TOSHIBA
Leading Innovation

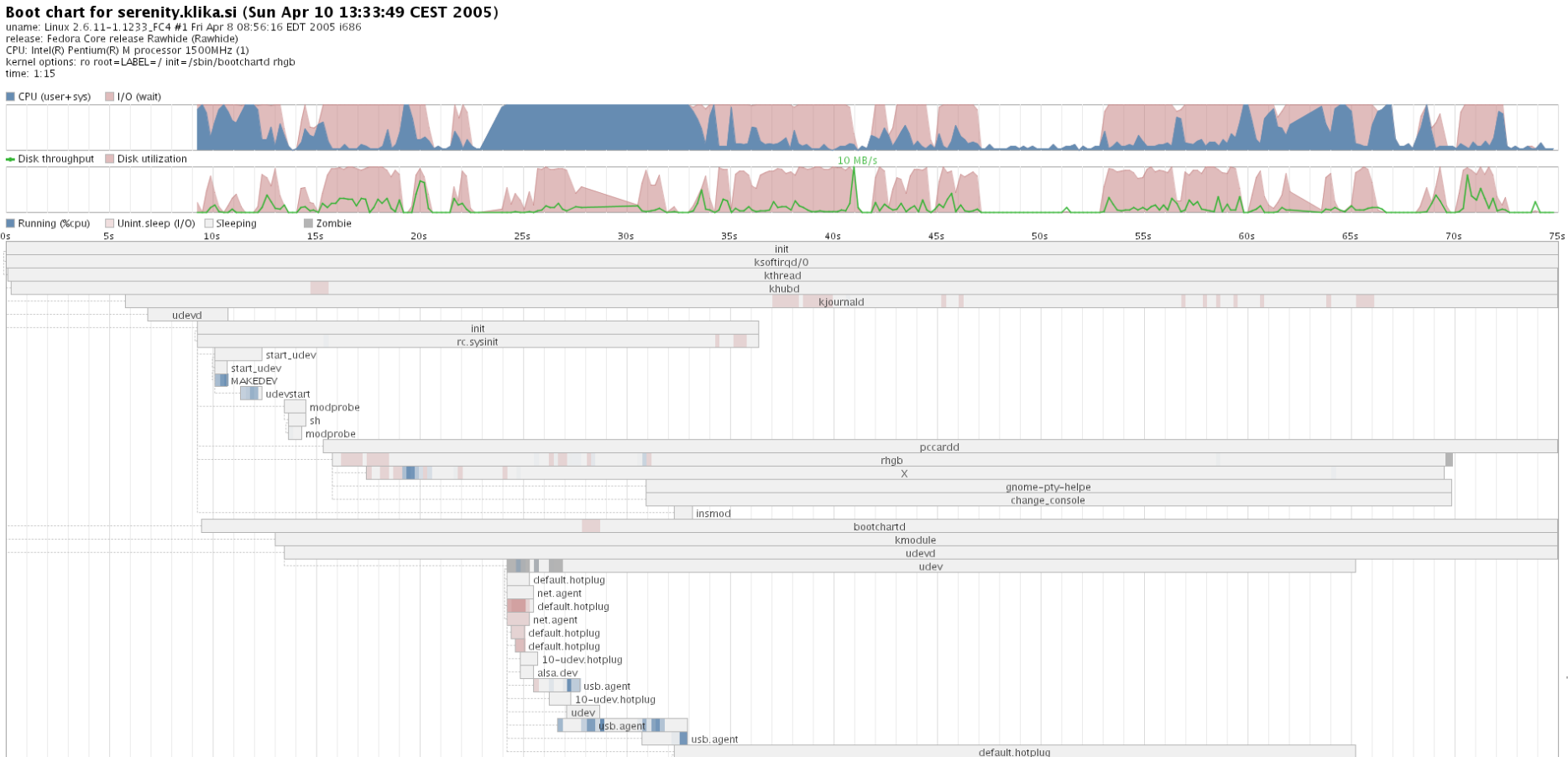
Evaluation of Busybox bootchartd

Keisuke Yasui, TOSHIBA CORPORATION

CELF Japan Technical Jamboree #33, June 4, 2010

Bootchart

- **Bootchart**
  - http://www.bootchart.org/
  - a tool for performance analysis and visualization of the GNU/Linux boot process.

