FUJITSU

shaping tomorrow with you

Sharing & Reducing the test effort on AGL kernels by using LTSI and LTSI Test Framework

Jun 1st, 2015

Fan Xin, Teppei Asaba
Fujitsu Computer Technologies Limited

LINUX 4.0 anniversary T-shirt

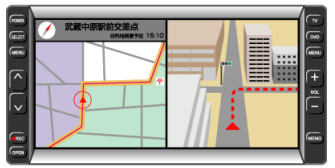
FUJITSU

FUJITSU supports KAWASAKI FRONTALE football club,
FUJITSU supports LINUX 4.0!

whoami

- Embedded Linux development from 2013
- Born in China, live in Japan now

- In-House Embedded Linux Distributor of Fujitsu
- Our Distribution includes LTSI Kernel and is built with Yocto Project
- Our Distribution is used for
  - IVI
  - Server System Controller
  - Storage System
  - Network Equipment
  - Printer
  - etc.



IVI : In-Vehicle Infotainment

Agenda

FUJITSU

- Introduction of LTSI
- Kernel Test Cases for our distro
- Introduction of LTSI Test Project
- Proposal for AGL

FUJITSU

Introduction of LTSI

Release Frequency

FUJITSU

- Mainline Kernel

Kernel Release	Version Date	Days of development
3.11	2013-09-02	64
3.12	2013-11-03	62
3.13	2014-01-19	77
3.14	2014-03-30	70
3.15	2014-06-08	70
3.16	2014-08-03	56
3.17	2014-10-05	63
3.18	2014-12-07	63
3.19	2015-02-09	65
4.0	2015-04-12	62
4.1-rc5	2015-05-25	43

- Over time, kernel development cycles have slowly been getting shorter. The average cycle is about 65 days.

http://www.linuxfoundation.org/publications/linux-foundation/who-writes-linux-2015

Stable Kernel

FUJITSU

- Stable Kernel update ensures the patches will be made to fix the founded problems.

Kernel Release		Updates	Fixes
3.10	longterm	79	4,564
3.11		10	677
3.12	longterm	43	4,885
3.13		11	903
3.14	longterm	43	3,354
3.15		10	703
3.16		7	871
3.17		8	884
3.18	longterm	14	1,444
3.19		8	873

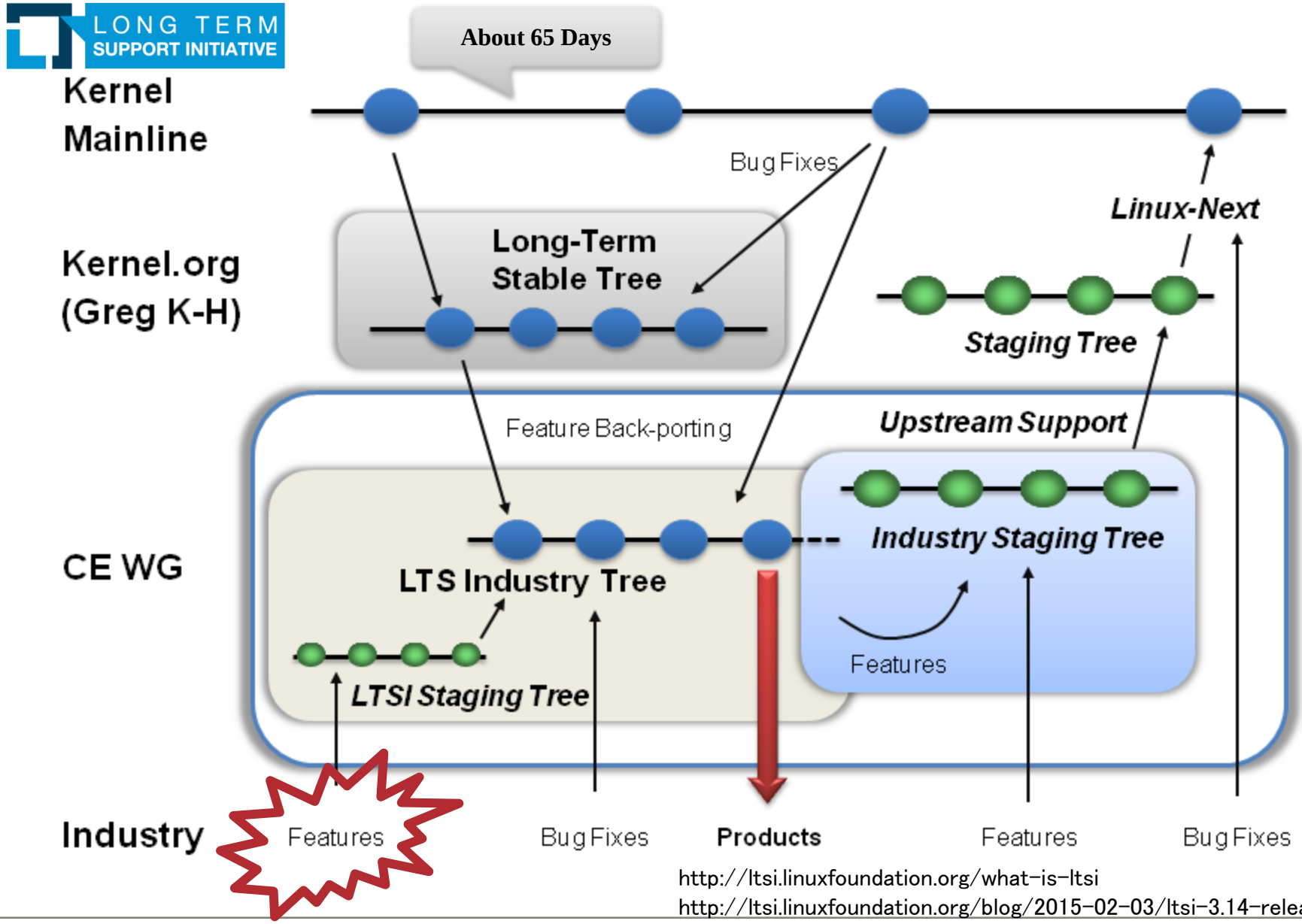
- Longterm Stable (LTS)

There are usually several "longterm maintenance" kernel releases provided for the purposes of backporting bugfixes for older kernel trees. Only important bugfixes are applied to such kernels and they don't usually see very frequent releases, especially for older trees.

http://www.linuxfoundation.org/publications/linux-foundation/who-writes-linux-2015
https://www.kernel.org/category/releases.html

