Contribute To Linux Mainline

Wu Zhangjin / Falcon
wuzhangjin@gmail.com

Tiny Lab – 泰晓实验室
http://tinylab.org

June 3, 2013

Outline

About Linux Kernel Development

- Linux Introduction
- Development Model
- Release Frequency
- Rate of Change
- Kernel Source Size
- Who is Doing the Work
- Who is Sponsoring the Work
- Linux Kernel Patch Statistic

NOTE: this "Linux kernel development" part refers to the following paper

http://www.linuxfoundation.org/publications/linuxkerneldevelopment.php

Linux Introduction

- Launched at 1991
- The original author: Linus
- Linux is the kernel itself, don't forget GNU project...
- http://www.kernel.org
- http://git.kernel.org
- linux-kernel@vger.kernel.org

From 2.6.x series,

- Time-based release model
- Releases happen every 2-3 months
- Include new features and internal API changes
- Relesae type:
    - mainline: 3.xx/-rcY
    - stable: 3.xx.yy
    - longterm (stable): 3.xx.yy, 2.6.xx.y
    - linux-next (merge conflicts fixup): next-YEARMONTHDAY

Release Types

- mainline: 3.xx/-rcY, Linus Torvalds

  git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git

- stable and longterm: 3.xx.yy, Greg Kroah-Hartman and others

  git://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable.git

- linux-next, Stephen Rothwell

  git://git.kernel.org/pub/scm/linux/kernel/git/next/linux-next.git

- experimental patchsets
  - -mm, Andrew Morton

    http://www.ozlabs.org/ akpm/mmotm/
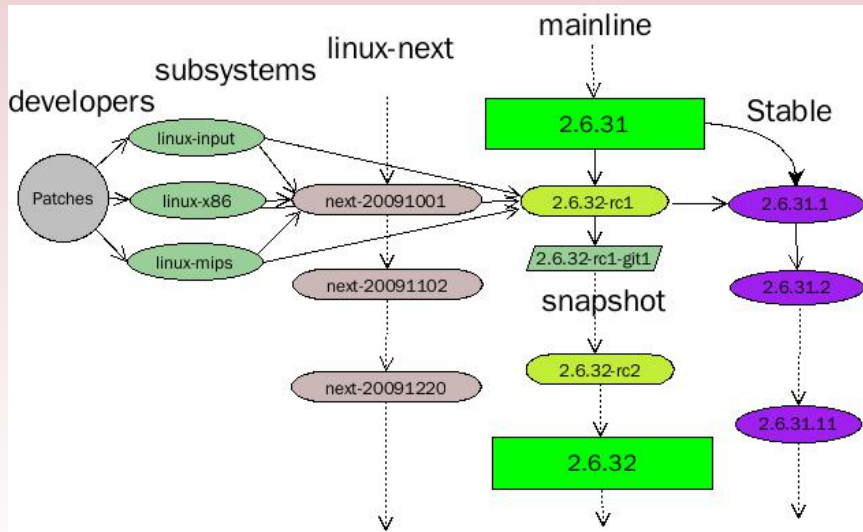
  - patchwork

    http://patchwork.kernel.org/

Development Model(cont.)

Figure: Linux Kernel Release Cycle

Kernel Version	Release Date	Days of Development
2.6.11	2005-03-02	69
2.6.12	2005-05-17	108
2.6.13	2005-08-28	73
2.6.14	2005-10-27	61
2.6.15	2006-01-02	68
2.6.16	2006-03-19	77
2.6.17	2006-06-17	91
2.6.18	2006-09-19	95
2.6.19	2006-11-29	72
2.6.20	2007-02-04	68
2.6.21	2007-04-21	81
2.6.22	2007-07-08	75
2.6.23	2007-10-09	94
2.6.24	2008-01-24	108

