



Upstream Linux Development

Renesas Mobile Corporation
Renesas Mobile Europe Ltd.
Renesas Solutions Corp.

Rev. 1.00

RENESAS Group CONFIDENTIAL

00000-A

Outline

- Statistics of Linux development
- Renesas connection to the upstream
- LTSI : Long Term kernel support initiative
- Resolution of past Android isolation issue
- Runtime PM (Power Management) update
- ARM Linux updates
- Conclusion

[prologue] Statistics of Linux development

Renesas has been major contributor to the Linux and other opensource (=OSS) upstream development. For this reason we believe we can provide the best solution to utilize latest opensource resources to make competitive consumer product with the most advanced SoC.

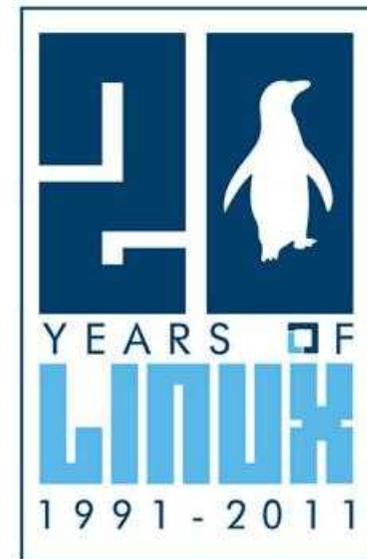
Linux is good, but so different from RTOS

RTOS	Linux
deterministic	heuristics
static	dynamic
fixed resource allocation	flexible resource allocation
predictable behavior	runtime coordination
real address space	virtual address space
single point development	distributed development

So it worth spending sometime for understanding "Linux" first.

We celebrated 20th year anniversary in 2011

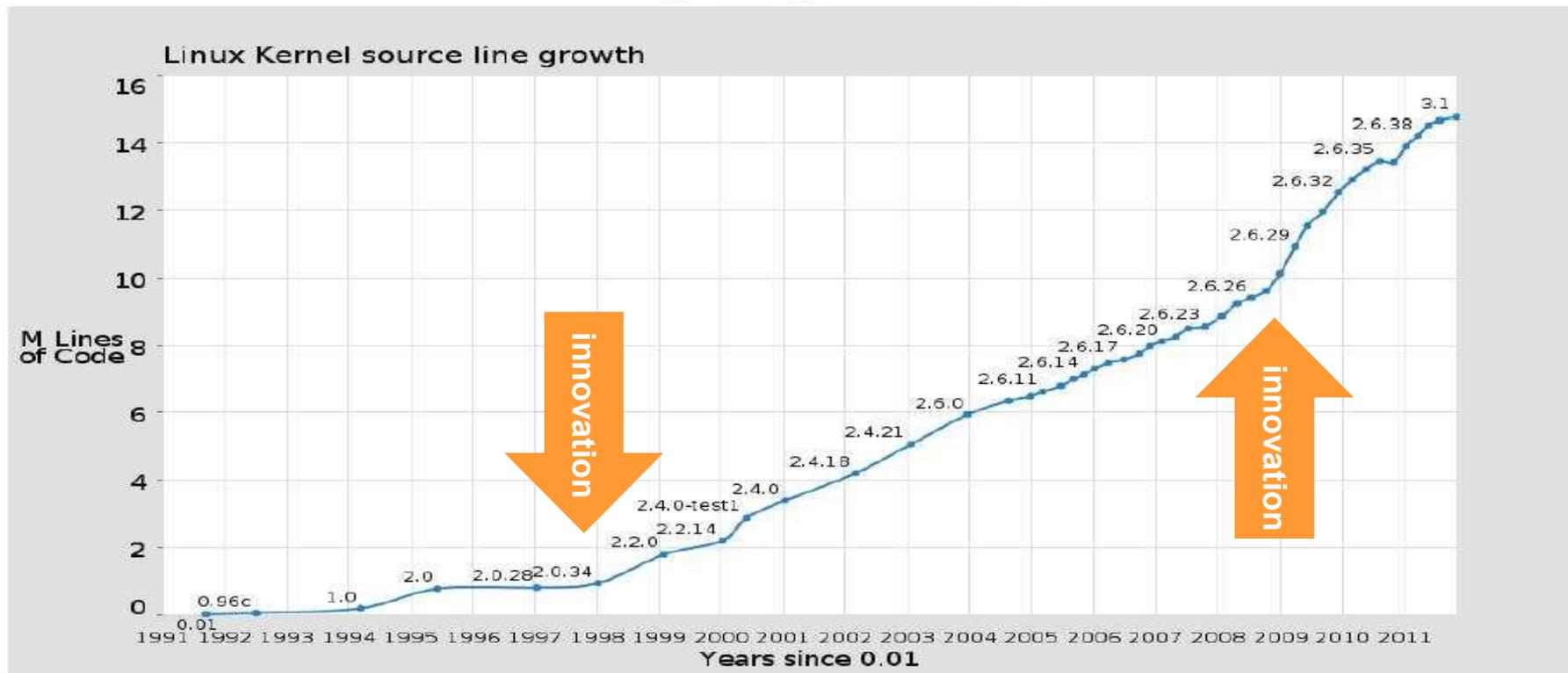
- Development statistics
 - code size
 - release interval
 - developer size
- Development process
- Code Management system
- kernel flexibility