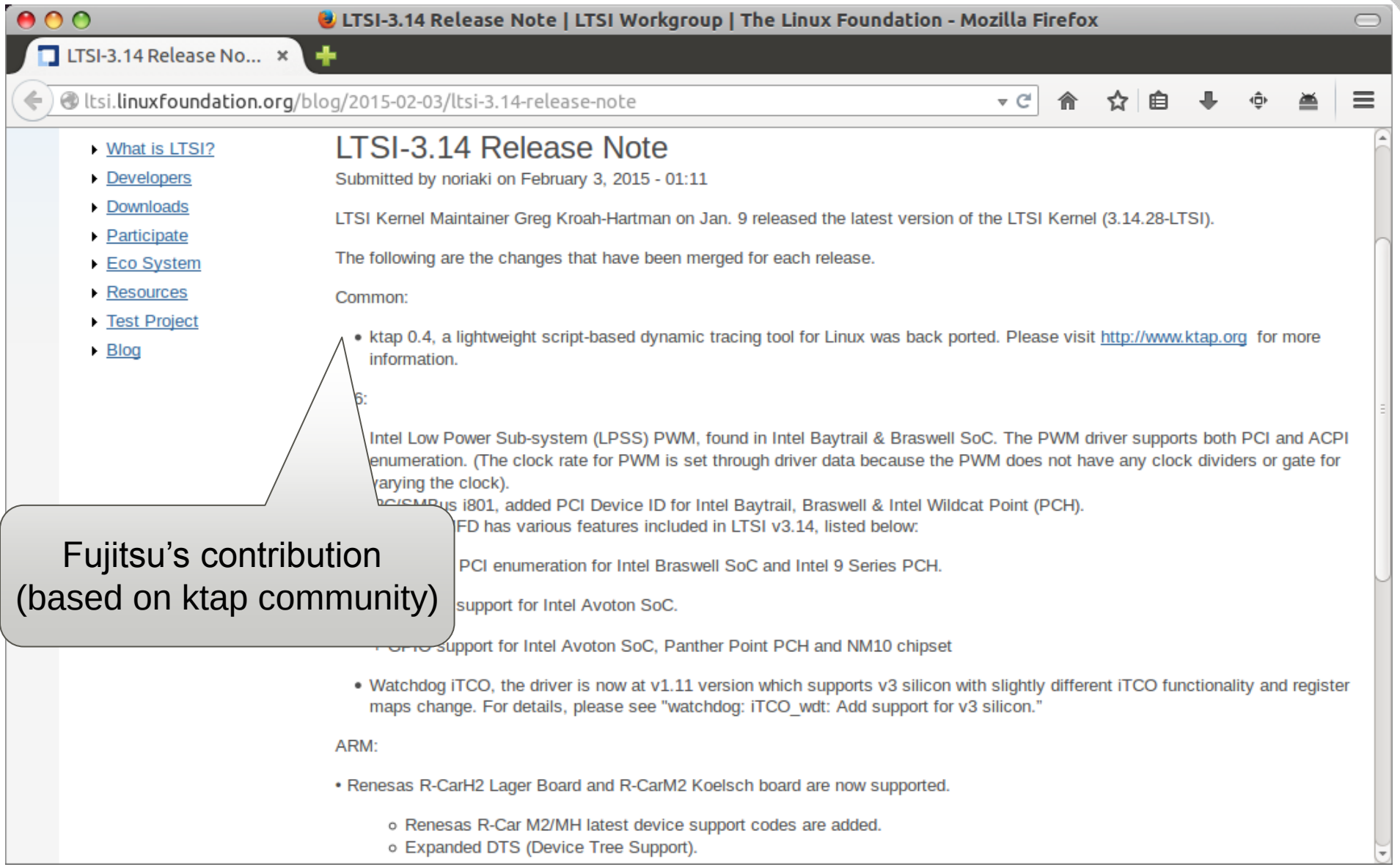
What's LTSI

FUJITSU



http://ltsi.linuxfoundation.org/what-is-ltsi
http://ltsi.linuxfoundation.org/blog/2015-02-03/ltsi-3.14-release-note

New Features



Fujitsu's contribution
(based on ktap community)

http://ltsi.linuxfoundation.org/blog/2015-02-03/ltsi-3.14-release-note

Advantages of LTSI

FUJITSU

- Reduce Cost
- Increase Quality
- Minimize fragmentation

FUJITSU

Kernel Test Cases for our distro

- LTP
- LSB
- IPv6

LTP (Linux Test Project)

FUJITSU

- What's LTP
  http://ltp.sourceforge.net
- Reason to use LTP
  - Regression test at Kernel porting
    - Timer settings...
  - Check for system calls are changed or not when Kernel version is upgraded
    - Return values of error are often changed
  - Check for system calls differences among CPUs (x86, ARM, PowerPC, MIPS...)
    - e.g. "getcontext" is not supported by ARM
  - Check for function differences of systems
    - e.g. "splice" is not supported by NFS (now supported)

LTP (contd.)

FUJITSU

- Case Study 1
  Find a bug of distribution in LTP
  - Float test cases are failed on PowerPC (e500mc core) in Kernel 3.0
  - Optimize option “-O2” is turned off “-fcprop-register” flag in GCC 4.4.6

```
$ powerpc-e500mc-linux-gnu-gcc -I/home/ubinux/e500mc/target/usr/include -O -g -fsigned-char        ¥
-mcpu=e500mc -Wa,-me500mc -mhard-float -g -O2 -fno-strict-aliasing -pipe -Wall                     ¥
-I/home/ubinux/e500mc/target/usr/include                                                           ¥
-I/home/ubinux/e500mc/build/ltp-full-20101031/testcases/kernel/include                             ¥
-I../../../../include -I../../../../include -L/home/ubinux/e500mc/target/lib                       ¥
-L../../../../lib float_exp_log.c -lltp -lpthread -lm -o float_exp_log
```

```
-O2 turns on all optimization flags specified by -O.  It also turns on the following optimization flags:
          -fthread-jumps
          -falign-functions  -falign-jumps
          -falign-loops  -falign-labels
          -fcaller-saves
          -fcrossjumping
          -fcse-follow-jumps  -fcse-skip-blocks
          -fdelete-null-pointer-checks
          -fexpensive-optimizations
          -fgcse  -fgcse-lm
          -findirect-inlining
          -foptimize-sibling-calls
          -fpeephole2
          -fregmove
          -freorder-blocks  -freorder-functions
          -frerun-cse-after-loop
          -fsched-interblock  -fsched-spec
          -fschedule-insns  -fschedule-insns2
          -fstrict-aliasing -fstrict-overflow
          -ftree-switch-conversion
          -ftree-pre
          -ftree-vrp
```

No -fcprop-register!

LTP (contd.)

FUJITSU

- Case Study 2
  - Conduct LTP test cases (float_bessel, float_exp_log, float_power)