Figure: Frequency of kernel releases
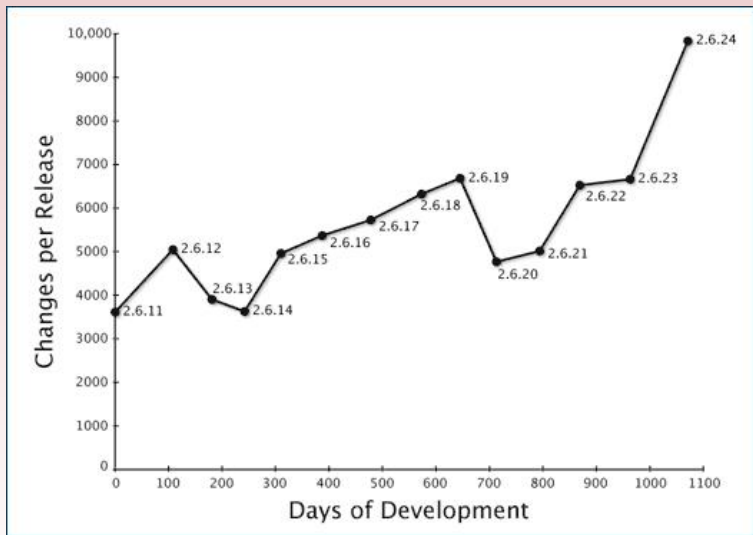
Rate of Change

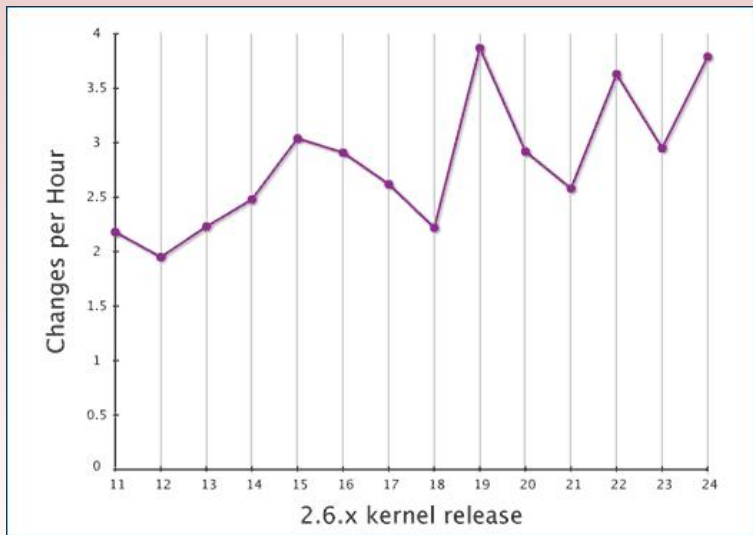Figure: Changes per kernel release

Rate of Change

Figure: Changes per hour by kernel release
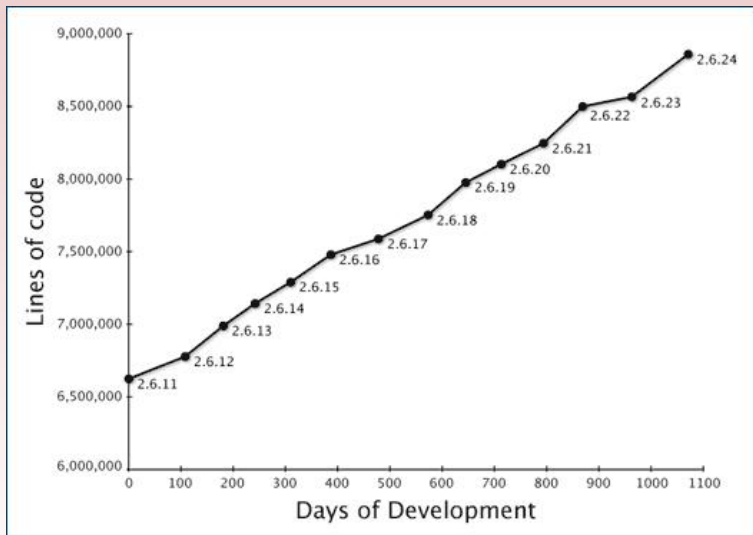
Kernel Source Size