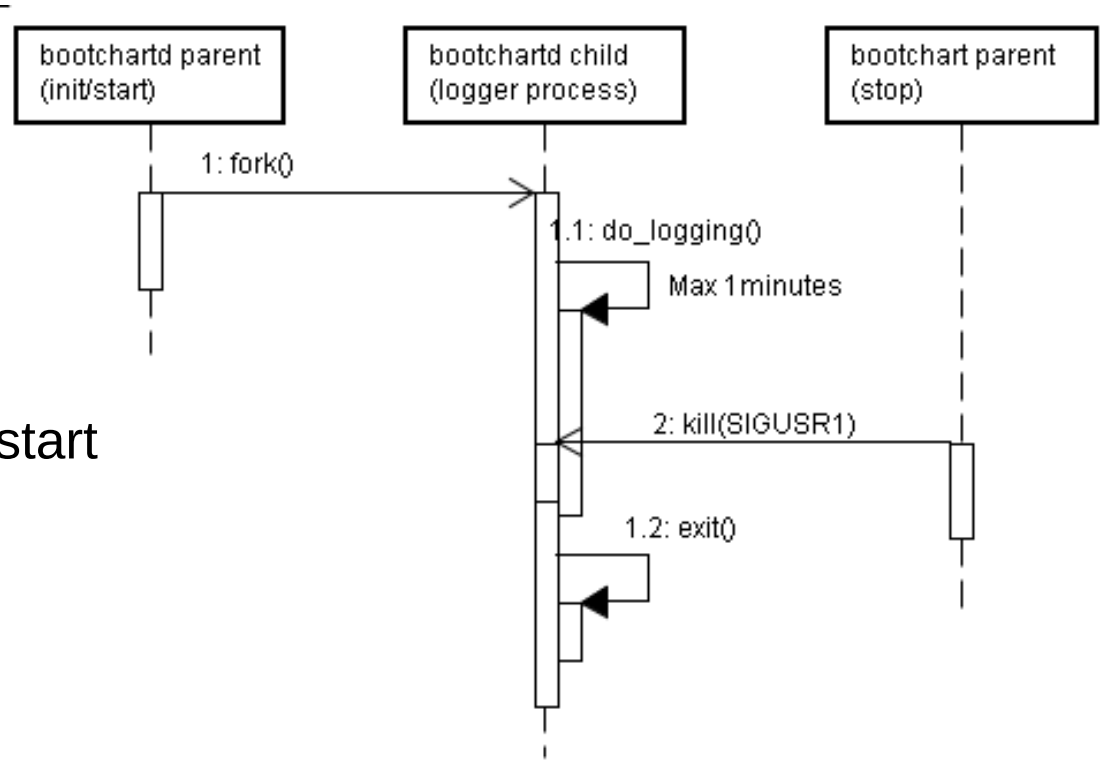


Environment

- **Bootchartd patch for busybox**
  - [PATCH] bootchartd applet
  - **Denys Vlasenko** *Tue May 4 13:41:00 UTC 2010*
    http://lists.busybox.net/pipermail/busybox/2010-May/072365.html
- **Environment**
  - Cortex-A9 433MHz
  - Compiler: gcc-4.4.2
  - Libc: eglibc-2.10.2-2
  - Binutils: binutils-2.20
  - Linux kernel: linux-2.6.28.10-arm2
  - Busybox: busybox-1.15.3
  - Visualization tool：bootchart-0.9
    - http://prdownloads.sourceforge.net/bootchart/bootchart-0.9.tar.bz2