Arch	x86	ARM	PowePC
3.10-ltsi	PASS	PASS	PASS
3.14-ltsi	PASS	PASS	**FAIL**

  - Conduct LTP on the same PowerPC platform with different hardware

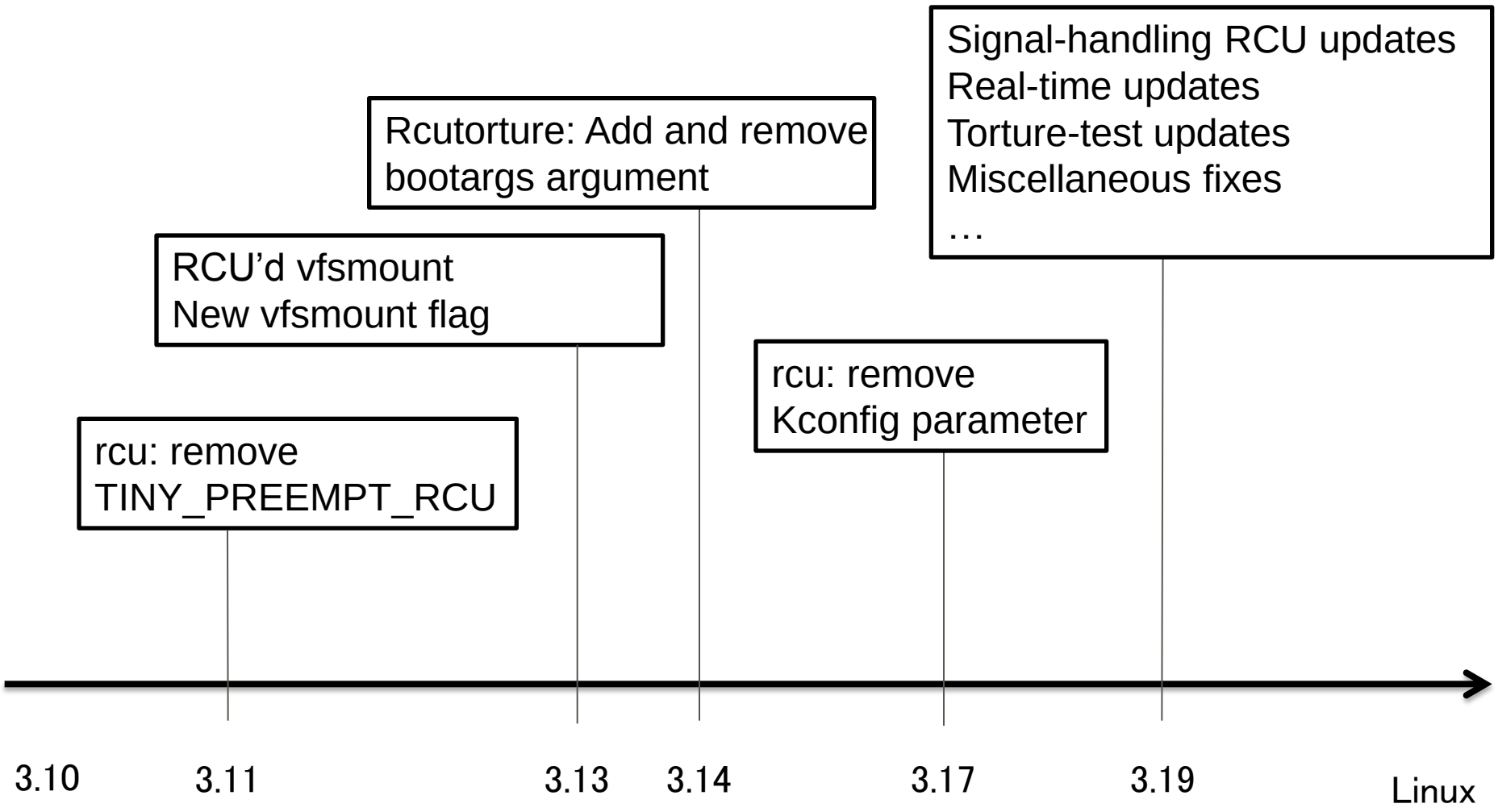
CPU	P2020 (memory 1GB)	P1020 (memory 512MB)
3.14-ltsi	PASS	**FAIL**

- RCU Implementation is changed
  - from 3.10 to 3.11
  - the sys_prctl is heavily used to get/set the floating point exception mode.

LTP (contd.)

FUJITSU

- RCU(read-copy-update) implementation is changing



https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/commit/?id=c30110608cfba7efff3a5e71914aee7c816115c5

LTP (contd.)

- In PowerPC arch, unnecessary memory is allocated and released heavily.
- We find a patch to solve this problem.

```
commit 6d6f3328422a3bc56b0d8dd026a5de845d2abfa7
Author: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
Date:   Tue Jul 22 21:20:01 2014 +0900

    commoncap: don't alloc the credential unless needed in cap_task_prctl

    In function cap_task_prctl(), we would allocate a credential
    unconditionally and then check if we support the requested function.
    If not we would release this credential with abort_creds() by using
    RCU method. But on some archs such as powerpc, the sys_prctl is heavily
    used to get/set the floating point exception mode. So the unnecessary
    allocating/releasing of credential not only introduce runtime overhead
    but also do cause OOM due to the RCU implementation.

    This patch removes abort_creds() from cap_task_prctl() by calling
    prepare_creds() only when we need to modify it.
```

Different platform and hardware will have different test result.
Variety of test environments is important.

https://github.com/torvalds/linux/commit/6d6f3328422a3bc56b0d8dd026a5de845d2abfa7

LTP (contd.)

FUJITSU

- Case Study 3
- TLS (Thread Local Storage) test case on x86 platform, the test result is fail

V3.14	V3.14.28
PASS	FAIL

- Check the change between v3.14 and v3.14.28

```
e990e54 x86/tls: Don't validate lm in set_thread_area() after all
643152b x86/tls: Disallow unusual TLS segments
aeb83c0 x86/tls: Validate TLS entries to protect espfix
```

The implementation of TLS is changed !

- Backport these patches, the test result is pass
- Fixed in 4.0.0-rc6, confirmed fixed in 3.14.37

LSB (Linux Standard Base)

FUJITSU

- What's LSB
  https://wiki.linuxfoundation.org/en/LSB
- LSB is test for distribution compatibility
- Sometimes find a bug of Kernel

LSB (contd.)

FUJITSU

- Case Study 1
  Find a bug of distribution in LSB
  - fpathconf test case is failed in Kernel 2.6.36
  - include/linux/pipe_fs_i.h was changed in 2.6.34..2.6.35
  - Problem of combination with Glibc

```
commit 35f3d14dbbc58447c61e38a162ea10add6b31dc7
Author: Jens Axboe <jens.axboe@oracle.com>
Date:   Thu May 20 10:43:18 2010 +0200

    pipe: add support for shrinking and growing pipes

    This patch adds F_GETPIPE_SZ and F_SETPIPE_SZ fcntl() actions for
    growing and shrinking the size of a pipe and adjusts pipe.c and splice.c
    (and relay and network splice) usage to work with these larger (or smaller)
    pipes.

    Signed-off-by: Jens Axboe <jens.axboe@oracle.com>

diff --git a/include/linux/pipe_fs_i.h b/include/linux/pipe_fs_i.h
index b43a9e0..65f4282 100644
--- a/include/linux/pipe_fs_i.h
+++ b/include/linux/pipe_fs_i.h
@@ -3,7 +3,7 @@

 #define PIPEFS_MAGIC 0x50495045

-#define PIPE_BUFFERS (16)
+#define PIPE_DEF_BUFFERS       16
```

LSB (contd.)

FUJITSU

- Case Study 2
  Find a bug of Kernel in LSB
  - c_lflag test case is failed in Kernel 2.6.31
  - n_tty driver has degraded in 2.6.28..2.6.29
  - Fixed in 2.6.31..2.6.32

```
commit ee5aa7b8b98774f408d20a2f61f97a89ac66c29b
Author: Joe Peterson <joe@skyrush.com>
Date:   Wed Sep 9 15:03:13 2009 -0600

    n_tty: honor opost flag for echoes

    Fixes the following bug:

          http://bugs.linuxbase.org/show_bug.cgi?id=2692

    Causes processing of echoed characters (output from the echo buffer) to
    honor the O_OPOST flag, which is consistent with the old behavior.

    Note that this and the next patch ("n_tty: move echoctl check and
    clean up logic") were verified together by the bug reporters, and
    the test now passes.

    Signed-off-by: Joe Peterson <joe@skyrush.com>
    Cc: Linux Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@suse.de>

diff --git a/drivers/char/n_tty.c b/drivers/char/n_tty.c
```

Sometimes find a bug of Kernel

IPv6 Tests

FUJITSU

- What's IPv6 Ready Logo
  https://www.ipv6ready.org
- IPv6 Ready Logo test suite
  http://www.tahi.org

- Conformance test suite of IPv6 specification
- Test for IPv6 protocol stack (only Kernel)

