LTSI Project update

Long Term Support Initiative

Tsugikazu SHIBATA, NEC

23, March 2015

Embedded Linux Conference @ San Jose

Who am I

Tsugikazu Shibata, NEC

Project lead of LTSI

Member of CE Working group

Member of the board of the Linux Foundation

Key activities of LTSI

Long Term Support Initiative

- □ Provide a industry managed kernel and maintain Long Term
- □ Provide a common place for embedded industry to share information: Mailing list, Workshop …
- □ Provide place to support upstream activity : Help industry developer

- LTSI had been started 2011 in Prague
- After long discussion of industry demand and how to solved

Contents

- Linux Development process and status
- LTSI current status
- LTSI TEST Project

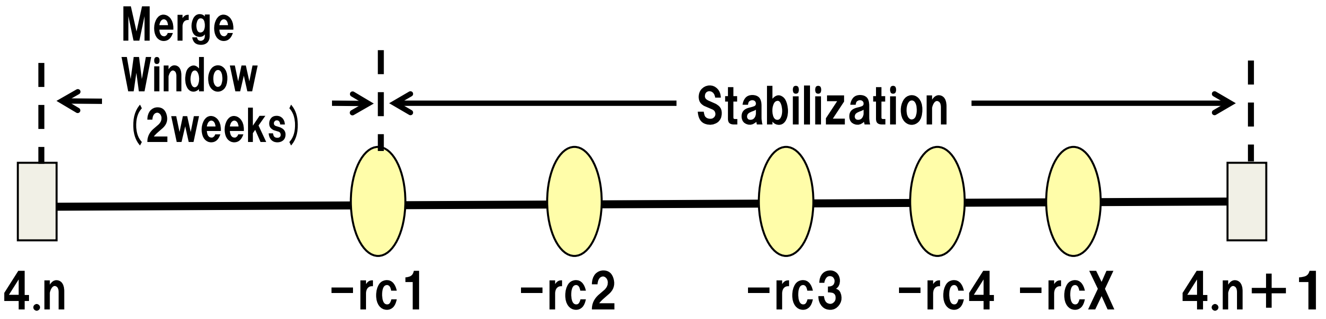
Status of Latest Linux Kernel

- Latest released Kernel : 3.19
  - Released February 10th
  - Lines of code : 19,130,705
  - Files : 48,425
- Current Stable Kernel: 3.19.2
- Current development kernel: 4.0-rc4

Next version is decided to be 4.0!

Linux Development process

- Just after the release of 4.n, two weeks of merge window will be opened for proposal of new features
- After 2 weeks of merge window, -rc1 will be released and the stabilization will be started
- 4.n+1 will be released when it becomes reasonably stable by some of -rcX released



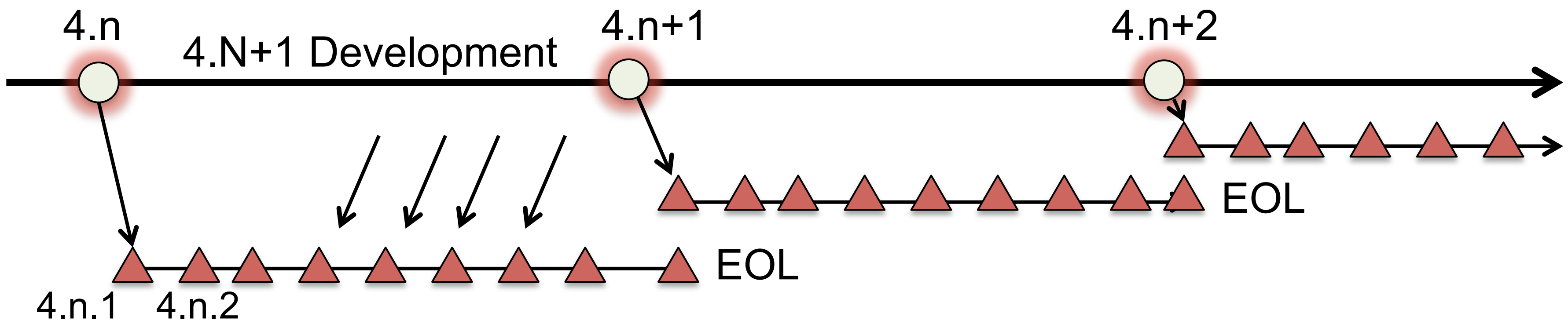
Kernel release cycle

- Release cycle of Linux kernel is about 65days

Version	Release	Duration
3.0	2011-7-21	
3.1	2011-10-24	95
3.2	2012-1-4	72
3.3	2012-3-18	74
3.4	2012-5-20	63
3.5	2012-7-21	62
3.6	2012-9-30	71
3.5	2012-12-10	71
3.8	2013-2-18	70
3.9	2013-4-28	69