Figure: Size per kernel release

Kernel Source Size
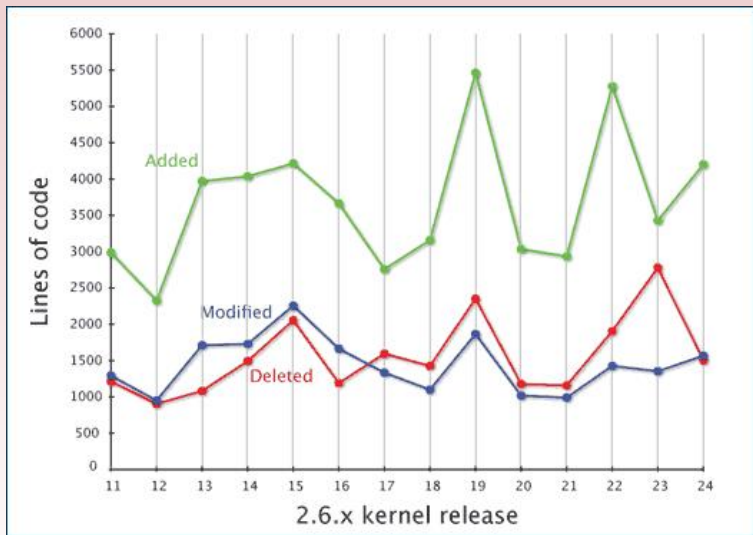
Figure: Rate of change by kernel release

Kernel Version	# of Developers	# of Known Companies
2.6.11	483	71
2.6.12	701	90
2.6.13	637	91
2.6.14	625	89
2.6.15	679	96
2.6.16	775	100
2.6.17	784	106
2.6.18	897	121
2.6.19	878	126
2.6.20	728	130
2.6.21	834	132
2.6.22	957	176
2.6.23	991	178
2.6.24	1,057	186
All	3,678	271

Figure: Number of individual developers and employers

Name	# of Changes	% of Total Changes
Al Viro	1571	1.9%
David S. Miller	1520	1.8%
Adrian Bunk	1441	1.7%
Ralf Baechle	1346	1.6%
Andrew Morton	1222	1.5%
Andi Kleen	993	1.2%
Takashi Iwai	963	1.2%
Tejun Heo	938	1.1%
Russell King	926	1.1%
Stephen Hemminger	920	1.1%
Thomas Gleixner	754	0.9%
Patrick McHardy	740	0.9%
Ingo Molnar	735	0.9%
Trond Myklebust	664	0.8%
Neil Brown	646	0.8%
Randy Dunlap	645	0.8%
Jean Delvare	617	0.7%
Jeff Garzik	615	0.7%
Christoph Hellwig	615	0.7%
David Brownell	588	0.7%
Paul Mundt	581	0.7%
Alan Cox	571	0.7%
Jeff Dike	558	0.7%
Herbert Xu	538	0.6%
David Woodhouse	503	0.6%
Greg Kroah-Hartman	496	0.6%
Linus Torvalds	495	0.6%
Dmitry Torokhov	494	0.6%
Alan Stern	478	0.6%
Ben Dooks	477	0.6%

Figure: Individual kernel contributors

Who is Sponoring the Work

Company Name	# of Changes	% of Total
None	11,594	13.9%
Unknown	10,803	12.9%
Red Hat	9,351	11.2%
Novell	7,385	8.9%
IBM	6,952	8.3%
Intel	3,388	4.1%
Linux Foundation	2,160	2.6%
Consultant	2,055	2.5%
SGI	1,649	2.0%
MIPS Technologies	1,341	1.6%
Oracle	1,122	1.3%
MontaVista	1,010	1.2%
Google	965	1.1%
Linutronix	817	1.0%
HP	765	0.9%
NetApp	764	0.9%
SWsoft	762	0.9%
Renesas Technology	759	0.9%
Freescale	730	0.9%
Astaro	715	0.9%
Academia	656	0.8%
Cisco	442	0.5%
Simtec	437	0.5%
Linux Networx	434	0.5%
QLogic	398	0.5%
Fujitsu	389	0.5%
Broadcom	385	0.5%
Analog Devices	358	0.4%
Mandriva	329	0.4%
Mellanox	294	0.4%
Snapgear	285	0.3%

Figure: Companies working toward the improvement of the kernel

Linux Kernel Patch Statistic

- WebSite

  http://www.remword.com/kps_result/index.php

Upstream your source code

- Clone a git tree
- Code Reading
- Coding Style
- SCM
- Git Tips
- Code Test & Debug
- Generate & Release your patches
- Incorporate with feedbacks & Resend
- Bug Report