Case Study: IPv6 test for 3.10.31-ltsi

FUJITSU

- Test result

v3.10.31−ltsi

Section	RFC	Summary	Total	Pass	Fail	N/A
Section 1	RFC 2460	IPv6 Specification	54	53	0	1(*)
Section 2	RFC 4861	Neighbor Discovery for IPv6	236	214	**14**	**8**
Section 3	RFC 4862	IPv6 Stateless Address Autoconfiguration	45	44	**1**	0

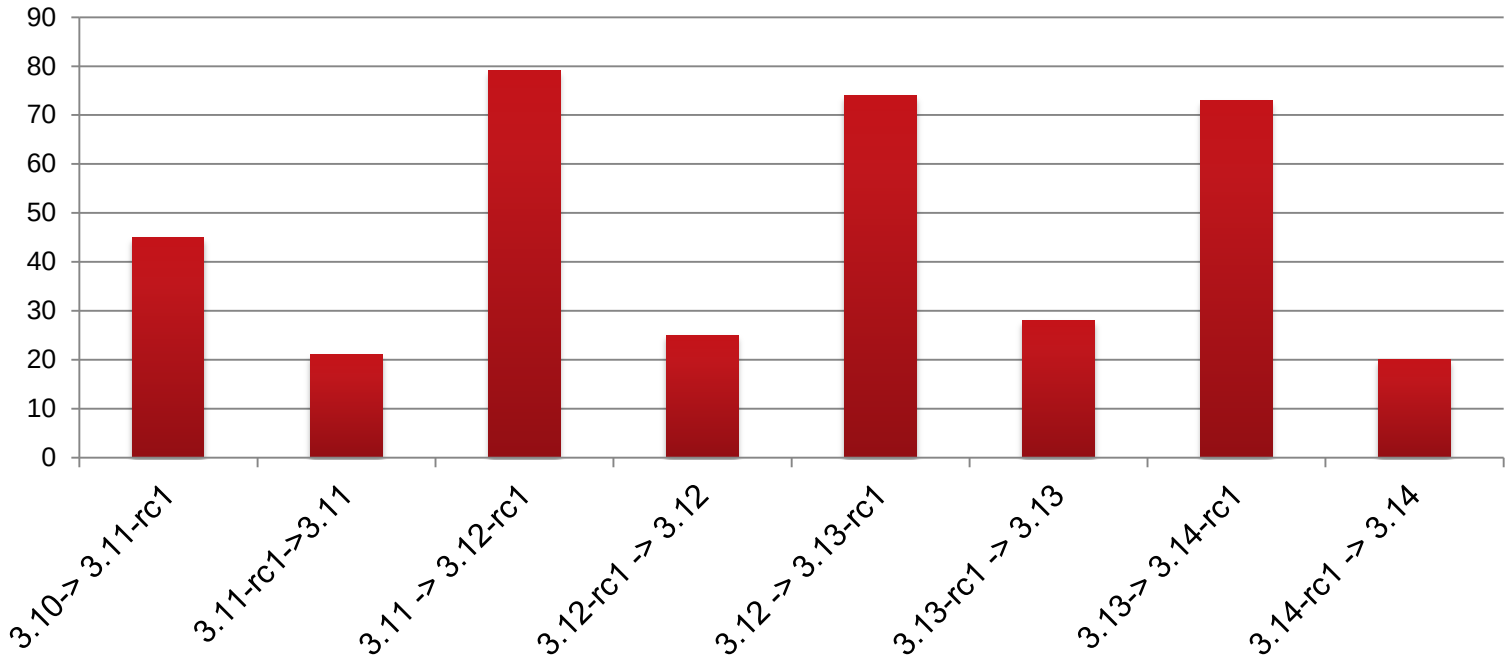
- Analysis of section 2 (Neighbor Discovery for IPv6)
  - Recieved Redirect ICMP Message with no Redirected Header option
  - EchoReply has not been forwarded to the next hop that is specified

FUJITSU

- IPv6 stack has a lot of changes

```
% git log --oneline v3.10..v3.14 -- net/ipv6 | wc -l
364
```

- Number of Changes (-- net/ipv6)



FUJITSU

Result of Section 2 Test(Neighbor Discovery for IPv6)

21 Changes

No.	v3.10	v3.11-rc1	v3.11
185	FAIL	FAIL	PASS
187	FAIL	FAIL	PASS
189	FAIL	FAIL	PASS
190	FAIL	FAIL	PASS
200	FAIL	FAIL	PASS
201	FAIL	FAIL	PASS
202	FAIL	FAIL	PASS
203	FAIL	FAIL	PASS
204	FAIL	FAIL	PASS
206	FAIL	FAIL	PASS
210	FAIL	FAIL	PASS
215	FAIL	FAIL	PASS
220	FAIL	FAIL	PASS
225	FAIL	FAIL	PASS

FUJITSU

- Found a patch

```
commit c92a59eca86f5d13ae4d481c3bae6b54609fe006
Author: Duan Jiong <duanj.fnst@cn.fujitsu.com>
Date:   Thu Aug 22 12:07:35 2013 +0800

    ipv6: handle Redirect ICMP Message with no Redirected Header option

    rfc 4861 says the Redirected Header option is optional, so
    the kernel should not drop the Redirect Message that has no
    Redirected Header option. In this patch, the function
    ip6_redirect_no_header() is introduced to deal with that
    condition.

    Signed-off-by: Duan Jiong <duanj.fnst@cn.fujitsu.com>
    Acked-by: Hannes Frederic Sowa hannes@stressinduktion.org
```

FUJITSU

Backport to v3.10.57 and Retry the Section 2

1 change

No.	v3.10.57	v3.10.57 + Patch
185	FAIL	PASS
187	FAIL	PASS
189	FAIL	PASS
190	FAIL	PASS
200	FAIL	PASS
201	FAIL	PASS
202	FAIL	PASS
203	FAIL	PASS
204	FAIL	PASS
206	FAIL	PASS
210	FAIL	PASS
215	FAIL	PASS
220	FAIL	PASS
225	FAIL	PASS

Common tests for share

FUJITSU

- LTP
  - Should use LTP, and decide GCC, Binutils, Glibc versions
  - Should decide a release criteria
    - e.g. All of Syscalls test cases are passed on x86 and ARM boards
  - HTML formatted results should be opened to the public
- LSB
  - Should use LSB test suites with Yocto Project
  - Should decide a release criteria
    - e.g. All of Core test cases are passed
  - HTML formatted results should be opened to the public
- IPv6
  - Should use IPv6 tests
  - Should decide a release criteria
    - e.g. All of Core and IPsec test cases are passed
  - HTML formatted results should be opened to the public

FUJITSU

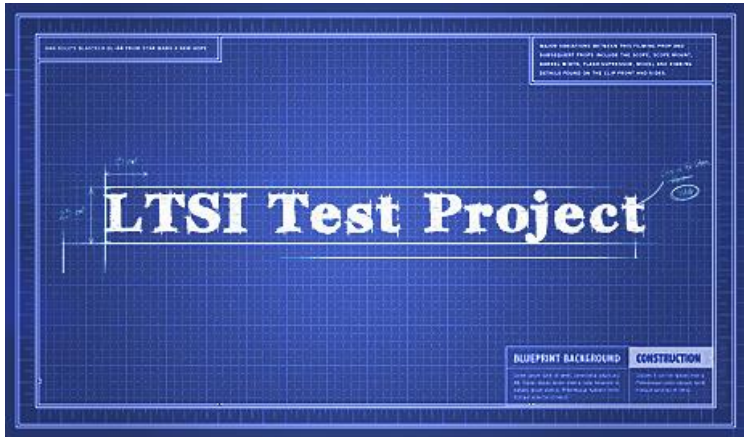
Introduction of LTSI Test Project

LTSI Test Project

FUJITSU

- A project to create a “common test platform” which anyone can use and reduce a lot of duplicated effort.
- Gather test cases from companies to try to find the common test elements among the companies.
- Based on the gathered test cases, create “LTSI Common Test Platform”
- http://ltsi.linuxfoundation.org/ltsi-test-project