Version	Release	Duration
3.10	2013-6-30	63
3.11	2013-9-2	64
3.12	2013-11-15	74
3.13	2014-1-21	67
3.14	2014-3-30	68
3.15	2014-6-8	70
3.16	2014-8-3	56
3.17	2014-10-5	63
3.18	2014-12-7	63
3.19	2015-2-9	64

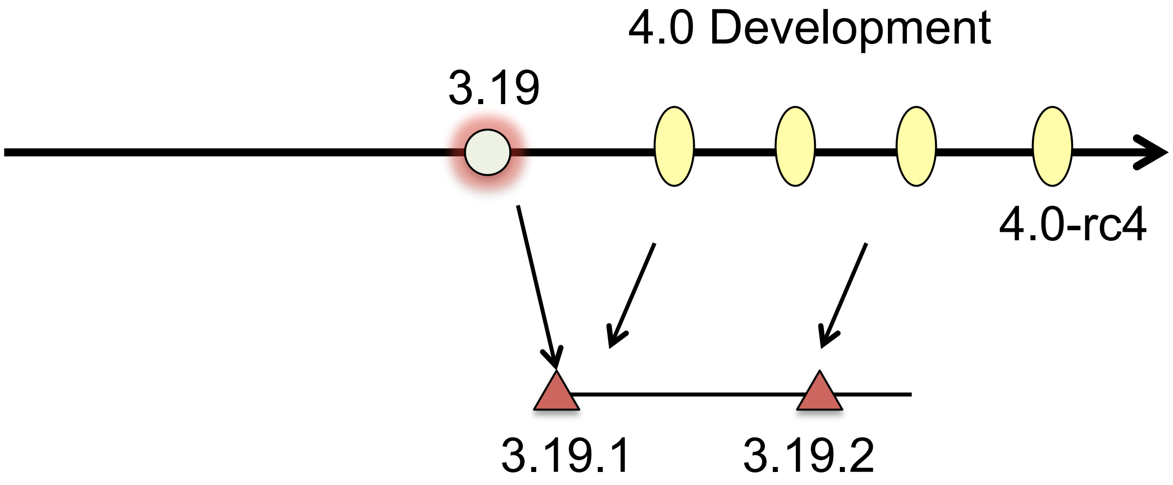
Stable kernel release



- Recommended branch for users who want the most recent stable kernel
- 3 part version like 4.n.m
- Contain small and critical fixes for security problems or significant regressions discovered in a latest development version
- Becomes End Of Line when next stable kernel were released

Status of Latest Linux Kernel Again

- Latest released Kernel : 3.19
- Current Stable Kernel: 3.19.2
- Current development kernel: 4.0-rc4



Stable_kernel_rules.txt

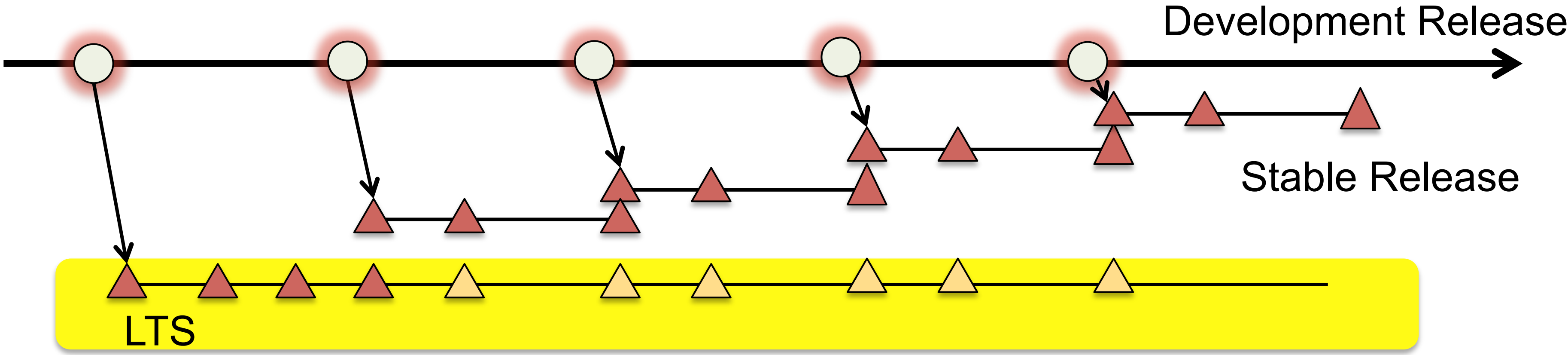
- Published in 2006. Strict rule to back port from latest version

