)

We like it this way.



That *thing* between you and your goal...

- Complaints mean that you're project is in use!
 - Right? Maybe? :-)
- What next?
 - Stay relevant
 - Engage developers
 - Get developer help to address complaints
 - Give talks like these



Overview

- **Welcome!**
- Whirlwind tour of new stuff!
 - Community status
 - Release status
 - New generic features
 - New machine support
- Where do we need help?
 - Math library variants
 - Performance and benchmarking
- Questions?



Welcome!

- Kernel or application developer alike, glibc has an impact on you (unless you're on android)!
- Excellent audience feedback last year
- We listen!

Remember your questions and save them for the end of the talk!



Community status

- We've made it through our documented time boxed release process two more times!
- Two developers (Andreas Jaegar and myself) stepped up to become dedicated x86-64 and x86 reviewers (previously only community-based review)
- New developers pitching in to help with guidance from old hands.
- Renewed attention to locales and fixing locale issue



Community status

- Still in need of an effective bugzilla triage process
- More patches than core community can review
- Less than optimal testing infrastructure

Why am I excited by this?



Digression? Logo?

- Logo contest on LWN?
- Initial ideas:
 - Keystone (holding up a bridge)?
 - Pillar (old Greek style)?
 - Bird (Glossy Ibis, Obese chimera)
 - Tree?
 - Diamond (Brilliant cut, Round cut)?
 - GNU in a library?
 - Bridge Troll?
- Commission a logo after the contest is done



Release status:

- 6th annual LFCS in April 2012
 - **“The Future of the GNU C Library”**
- 2.16 released June 2012
- 2.17 released December 2012
- 2.18 in active development
- Time boxed releases
- Two months left for 2.18 development
- Many features on the list for 2.18



Release status: 2.18 development

- Hardware lock elision support
 - Add support to glibc for hardware lock elision on Intel's Haswell cores (June 2013)
 - Currently under review (next week I promise Andi!)
- Library dependency handling cleanup
 - Unit test loader internals and dependency ordering
 - Auto-generate millions of possible orderings and test
- Math routine cleanups



Release status: 2.18 development

- IPv4/IPv6 dualstack issues.
 - Resolver issues
 - 24 getaddrinfo() bugs in sourceware bugzilla.
 - 14 getaddrinfo() bugs in Red Hat bugzilla.
- Parallelism and concurrency fixes
 - Unification of pthread_once implementations
 - Large TLS segments
 - Review race conditions



Release status: 2.18 development

- Header cleanup
 - Ended up mirroring the UAPI changes in the Linux kernel.
 - Looking to ensure that order of Linux header inclusion doesn't matter.
- Bugs
 - 54 bugs fixed
 - 2 CVE's



Release status: 2.18 development

2 months left!



New features?

- glibc-ports repo merged into main glibc repo
 - Based on last year's feedback
 - Ports repository held “2nd class architectures”
 - e.g. ARM, MIPS, etc.
 - History preserved
 - Working to ensure generic changes are done across all ports
 - If we had the resources we'd convert all targets to ports and then we'd have a clean core/ports split



New features?

- Minimum Linux kernel supported is 2.6.16 (2.17).
- Improved X support:
 - Cross-compilation
 - Cross-testing (wrapper)
 - Bootstrap builds (no previous glibc built)



New features?

- Performance
 - New micro-benchmark suite! (More about this later...)
 - Optimized functions for s390/s390x, POWER7, MIPS,
- Language
 - C++11 `thread_local` object destructors on thread and program exit.
- Parallelism and Concurrency
 - Support for PI mutexes in generic `pthread_cond*` code.



New features?

- Math library
 - IEEE 754:2008 conformance fixes
 - Clearer definition of errors i.e. `ulp(x)`.
- New locales
 - `mag_IN`, `ayc_PE`, `doi_IN`, `ia_FR`, `mni_IN`, `nhn_MX`, `niu_NU`, `niu_NZ`, `sat_IN`, `szl_PL`.



New features?