- Gaps with my expectation
  - The existing test cases is not enough.
  - Test results also have not been shared



Practice with LTSI Test Project

FUJITSU

- Installation
  - Download from bitbucket

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

  - Install JTA and configure the Environment variables

About 10 minutes

- Configuration
  - Install the cross compiler
  - Add the board file
  - Add the test plan file

About 2 hours

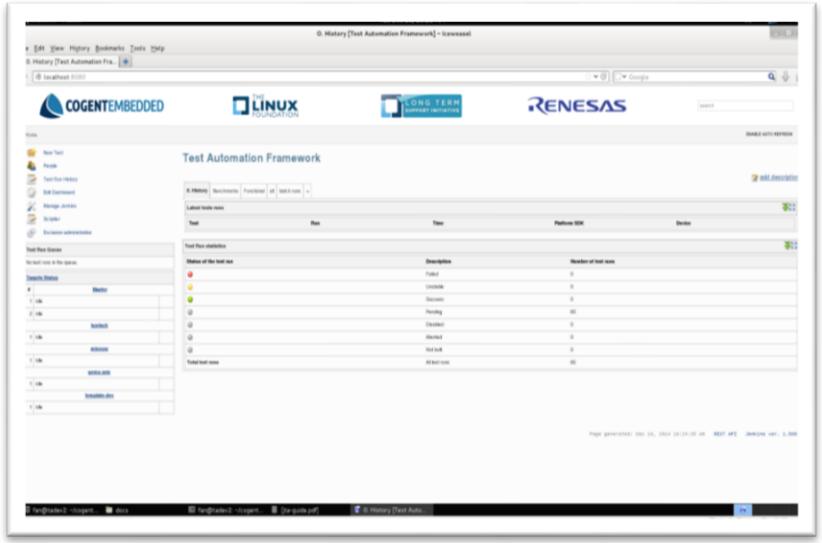
- Run the test

Host Environment

OS	Debian Wheezy
Architecture	AMD 64

Browser Environment

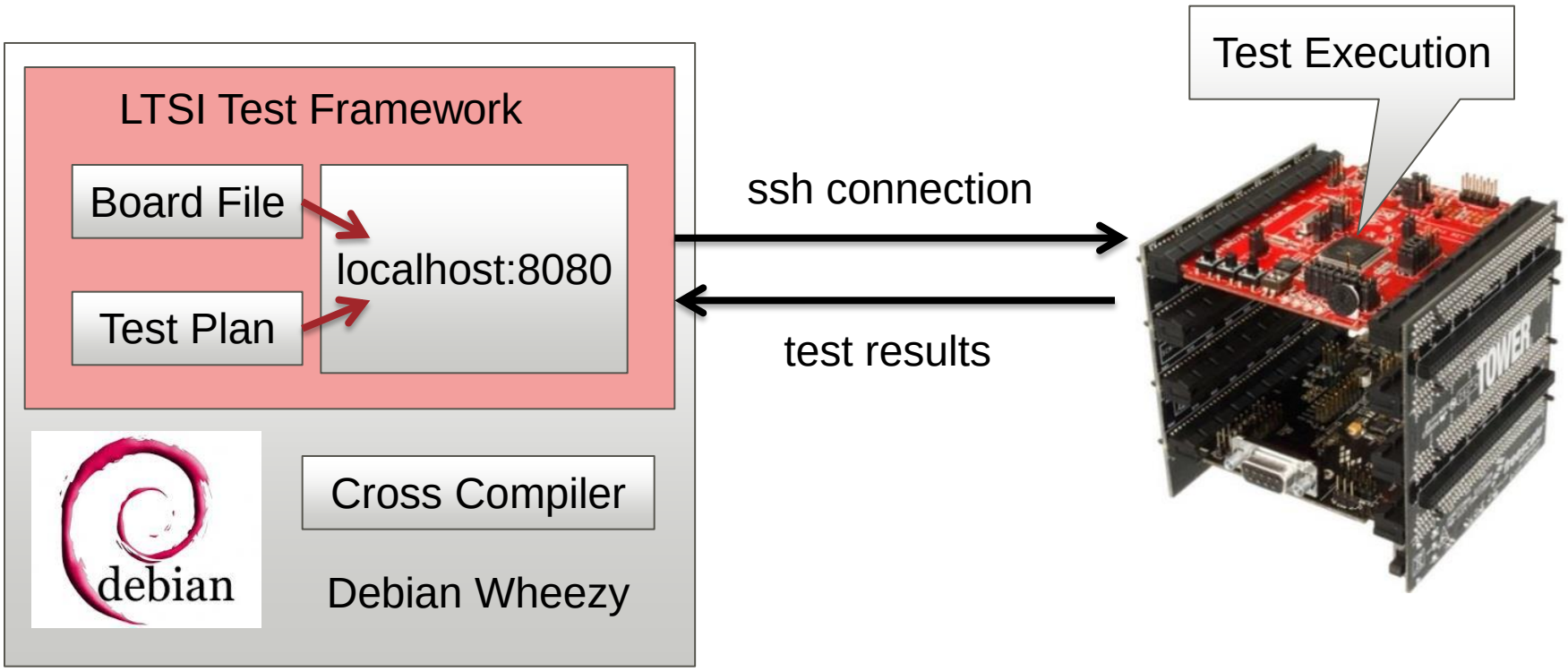
Web browser	Javascript and CSS support



JTA: Jenkins-based Test Automation

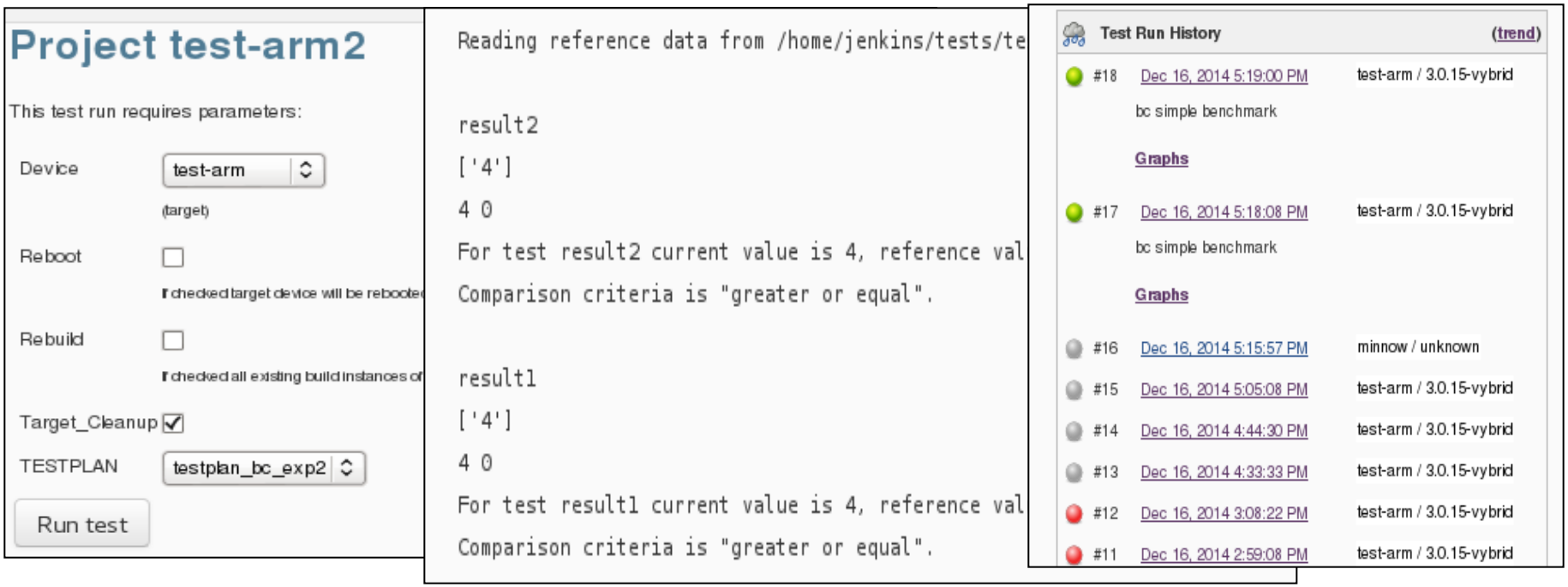
Evaluation Environment

- Target Device : Freescale Vybrid VF-6xx (Cortex-A5) Tower System



How to run test

- Conduct the bc computing test case
- Confirm the test environment easily and make a quick start



- Takes 4 hours from install to finish the first simple test
- Easy to use, prefer a video demonstration, more platforms

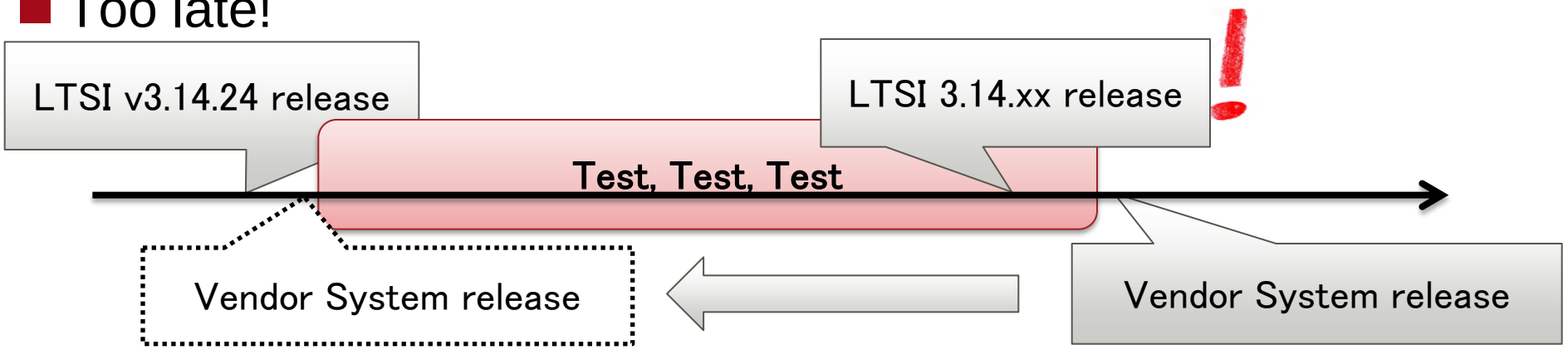
Release Time Gap

- The LTSI is released frequently

tags

4 months ago	**v3.14.28-ltsi**	This is the 3.14.28-ltsi release	tag | commit | shortlog | log
5 months ago	**v3.14.24-ltsi-rc1**	This is the 3.14.24-rc1-ltsi release	tag | commit | shortlog | log
12 months ago	**v3.4.91-ltsi**	This is the 3.4.91-ltsi release	tag | commit | shortlog | log
14 months ago	**v3.10.31-ltsi**	This is the v3.10.31-ltsi release	tag | commit | shortlog | log
14 months ago	**v3.4.81-ltsi**	This is the v3.4.81-ltsi release	tag | commit | shortlog | log
14 months ago	**v3.0.101-ltsi**	This is the v3.0.101-ltsi release	tag | commit | shortlog | log
15 months ago	**v3.10.28-ltsi-rc1**	This is the 3.10.28-ltsi-rc1 release	tag | commit | shortlog | log
2 years ago	**v3.4.46-ltsi**	This is the 3.4.46-ltsi release	tag | commit | shortlog | log
2 years ago	**v3.0.79-ltsi**	This is the 3.0.79-ltsi release	tag | commit | shortlog | log
2 years ago	**v3.4.25-ltsi**	This is the v3.4.25-ltsi release	tag | commit | shortlog | log
2 years ago	**v3.0.38-ltsi**	LTSI 3.0.38 release	tag | commit | shortlog | log