See Documentation/stable_kernel_rules.txt

- It must be obviously correct and tested.
- It cannot be bigger than 100 lines, with context.
- It must fix only one thing.
- It must fix a real bug that bothers people.
- It must fix a problem that causes a build error, an oops, a hang, data corruption, a real security issue, In short, something critical.
- Serious issues as reported by a user may also be considered.
- New device IDs and quirks are also accepted.
- No "theoretical race condition" issues, unless an explanation of how the race can be exploited is also provided.
- It cannot contain any "trivial" fixes in it .
- It or an equivalent fix must already exist in Linus' tree (upstream). ……

LTS: LongTerm Stable Kernel

- Extended maintenance period for stable kernel
- Pick one version per year and maintain 2 years
- Bug and security fixes found in latest version are continued to back ported



Current LTS versions

Version	Maintainer	Released	Projected EOL
3.18	Sasha Levin	2014-12-07	Jan, 2017
3.14	Greg Kroah-Hartman	2014-3-30	Aug, 2016
3.12	Jiri Slaby	2013-11-3	2016
3.10	Greg Kroah-Hartman	2013-6-30	Sep, 2015
3.4	Li Zefan	2012-5-20 (2014-9-4)	Sep, 2016
3.2	Ben Hutchings	2012-1-4	2016
2.6.32	Willy Tarreau	2009-12-3	Mid-2015

https://www.kernel.org/category/releases.html

LTS includes large number of fixes

- 600 – 700 fixes included in a Stable release
- LTS include fixes multi number of stable releases

Version		# of commits
From	To	
3.0	**3.0.101**	**3953 (EOL)**
3.1	3.1.10	695 (EOL)
3.2	**3.2.68**	**5829**
3.3	3.3.8	698 (EOL)
3.4	**3.4.106**	**5033**
3.5	3.5.7	816 (EOL)
3.6	3.6.11	757 (EOL)
3.7	3.7.10	718 (EOL)
3.8	3.8.13	996 (EOL)
3.9	3.9.11	746 (EOL)

Version		# of commits
From	To	
3.10	**3.10.72**	**4312**
3.11	3.11.10	677 (EOL)
3.12	**3.12.26**	**4262**
3.13	3.13.11	903 (EOL)
3.14	**3.14.36**	**2978**
3.15	3.15.10	703 (EOL)
3.16	3.16.7	871 (EOL)
3.17	3.17.8	884 (EOL)
3.18	**3.18.9**	**775**
3.19	3.19.2	359

As of 2015/3/22

Number of yearly commits on LTS

LTS Version	2012	2013	2014	2015
3.4	1636	1769	1233	
3.10		1724	2060	336
3.14			2401	500

1700 ~ 2000 of commits were back ported from latest upstream version into LTS.

What about new features

LONG TERM SUPPORT INITIATIVE

- New features or new drivers are always merging into latest upstream
  - That is Linux kernel development process
- LTSI is making a chance to merge new features
  - Enhanced drivers
  - New features merged in upstream
  - New tools related to kernel; Ktap, LTTng

Value of LTSI

- LTSI kernel
  - Be able to use large number of fixes by LTS
  - Be able to add vendor required feature patches
    - Already in upstream or at least maintainer accepted patches
- Share information among companies
- Get help for up-streaming from industries' developer

Difference between LTS and LTSI

LONG TERM SUPPORT INITIATIVE

LTS

- □ Release 1 version / year, Maintain 2 years
- □ Frequently and large number of bug /security fixes

LTSI

- □ Release 1 version / year, Maintain 2 years
- □ Frequently and large number of bug /security fixes
- □ Add vendor required features
- □ Huge testing by contributors
- □ Auto test frame work
- □ Share status, info, problem
- □ Provide help to developer for upstream

