

NPTL Support for uClibc

2006 Linux Symposium
July 20th, 2006

Steven J. Hill
Reality Diluted, Inc.

Agenda

- What? NPTL support for uClibc.
- Why? uClibc only supports linuxthreads.
- Who?
 - Broadcom Corporation
 - CodePoet Consulting, L.L.C.
 - Reality Diluted, Inc.
- How?

NPTL Background

- Fully POSIX compliant threading library
- POSIX signals and high-resolution timers
- Replace the linuxthreads hack
- Numerous architectures supported
 - x86, x86_64, Itanic 64
 - PPC, PPC64
 - Alpha, ARM, PA-RISC, SuperH
 - MIPS

NPTL for Embedded Systems?

- Faster and lower power processors
- Denser memory devices
- Demanding applications and middleware
- Cell phones, settop boxes, electric razors
- Deeper embedded applications should use linuxthreads

uClibc Project Goals

- Designed for embedded systems
- Smaller size
- Highly configurable
- Smaller size
- POSIX Compliance
- Smaller size
- Simpler toolchains

uClibc vs. glibc Round 1...FIGHT!

Feature	uClibc	glibc
Full POSIX compliance	N	Y
Binary compatibility	N	Y
NSS support	N	Y
NIS support	N	Y
double math support	N	Y
Locale support	Y	Y
LGPL	Y	Y
NPTL support	Y	Y
Small disk footprint	Y	N
Small memory footprint	Y	N
MMU-less support	Y	N
Highly configurable	Y	N
Simple build system	Y	N
Configuration system	Y	N
Ease of maintenance	Y	N
Linuxthreads support	Y	N
Multiple architectures	Y	N
Fast execution speed	Y	N

Linuxthreads vs. NPTL FATALITY!

Feature	Linuxthreads	NPTL
Disk Storage Size	Smallest	Largest
Run-time Memory Usage	Smallest	Largest
Number of Threads	Hard Coded Limit	Dynamic Limit
Thread Efficiency	Slowest	Fastest
Per-Thread Signals	No	Yes
Inter-Thread Synchronization	No	Yes
POSIX.1 Compliance	No	Yes

Source Code and Tool Versions

Component	Version
binutils	2.16.1
gcc	4.1.0
glibc	20050823
uClibc-nptl Branch	20060318
Linux Kernel Headers	2.6.15
LTP	20050804
Open POSIX Test Suite	20050804
buildroot	20060328
crosstool	0.38
Linux/MIPS Kernel	2.6.15

NPTL Implementation for uClibc

- Initial code import
- Futexes
- Thread Local Storage (TLS) support
- Thread cancellation points
- Thread library
- POSIX Timers

Initial Code Import and Build

- Makefiles
- Headers
- Include paths
- Missing macros and functions
- Stubs and auto-generated files
- Symbol aliases
- Unused files

Futexes

- Fast user-level mutexes needed for high-performance threading
- Uses Linux kernel locking primitives
- Read Rusty's paper
- Futexes used in uClibc I/O subsystem

Thread Local Storage (TLS)

- Dynamic Loader a lot of work
 - 2200 lines of code changes for loader and the libdl.so library.
 - No TLS variables in loader itself for uClibc.
- TLS variables prefixed with `__thread`.
- Statically linked threaded applications are not supported.
- Total of 18 TLS variables in glibc.

Thread Local Storage cont.'d

- Only 4 TLS variables were implemented.
 - errno
 - h_errno
 - _res
 - RPC_VARS
- Remaining variables for locale support.
- Threaded locales unsupported in uClibc.

Cancellation Points

- Thread cancellation?
- Cancellation Points.
- Asynchronous Cancellation.
- Most cancellation points were ported.
- Each library has own enable/disable.
- Aliases haunt my dreams.

Thread Libraries

- uClibc library almost verbatim to glibc.
- All code with SHLIB_COMPAT deleted.
- All files with old_pthread_xxx deleted.
- No Asynchronous I/O files ported.
- libthread_db ported near perfectly.

POSIX Timers

- Not original Statement of Work.
- Ported for completeness.
- Tests with POSIX timers and NPTL pass.

Testing the Implementation

- uClibc Testsuite
- glibc Testsuite
- Linux Test Project (LTP)
- Open POSIX Testsuite

uClibc and glibc Testsuites

- Entire uClibc testsuite passed.
- All TLS tests pass.
- All NPTL tests pass.

LTP Test Results

glibc	
fcntl	14,23,24,25,26
gettimeofday	02
mem	02
syslog	01,02,03,04,05,06,07,09,10
msgctl	02
msgsnd	01
rename	03,04
rwtest	03,04
setregid	02
syscall	01
utime	01,02,03
writew	03,04

uClibc	
fcntl	24,25,26
gettimeofday	01,02
mem	02
syslog	12
gf	01,10,11,14,15,16,17,18,19
inode	02
socketcall	02
sysfs	04,05,06

Open POSIX Testsuite

	glibc	uClibc
TOTAL	1823	1665
PASSED	1442	1428
FAILED	113	78
UNRESOLVED	146	63
UNSUPPORTED	22	29
UNTESTED	92	60
INTERRUPTED	0	0
HUNG	1	2
SEGV	6	5
OTHERS	1	0

Size Matters (Maybe?)

	glibc	uClibc NPTL	uClibc Linuxthreads
libc.so	1673805	717128	719836
ld.so	148652	35836	26864
libpthread.so	120825	97189	98940
libthread_db.so	37277	23252	14608

Future Work

- Add other processor architectures.
- Get static libraries working.
- Implement POSIX message queues.
- Implement Asynchronous I/O.
- Merge NPTL code into trunk.
- Sync with latest glibc NPTL code.

Further Information

- Symposium Paper
- <http://www.realitydiluted.com/nptl-uclibc>
- <http://people.redhat.com/drepper/>
- <http://www.uclibc.org/>