- After LTSI releases, the vendor has to conduct a series of tests to validate and then release their systems.
- This procedure takes few months or more.
- Too late!



Our contribution to LTSI Test Project

FUJITSU

- As the first step, we contribute our ethtool test set to LTSI Test Project
- ethtool
  - A utility for Linux kernel-based operating system for displaying and modifying some parameters of network interface controllers (NICs) and their device drivers.
  - https://www.kernel.org/pub/software/network/ethtool/
- Advantages
  - Common test element
  - Easy to check each hardware support features
  - Useful for not only person but also the vendors which provide systems for multiple platforms

[LTSI-dev] Ethtool test suite for LTSI Test Project

Bian, Naimeng biannm at cn.fujitsu.com
Mon Nov 3 01:07:51 UTC 2014

- Next message: [LTSI-dev] To enquire LTSI submission requirement
- **Messages sorted by:** [date] [thread] [subject] [author]

```
Hi all,

As I know, the LTSI Test Project is collecting test cases.
I provide our test set of ethtool.

We have implemented test cases of 50/67 ETHTOOL COMMANDs.
You can obtain details from README in attached archive.

Let's test together!

Questions or comments?

Regards
 Bian
-------------- next part --------------
A non-text attachment was scrubbed...
Name: testset-ethtool-0.1.tar.gz
Type: application/x-gzip
Size: 18989 bytes
Desc: testset-ethtool-0.1.tar.gz
URL: <http://lists.linuxfoundation.org/pipermail/ltsi-dev/attachments/20141103/0a0ffcae/attachment.gz>
```

- Next message: [LTSI-dev] To enquire LTSI submission requirement
- **Messages sorted by:** [date] [thread] [subject] [author]

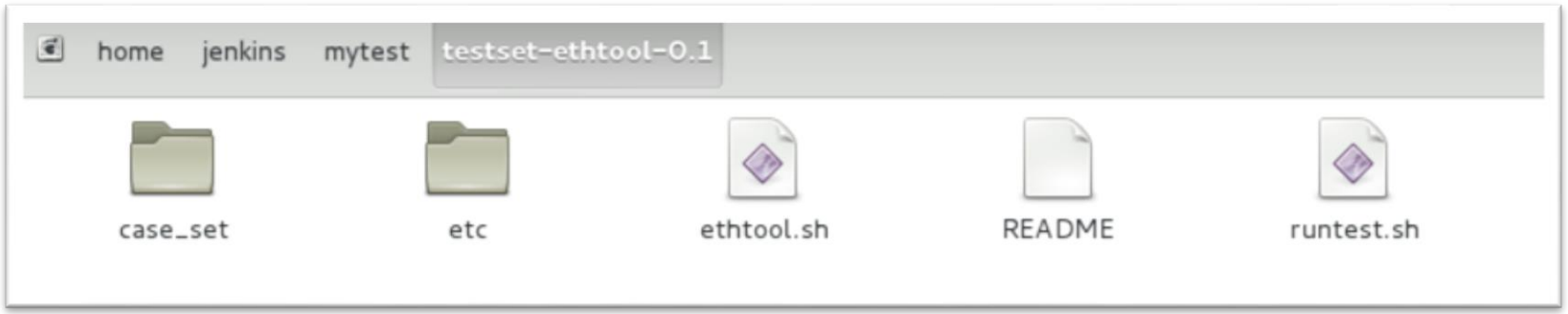
More information about the LTSI-dev mailing list

http://lists.linuxfoundation.org/pipermail/ltsi-dev/2014November/005461.html

Quick Start of ethtool test set

FUJITSU

ethtool test set



Create New Test

↓

Change script name to ethtool.sh

↓

Make Directory : /home/jenkins/tests/test-ethtool

↓

Decompress testset-ethtool-0.1 to
/home/jenkins/tests/test-ethtool

↓

Start Test

Quick Start of ethtool test set (contd.)