3.14 LTSI kernel history

Items	Date
kernel 3.14 merge window open kernel	2014.1.20
3.14 merge window close	2014.2.2
kernel 3.14 release	2014.3.30
3.14 becomes LTS (=3.16 release)	2014.8.3
Announce of 2014 LTSI kernel version	2014.8.4
LTSI-3.14 merge window open	2014.8.28
patch collection period	78 days
LTSI-3.14-rc1 (=merge window close)	2014.11.14
validation period	56 days
LTSI-3.14 release	2015.1.9

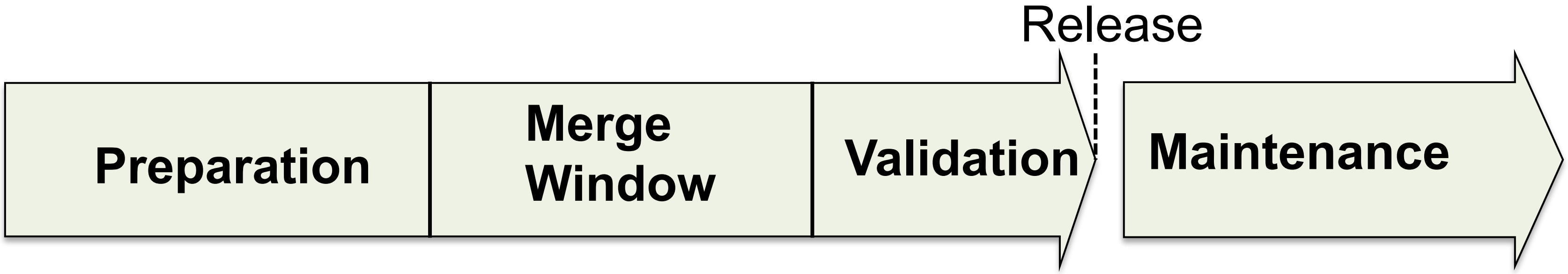
Achievements for LTSI 3.14

- Common:
  - ktap 0.4, a lightweight script-based dynamic tracing tool
- x86:
  - Intel Low Power Sub-system (LPSS) PWM, found in Intel Baytrail & Braswell SoC.
  - I2C/SMBus i801, added PCI Device ID for Intel Baytrail, Braswell & Intel Wildcat Point (PCH).
  - supports PCI enumeration for Intel Braswell SoC and Intel 9 Series PCH
  - iTCO v3 support for Intel Avoton SoC.
  - GPIO support for Intel Avoton SoC, Panther Point PCH and NM10 chipset
- ARM:
  - Renesas R-CarH2 Lager Board and R-CarM2 Koelsch board

LTSI version for this year?

LTSI DEVELOPMENT PROCESS

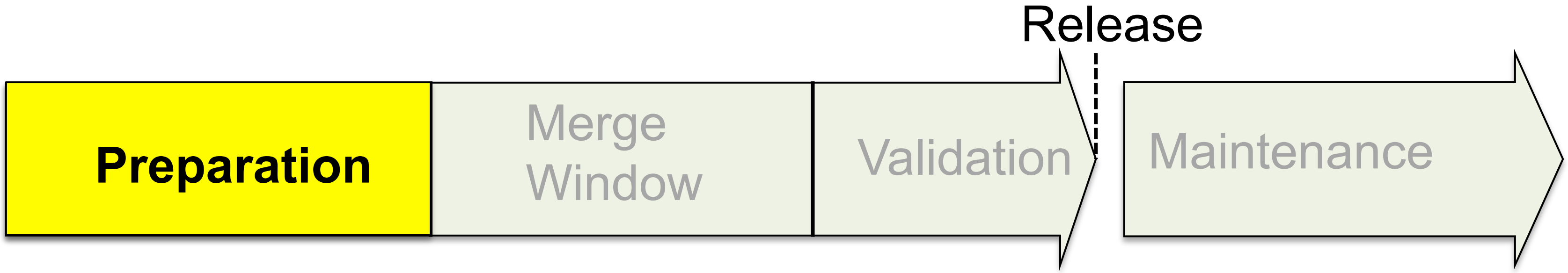
LTSI development process



Discussing about process as;