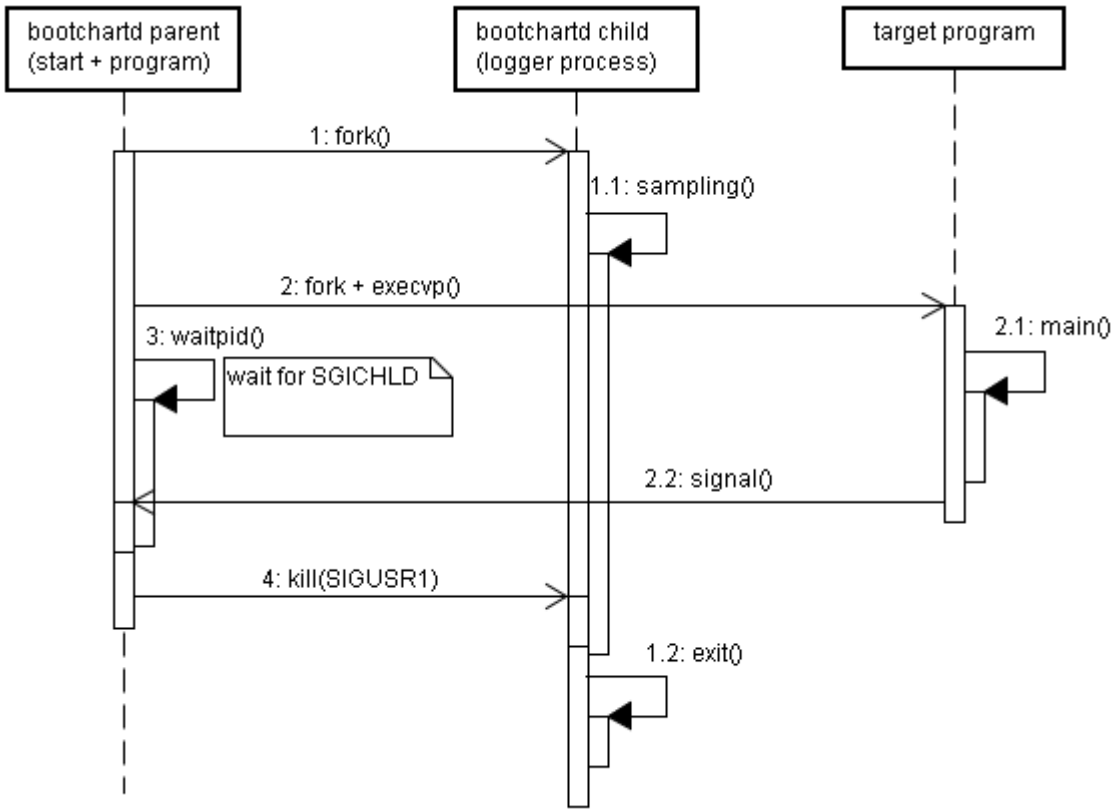
bootchartd start / stop

- **start/stop**
  - start
    - # bootchartd start
  - stop
    - # bootchartd stop
    - or 1minutes after start