Clone a git tree

- Linux-official git tree

  git://git.kernel.org/pub/scm/ + project path

- Subsystem specific git tree
  E.g. MIPS-specific git tree:

  git://git.linux-mips.org/pub/scm/linux

- Clone a git tree, we use linux-mips as an example

  git clone git://git.linux-mips.org/pub/scm/linux linux-mips

- Checkout the latest development branch

  git remote add origin git://git.linux-mips.org/pub/scm/linux

  git checkout -b branch-name --track origin/master

- Document
  Documentation/
  make help |grep doc
- Tools
    - vim + cscope + ctags
      :help tag
      :help cs-find
    - LXR: http://lxr.linux.no/linux

- Document
  Documentation/CodingStyle
- Tools
  indent -linux source-file
  :gqap

- SCM tools
  switch cvs, svn to git

- git & gitk & git-gui
  init, fetch, pull, push, cherry-pick, merge, gc
  history: git log
  status: git status
  basic op: add, commit, rm, mv
  patch: diff, apply, am, format-patch, send-email

Git Tips

- Merge another branch of the same git-repo

  git merge branch-name

- Pick up a specific commit from another branch

  git cherry-pick commit-name

- Merge a branch from another local git repo

  git pull /path/to/git-repo branch-name

- Rebase your commits

  git rebase -i commit-name^

- Rebase a branch onto master

  git rebase --onto master --root

- Delete a remote branch

  git push origin :branch

Recommendation: Get more tips from 'Git Usage: Hello, The Git World', http://www.tinylab.org/git-usage-hello-the-git-world/

Code Test & Debug

- Document
  Documentation/SubmitChecklist

- Test
  compile with sparse, coccicheck, smatch ...
  run with lockdep detection, kmemleak detection ...
  module: insmod, rmmod
  check with Ftrace, Perf

- Debug
  printk, WARN_ON, BUG_ON, kgdb, kernel hacking
  Virtual Machine. E.g. qemu
  hardware-based, E.g. watchpoint, breakpoint

Generate & Release your patches

- Generate patches
  git format-patch -s –cover-letter –thread commit1^..commit2 –cc="Name <user@example.com>"

- Check&fix up your patches
  Documentation/SubmitChecklist
  script/checkpatch.pl,cleanfile

- Release your patches
  Config send-email: git config, man git-send-email(Sending)
  Get maintainers: ./scripts/get_maintainer.pl
  Send out your patch:

  git send-email –to="LKML <linux-kernel@vger.kernel.org>" –cc="Maintainers' Email Address"

- Who give feedback
  Mailing lists, http://vger.kernel.org/vger-lists.html
  E-mails listed by ./scripts/get_maintainer.pl
- Feedbacks
  Test Report. E.g. Don't work on x86
  Review, E.g. something is wrong.

- Incorporate with feedbacks
  Fix up patches and make a revision
- Resend with feedback information
  Tested-by: ...
  Reviewed-by: ...
  Signed-off-by: ...

Bug report

- Kernel Bug Tracker

  http://bugzilla.kernel.org/

- Using Bugzilla

  http://bugzilla.kernel.org/docs/en/html/using.html

- Linux kernel Development:

  http://www.linuxfoundation.org/publications/linuxkerneldevelopment.php

- Linux Kernel Development: Getting Started
- Documentation/SubmittingPatches
- Documentation/SubmittingDrivers
- Documentation/SubmitChecklist
- http://vger.kernel.org/vger-lists.html

Contribute to GNU Toolchain

- Coding Style: totally differs from Linux
  indent -gnu
- Mailing list
  Gcc: http://gcc.gnu.org/lists.html
  Binutils: http://www.gnu.org/software/binutils/
- Git Repository or Git Mirror
  Gcc: git://gcc.gnu.org/git/gcc.git
  Binutils: git://github.com/bneumeier/binutils.git
- Bug Tracking
  Gcc: http://gcc.gnu.org/bugzilla/
  Binutils: http://sourceware.org/bugzilla/

Upstream your patch

- Make your patch acceptable
    - Cleanup your patch and test it
    - Send it to the maintainer of the subsystem
    - Incoporate with the feedbacks
- Assign your copyright to FSF

Thank You!