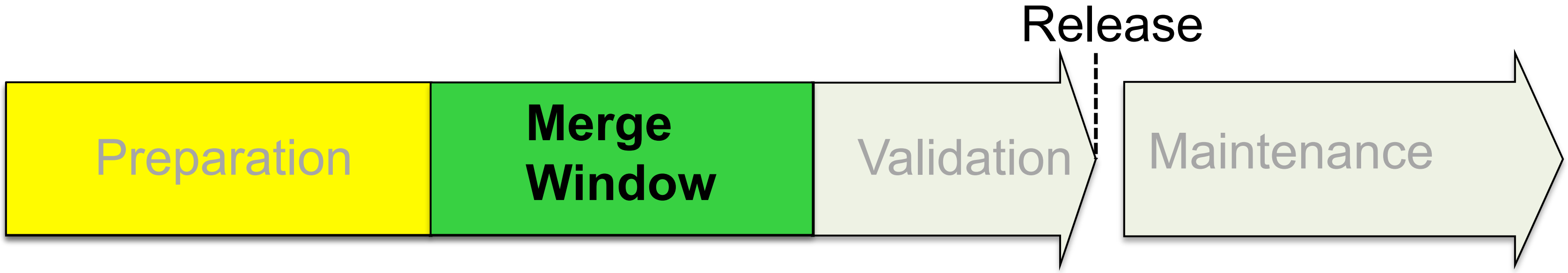
- Preparation
- Merge window
- Validation
- Maintenance

LTSI Process: Preparation



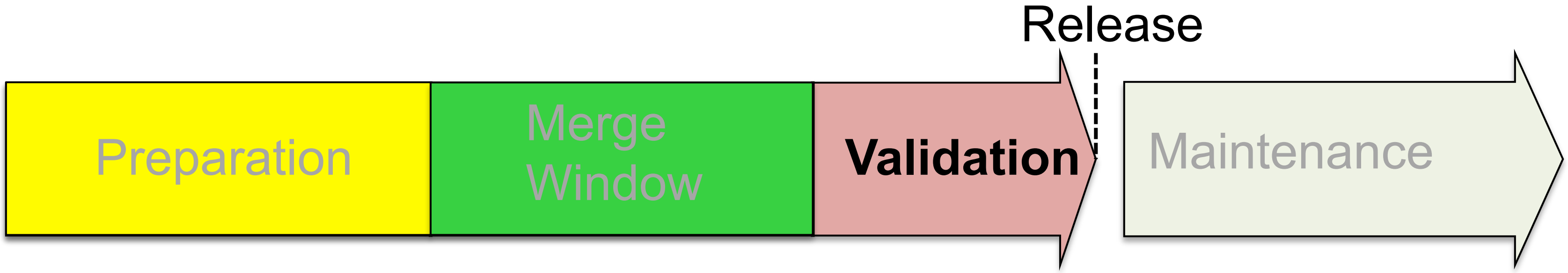
- Decide LTS version
- Decide LTSI development schedule
- Before the merge window open, developer should;
  - Merge own patches into upstream to prepare backport into LTSI. That will be easier to maintain production kernel (ie. Driver)
  - Prepare to back port important features in the upstream to LTSI

LTSI Process: Merge window



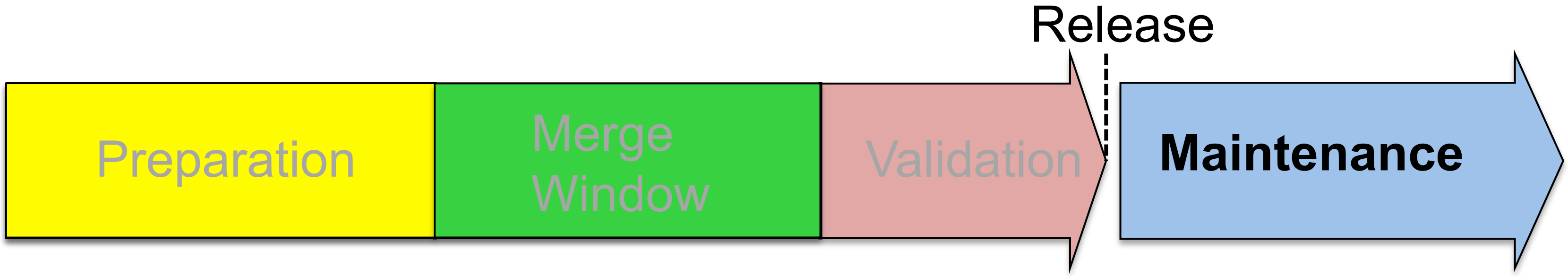
- Merge window will open after 4-5 month of LTS release
- Anyone can propose patches in the merge window period
- Patches should be in the upstream or at least in the –next
- Exceptional case: beneficial for the wide range of users, or projected to be in the upstream
- Merge window period is about two month and –rc1 will be released right after the window was closed
- No additional patches are accepted after the window closed

