uClibc

Erik Andersen
Codepoet Consulting
26 January 2005
In the Beginning...

- Began doing embedded Linux in the middle of 1999
- The state of the art...
  - Copy binaries from RedHat, Debian, etc
  - Build from scratch – just like a workstation
  - Linux Router Project
How to make Linux smaller?

- Remove things you don't need
- Shrink everything else....
  - the Linux kernel
  - the shared libraries
  - the utility applications
Common user applications

- coreutils, util-linux, init, sed, grep, gzip, etc
- Standard utilities are full featured, but big big big.
- Slimming them down was too much work and changes would not be accepted upstream
- Time to look for alternatives...
BusyBox
1999 to the present

- Originally
  - Used on Debian Boot Floppies
  - Maintained by Bruce Perens
  - Not quite ready for prime time...

- Became the maintainer

- Years of work go by....

- BusyBox 1.00
Looking at System Libraries

Starting in 2000, I began to seriously look at system libraries...

- The GNU C library (glibc) is huge
- Designed for servers and workstations
- Strives for features and performance
- Embedded Linux needed an alternative
Why is it called uClibc?

- The letter 'u' is short for "micro"
- The capital "C" is short for "controller"
- So uClibc is "the microcontroller C library".
- The name is somewhat historical since uClibc was originally created to support uClinux on MMU-less microcontrollers.
How can uClibc be smaller?

• Highly configurable
• Some features disabled by default
• Focus first on small size, speed second
• Aggressive refactoring of code to eliminate redundancy
Some Numbers

284K  libuClibc-0.9.27.so
20K   ld-uClibc-0.9.27.so
12K   libcrypt-0.9.27.so
8.0K  libdl-0.9.27.so
56K   libm-0.9.27.so
4.0K  libnsl-0.9.27.so
84K   libpthread-0.9.27.so
4.0K  libresolv-0.9.27.so
8.0K  libutil-0.9.27.so
480K  total

1.2M  libc-2.3.2.so
92K   ld-2.3.2.so
20K   libcrypt-2.3.2.so
12K   libdl-2.3.2.so
136K  libm-2.3.2.so
76K   libnsl-2.3.2.so
84K   libpthread-0.10.so
68K   libresolv-2.3.2.so
8.0K  libutil-2.3.2.so
1.7M  total – with more libraries and files needed....
Example -- hello.c

```c
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    printf("hello world\n");
    return 0;
}
```

<table>
<thead>
<tr>
<th>text</th>
<th>data</th>
<th>bss</th>
<th>total</th>
<th>filename</th>
</tr>
</thead>
<tbody>
<tr>
<td>939</td>
<td>256</td>
<td>4</td>
<td>2928</td>
<td>hello.glibc</td>
</tr>
<tr>
<td>407925</td>
<td>3812</td>
<td>4384</td>
<td>421820</td>
<td>hello.glibc.static</td>
</tr>
<tr>
<td>646</td>
<td>204</td>
<td>4</td>
<td>2132</td>
<td>hello.uclibc</td>
</tr>
<tr>
<td>3647</td>
<td>504</td>
<td>8300</td>
<td>5316</td>
<td>hello.uclibc.static</td>
</tr>
</tbody>
</table>