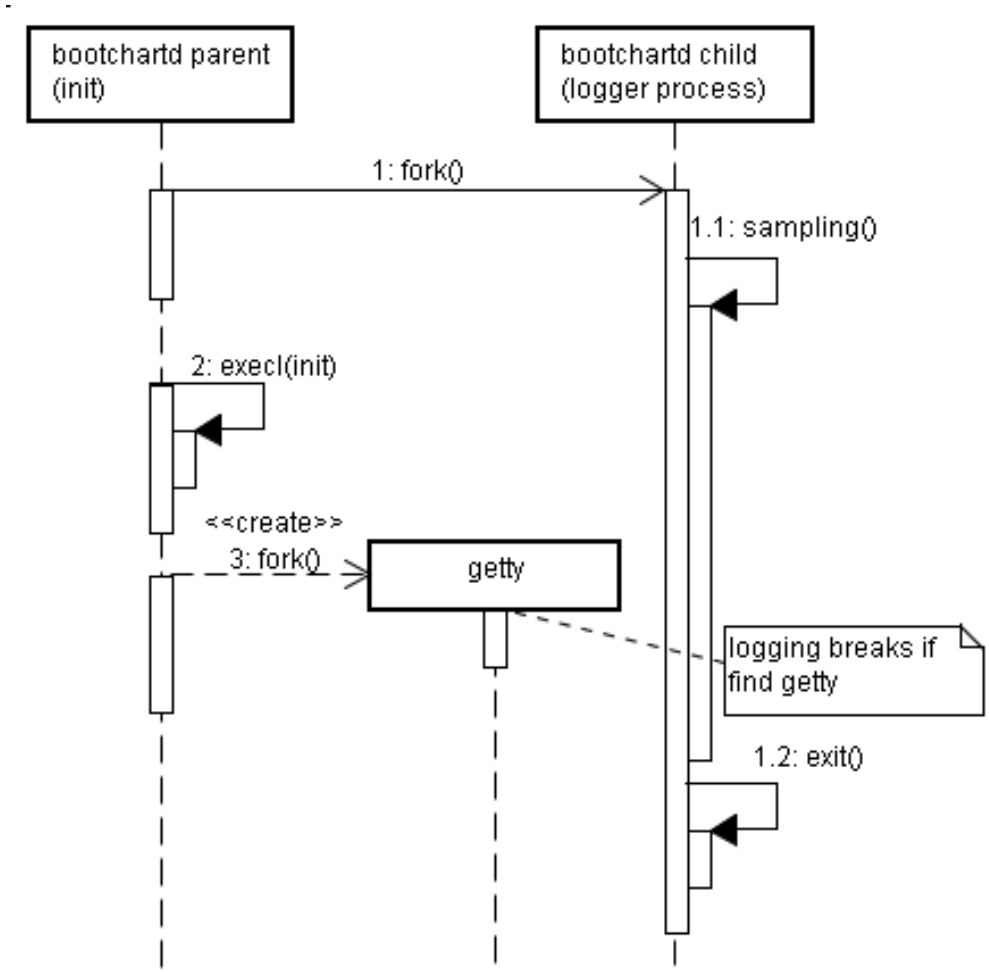
bootchartd with program

- **program**
  - Run program with logging.
    - # bootchartd start {path of program} {args}



bootchartd as init

- **with initramfs**
  - rdinit=/sbin/bootchartd
- **with cramfs**
  - init=/sbin/bootchartd
- **Selecting init**
  - bootchart_init=xxx
  - default: /init /sbin/init
- **Auto stop**
  - console + 2 sec



Problems

- **Illegal timeout value (fixed in repository)**
  - timeout after start.
    - current: 60*1000*1000 / 200*1000 = 60,000,000 seconds
    - correct: 60*1000*1000 /(200*1000) = 60 seconds
  - timeout after getty.
    - current: 2*1000*1000 / 200*1000 = 2,000,000 seconds
    - correct: 2*1000*1000 /(200*1000) = 2 seconds
- **Sampling rate is 200ms only**
- **No header file**
  - original bootchartd creates header including the following data,
    - version
    - title
    - system.uname
    - system.release
    - system.cpu
    - system.kernel.options

Problems

- **Cannot recognize theads**
- **Process/Thread informations under /proc has changed for multi thread.**
- **Getting stat information for only main thread.**
- **/proc in 2.6.20 is below**

/proc/401

+- stat : main thread

+--task

+—401

| +—stat : main thread

| | ...