LTSI process: Validation



- After release of –rc1, developer should validate whether its features are correctly merged and worked fine
- Patches to fix problem should be sent
- Validation period is about a month or more
- LTSI development will be finished when maintainer recognized reasonably stable and then new LTSI kernel will be released

LTSI Process: Maintenance



- LTSI will be rebased at the same time each new LTS was released
  - If you want to create own kernel based on LTS such as 3.10.x, you will be able to find same version in LTSI git tree
  - There are tool to create LTSI full kernel source in “scripts” directory
- LTSI full source tarball will be released every 3month after QA team have done its test as a volunteer work

LTSI: TEST PROJECT

LTSI Test : Background

- LTS is quite stable because of picking patches with strict rule (stable_kernel_rules.txt)
- LTSI includes industry demanded non mainlined code

We want to validate LTSI kernel that does not include any bug or regression against LTS

LTSI Test : Share test case!

- Many companies may spend a lot of time for kernel validation by their own test phase
- Some of "fundamental test" might be duplicated and such portion might be shared

We can consider sharing some part of kernel test across the industry

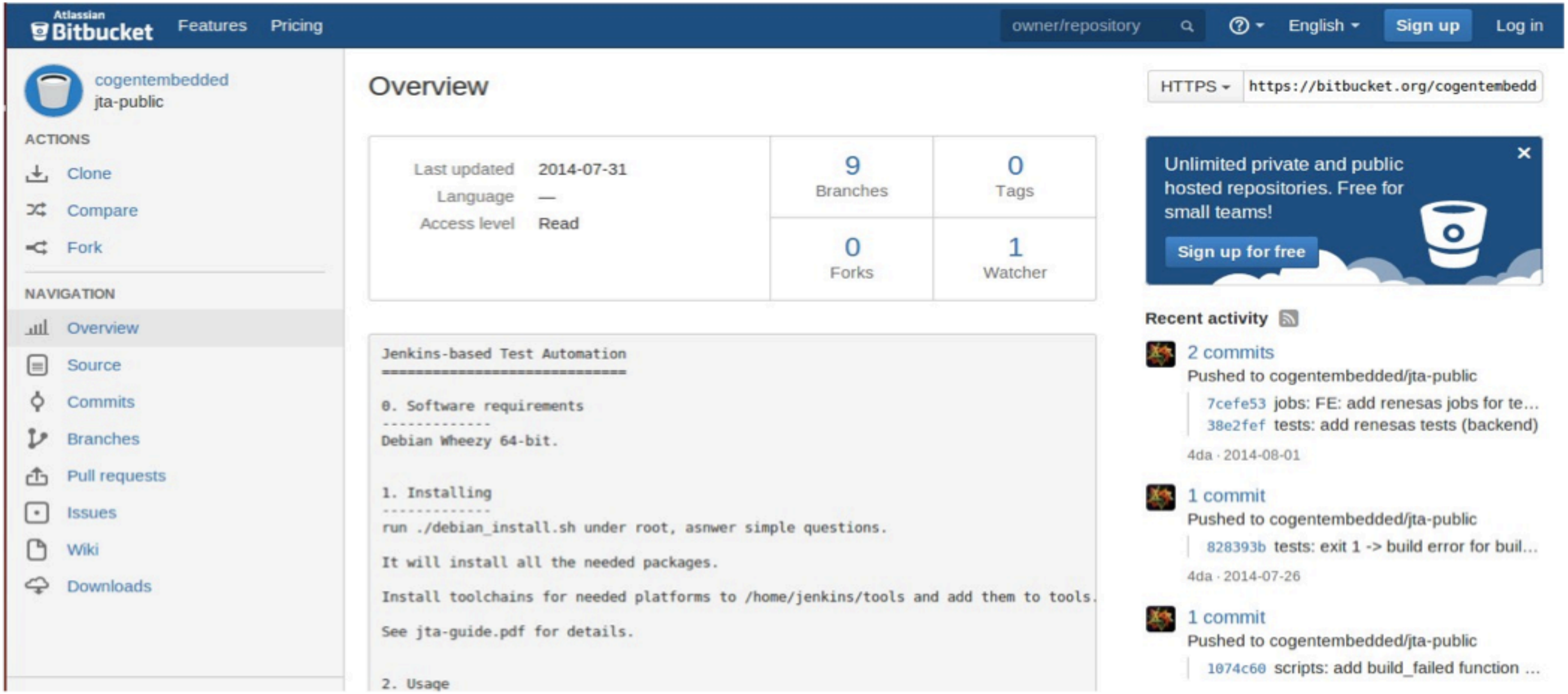
- **Test environment**
- **Test case**

LTSI Test environment

- We have developed LTSI test environment
  - Fully automated execution
  - Easy to manage
  - Monitoring what have happened
  - Web based GUI
  - Deliverable (Be able to install every servers)
  - 28 benchmarks and 33 test programs are integrated in our test environment

LTSI Test environment

Current stage is just started but You can down load from:
https://bitbucket.org/cogentembedded/jta-public/



Overview

Last updated	2014-07-31
Language	—
Access level	Read

9 Branches	0 Tags
0 Forks	1 Watcher

```
Jenkins-based Test Automation
================================

0. Software requirements
-------------
Debian Wheezy 64-bit.


1. Installing
------------
run ./debian_install.sh under root, asnwer simple questions.

It will install all the needed packages.

Install toolchains for needed platforms to /home/jenkins/tools and add them to tools.

See jta-guide.pdf for details.


2. Usage
```

Recent activity

2 commits
Pushed to cogentembedded/jta-public
7cefe53 jobs: FE: add renesas jobs for te...
38e2fef tests: add renesas tests (backend)
4da · 2014-08-01

1 commit
Pushed to cogentembedded/jta-public
828393b tests: exit 1 -> build error for buil...
4da · 2014-07-26

1 commit
Pushed to cogentembedded/jta-public
1074c60 scripts: add build_failed function ...