FUJITSU

- We can use this test case to check whether the ethtool in kernel is support commands or not.
- There are too many commands and we pickup some of them with high frequency to test.

```
jenkins@tadev2:/home/jenkins/mytest/testset-ethtool-0.1/case_set$ ls
ETHTOOL_FLASHDEV       ETHTOOL_GFLAGS          ETHTOOL_GREGS          ETHTOOL_GSG          ETHTOOL_RESET       ETHTOOL_SMSGLVL       ETHTOOL_SSET
ETHTOOL_GCHANNELS      ETHTOOL_GGRO            ETHTOOL_GRINGPARAM     ETHTOOL_GSSET_INFO   ETHTOOL_SCHANNELS   ETHTOOL_SPAUSEPARAM   ETHTOOL_SSG
ETHTOOL_GCOALESCE      ETHTOOL_GGSO            ETHTOOL_GRXCLSRLALL    ETHTOOL_GSTATS       ETHTOOL_SCOALESCE   ETHTOOL_SPFLAGS       ETHTOOL_STSO
ETHTOOL_GDRVINFO       ETHTOOL_GLINK           ETHTOOL_GRXCLSRLCNT    ETHTOOL_GSTRINGS     ETHTOOL_SEEE        ETHTOOL_SRINGPARAM    ETHTOOL_STXCSUM
ETHTOOL_GEEE           ETHTOOL_GMODULEEEPROM   ETHTOOL_GRXCLSRULE     ETHTOOL_GTSO         ETHTOOL_SEEPROM     ETHTOOL_SRXCLSRLDEL   ETHTOOL_SUFO
ETHTOOL_GEEPROM        ETHTOOL_GMODULEINFO     ETHTOOL_GRXCSUM        ETHTOOL_GTXCSUM      ETHTOOL_SET_DUMP    ETHTOOL_SRXCLSRLINS   ETHTOOL_SWOL
ETHTOOL_GET_DUMP_DATA  ETHTOOL_GMSGLVL         ETHTOOL_GRXFH          ETHTOOL_GUFO         ETHTOOL_SFEATURES   ETHTOOL_SRXCSUM       ETHTOOL_TEST
ETHTOOL_GET_DUMP_FLAG  ETHTOOL_GPAUSEPARAM     ETHTOOL_GRXFHINDIR     ETHTOOL_GWOL         ETHTOOL_SFLAGS      ETHTOOL_SRXFH
ETHTOOL_GET_TS_INFO    ETHTOOL_GPERMADDR       ETHTOOL_GRXRINGS       ETHTOOL_NWAY_RST     ETHTOOL_SGRO        ETHTOOL_SRXFHINDIR
ETHTOOL_GFEATURES      ETHTOOL_GPFLAGS         ETHTOOL_GSET           ETHTOOL_PHYS_ID      ETHTOOL_SGSO        ETHTOOL_SRXNTUPLE
```

- We finished 50/67 feature support tests in provided test set.

```
...
|62  |ETHTOOL_SSG     |Set Scatter-gather                    |PASS
|63  |ETHTOOL_STSO    |Set TCP Segmentation Offload          |PASS
|64  |ETHTOOL_STXCSUM |Set RX Checksumming                   |PASS
|65  |ETHTOOL_SUFO    |Set TCP Fragmentation Offload         |FAIL
|66  |ETHTOOL_SWOL    |Set Wake-on-lan Mode                  |PASS
...
```

Part of test log

notice: this test set needs serial connection

Test result

FUJITSU

- We conduct the ethtool test set on the following platform to test our distribution.

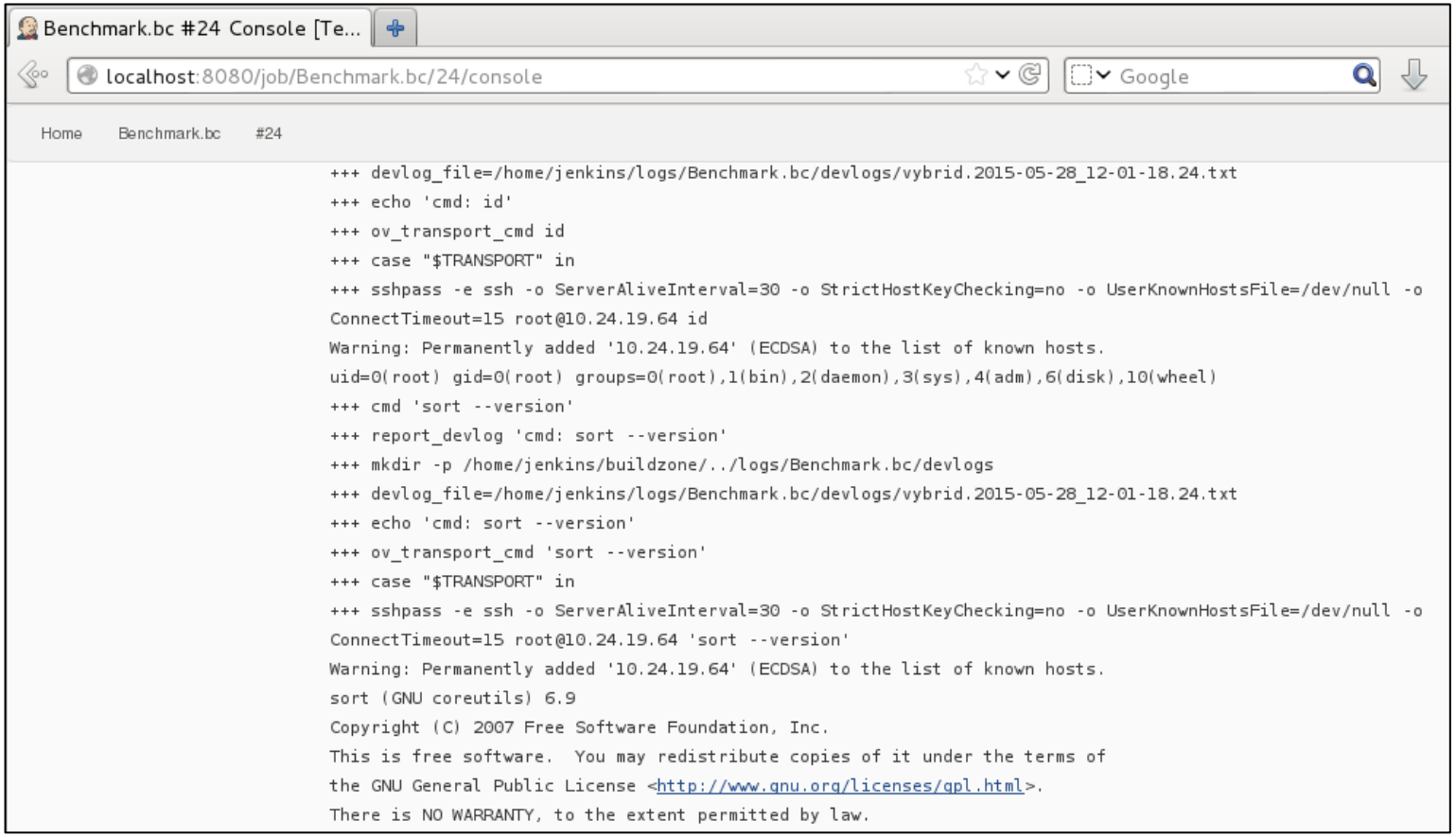
Ethtool commnad	MinnowBoard MAX (x86-64)	Helio (ARM)	P1020RDB (PowerPC)
ETHTOOL_GEEE	UNSUPPORTED	PASS	UNSUPPORTED
ETHTOOL_GPERMADDR	PASS	FAIL	PASS
ETHTOOL_GRXCLSRULE	UNSUPPORTED	UNSUPPORTED	FAIL
ETHTOOL_GRXCSUM	PASS	PASS	PASS
ETHTOOL_SPAUSEPARAM	PASS	PASS	FAIL
ETHTOOL_GRXFHINDIR	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED

- Easy understand result about the features on different hardware.
- Next Step
  - Need analysis for the result data
  - Public the test result

LTSI Test Framework for userland

FUJITSU

- Not only kernel, LTSI Test Framework also could be used for userland test.



Effort

FUJITSU

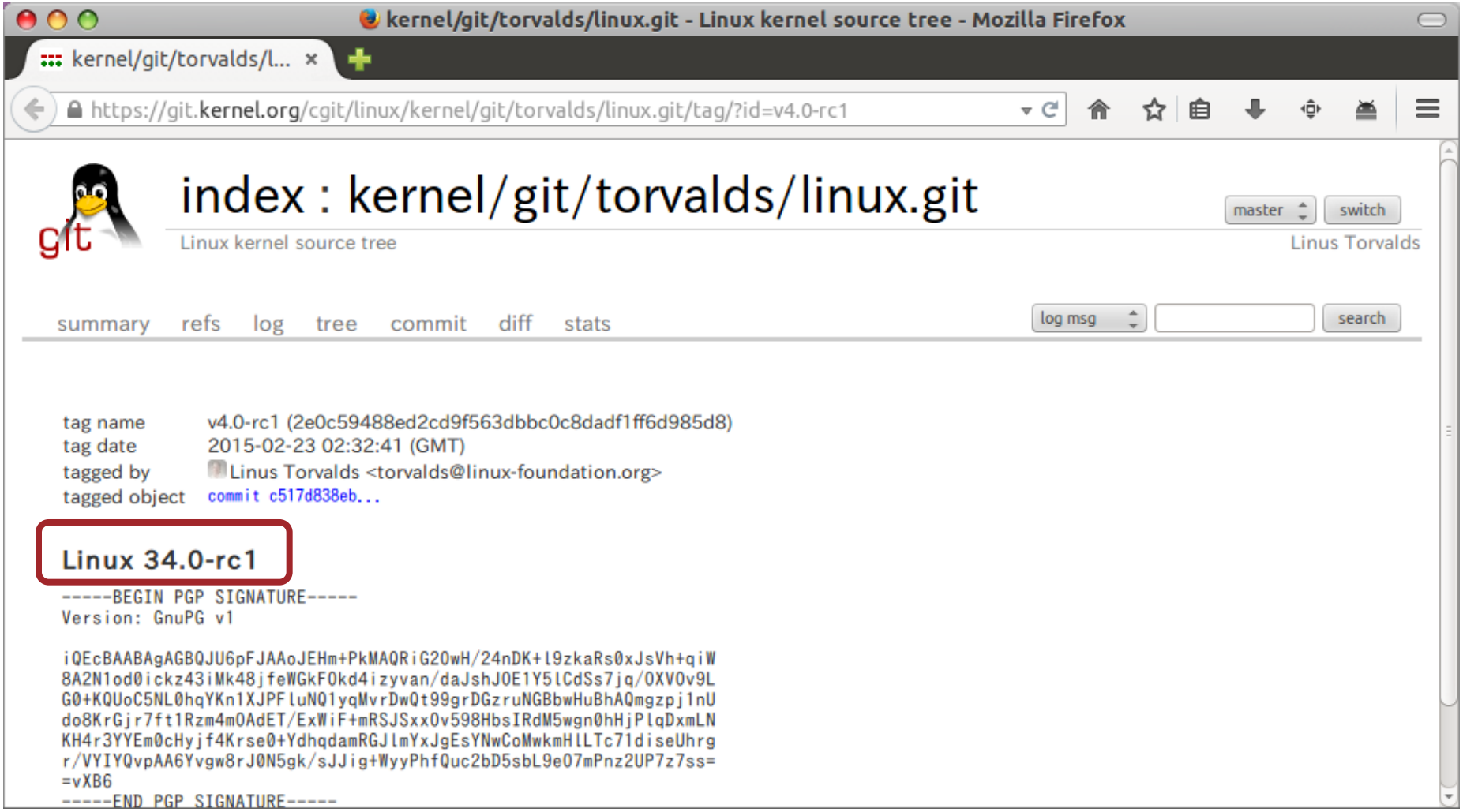
- We contribute our test set to community.
- Furthermore, by using our test set, more hardware platforms could be validated and evaluated.
- Finally, it is convenient for the customer to select and evaluate various kinds of hardware platforms.

FUJITSU

Proposal for AGL

Who makes this mistake?

FUJITSU



Testing is necessary because we all make mistakes.

Proposal and Summary

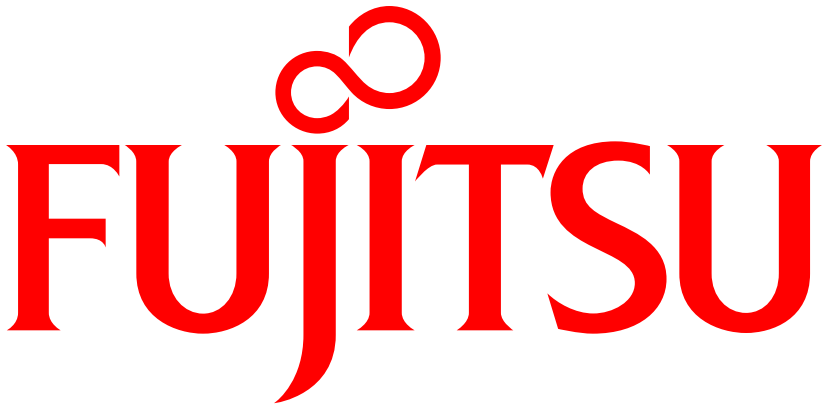
FUJITSU

- We practice the LTSI Test Project and contribute our test set to the community. Fujitsu will continue to contribute to LTSI Test Framework.

- Future Work
  - Contribute more test sets to community
  - Validate the LTSI Test Project on more platforms

- Let's contribute test cases together!

Sharing & Reducing the test effort on AGL kernels by using LTSI and LTSI Test Framework



shaping tomorrow with you