+—402

| +—stat : 2nd thread

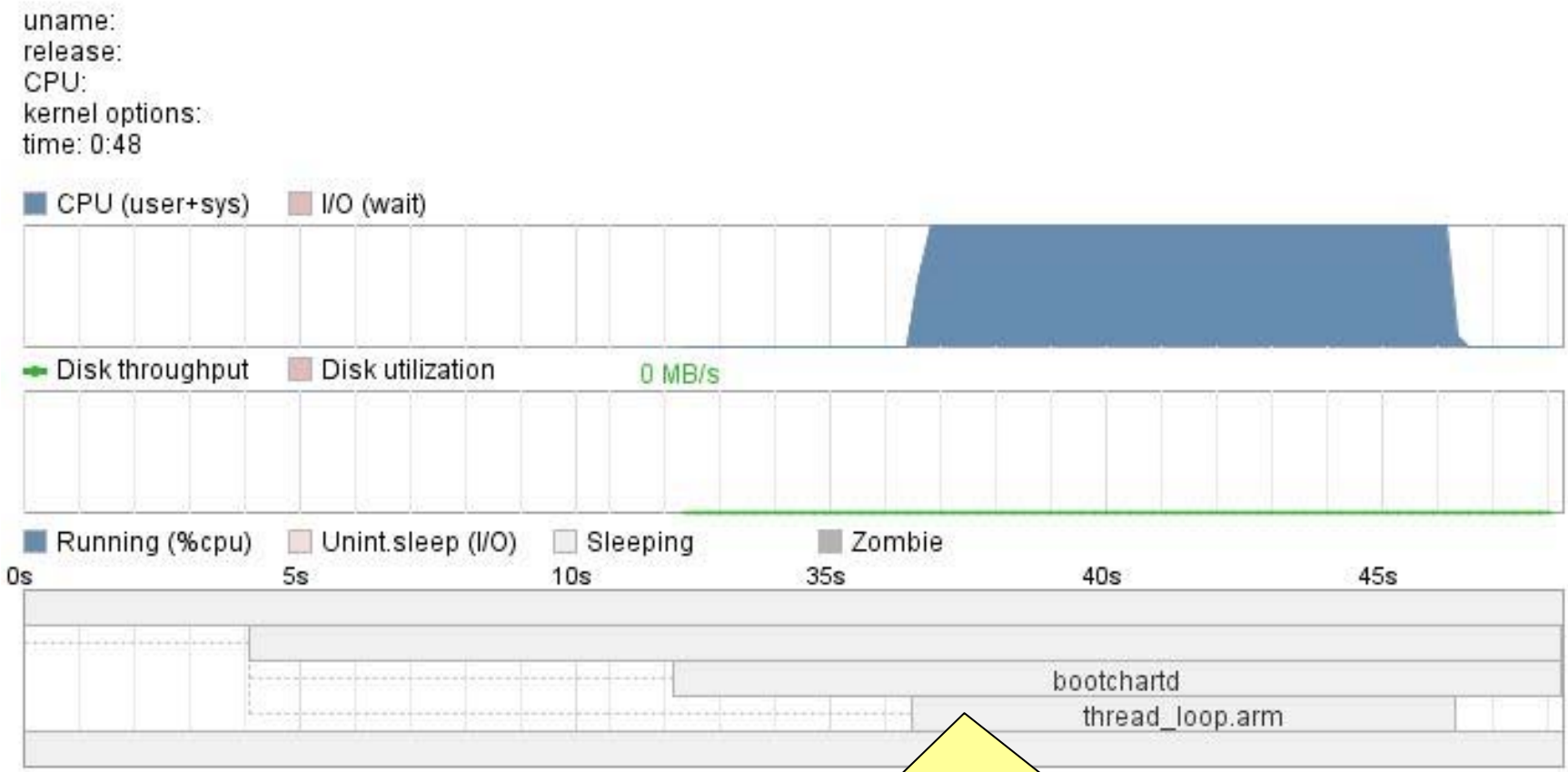
| | ...

+—403

+—stat : 3rd thread

| ...

Problems



2nd Thread is running in busy loop

TOSHIBA

Leading Innovation