https://bitbucket.org/cogentembedded/jta-public/

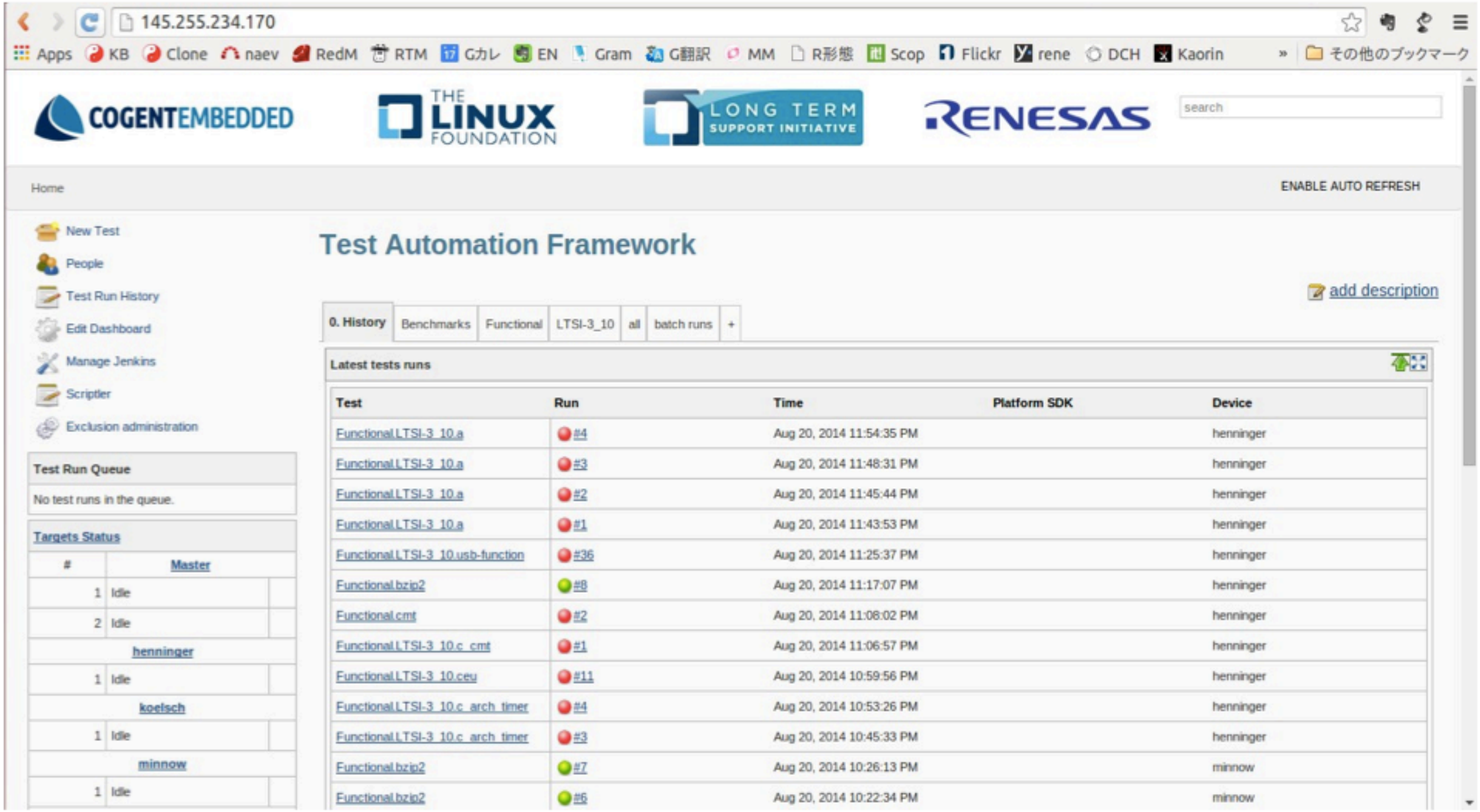
LTSI Test environment

- Initial documentation:

  https://bytebucket.org/cogentembedded/jta-public/raw/7cefe53a09b5028bf2c99663d81ecde39b486713/docs/jta-guide.pdf

Your feed back is welcome!

You will be able to refer its result



Test Automation Framework

Test Run Queue

No test runs in the queue.

Targets Status

#	Master
1	Idle
2	Idle
	henninger
1	Idle
	koelsch
1	Idle
	minnow
1	Idle

Latest tests runs

Test	Run	Time	Platform SDK	Device
Functional.LTSI-3_10.a	🔴 #4	Aug 20, 2014 11:54:35 PM		henninger
Functional.LTSI-3_10.a	🔴 #3	Aug 20, 2014 11:48:31 PM		henninger
Functional.LTSI-3_10.a	🔴 #2	Aug 20, 2014 11:45:44 PM		henninger
Functional.LTSI-3_10.a	🔴 #1	Aug 20, 2014 11:43:53 PM		henninger
Functional.LTSI-3_10.usb-function	🔴 #36	Aug 20, 2014 11:25:37 PM		henninger
Functional.bzip2	🟢 #8	Aug 20, 2014 11:17:07 PM		henninger