- ARM hard-float ABI variant uses `/lib/ld-linux-armhf.so.3`
 - As of 2.16 with fixes for mixed use environment in 2.18
- New class of installed header for low-level platform-specific functionality
 - PowerPC access to timebase register
- Timezone data no longer installed
 - See tzdata package



New machine support?

- Xilinx Microblaze as of 2.18
 - In review...
- ARM AArch64 support as of 2.17
 - 64-bit ARMv8
- Tiler support as of 2.16
 - TILE-Gx and TILEPro
- x32 ABI support as of 2.16
 - Hybrid 32-bit/64-bit ABI



New machine support?

- New machine?
 - Please work with the community.
 - Ensure your port arrives on time.
 - Ensure your port arrives in the release you need.
 - Contact any of the FSF glibc project stewards to talk about your needs.

Work with us!



Short term goals?

- More developers, reviewers and testers? YES.
- More distribution involvement? YES.
- Getting the word out? YES.
- Making releases? YES.
- Bug triage? NO.
- Cooperation with TIRPC? NO.



Medium term goals?

- Merging eglibc / glibc? SOME
- Testing? SOME
 - Criteria for performance management?
- Documentation? SOME
- Upgraded testing framework? NO



Long term goals?

- Autogenerated libm? YES
- Trace? SOME
 - Dynamic loader
 - Memory allocation routines
 - Threading
 - Math library slow paths
- Power awareness? NO
- Exceptionless syscalls? NO



Where is collaboration required?

- Math library variants
- Performance benchmarking
- Runtime tunables
- Networking
- Localization
- Documentation



Math Library:

- What's wrong with the math library?
 - Imprecise
 - Don't care
 - Too slow
- Three variants
 - High precision
 - Default
 - Constant runtime



Math Library:

- IP6 lab at UPMC in Paris France
 - Metalibm
 - Based on crlibm
- Uses sollya
 - Proved by gappa
 - Generates C code
 - Compiles into library variants
- Different functions selected by compile flag?
 - -mfast-math (exists), -mprecise-math (new)



Math Library: “Help Wanted”

- Expand microbenchmark coverage
- Validate new libm function implementations
 - Iterating the solution space works sometimes
 - Double? Long double? Multi-variate? NO
- Build out infrastructure for selecting alternate implementations



Performance and benchmarking:

- Claim:
 - high performance
 - No data to backup claim
- What next?
 - Measure performance
 - Allow users to gather data and identify problems
 - Simple
- Acceptance criteria for perf patches



Performance and benchmarking:

- Microbenchmarks
 - Relative differences on one target
 - Specific use cases
- Whole-system benchmarking
 - Performance impact for your workload?
 - Power consumption impact?



Performance and benchmarking: “Help Wanted”

- Add more microbenchmark tests
- Add more whole system benchmark tests
- Add systemtap probes to more places in glibc
- Methods to gather and analyze user results



Runtime Tunables: “Help Wanted”

- Defaults don't match all workloads
- Choosing defaults to match average workloads
- Provide users with the ability to capture their own data
- Live tuning?
- API?
- What's in the ABI?
- Lots of help required...



Networking: “Help Wanted”

- IPv4/IPv6 dualstack issues
 - Implementation of getaddrinfo() has issues
 - Doesn't match other OS implementations
 - Causes porting problems
 - Network experts wanted
 - Interpretation of RFCs
 - Experience with implementations across OSs



Locales: “Help Wanted”

- Language experts
- Stewardship of locales
- Strengthen ties to stakeholders in the community
 - GNOME, KDE, etc..
- Work with CLDR



Documentation: “Help Wanted”

- Started documenting compliance
 - Thread safety, Async-signal safety, POSIX compliance
- Whole new chapter on threads!
 - Authors welcome!



Questions?

Thank you for attending this talk!

Questions?

- Ask any questions on libc-help@sourceware.org!
- Post RFCs or WIPs on libc-alpha@sourceware.org
- Submit a bug in <http://sourceware.org/bugzilla>