Functional.cmt	🔴 #2	Aug 20, 2014 11:08:02 PM		henninger
Functional.LTSI-3_10.c_cmt	🔴 #1	Aug 20, 2014 11:06:57 PM		henninger
Functional.LTSI-3_10.ceu	🔴 #11	Aug 20, 2014 10:59:56 PM		henninger
Functional.LTSI-3_10.c_arch_timer	🔴 #4	Aug 20, 2014 10:53:26 PM		henninger
Functional.LTSI-3_10.c_arch_timer	🔴 #3	Aug 20, 2014 10:45:33 PM		henninger
Functional.bzip2	🟢 #7	Aug 20, 2014 10:26:13 PM		minnow
Functional.bzip2	🟢 #6	Aug 20, 2014 10:22:34 PM		minnow

You can participate LTSI

- Follow on Twitter account:

  @LinuxLTSI

LinuxLTSI

@LinuxLTSI

LTSI stands for Long-Term Support Initiative. A group of CE Working Group of the Linux Foundation to provide Long-Term and stable Linux for Industry

- Web:

  http://ltsi.linuxfoundation.org

- Mailing list:

  https://lists.linuxfoundation.org/mailman/listinfo/ltsi-dev

- Git tree :

  http://git.linuxfoundation.org/?p=ltsi-ernel.git;a=summary

THANK YOU

LONG TERM SUPPORT INITIATIVE