Kernel code line number trends [long term]

This year is 20th anniversary of Linux

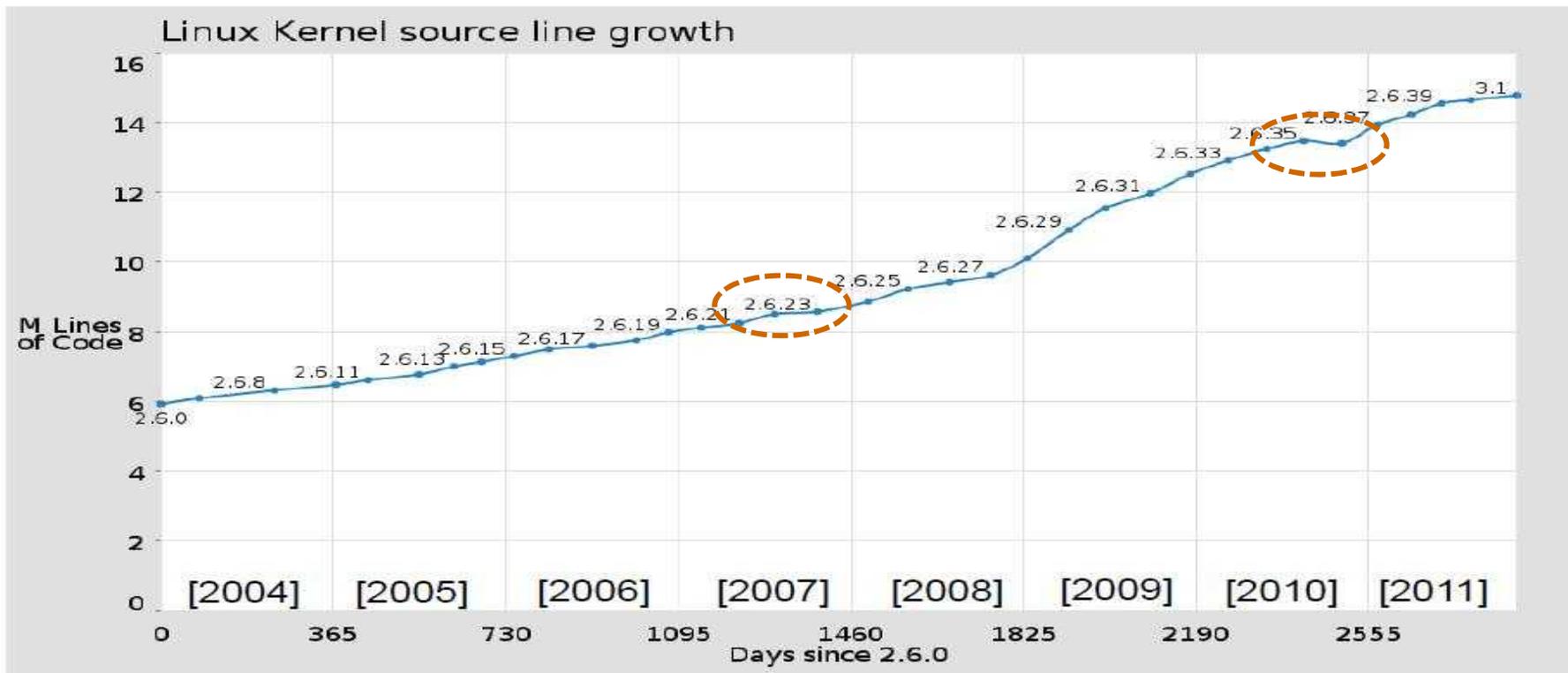
- Linux is growing with rapid pace and further acceleration is on going from 2009



Kernel code line number trends [recent]

Source code growth

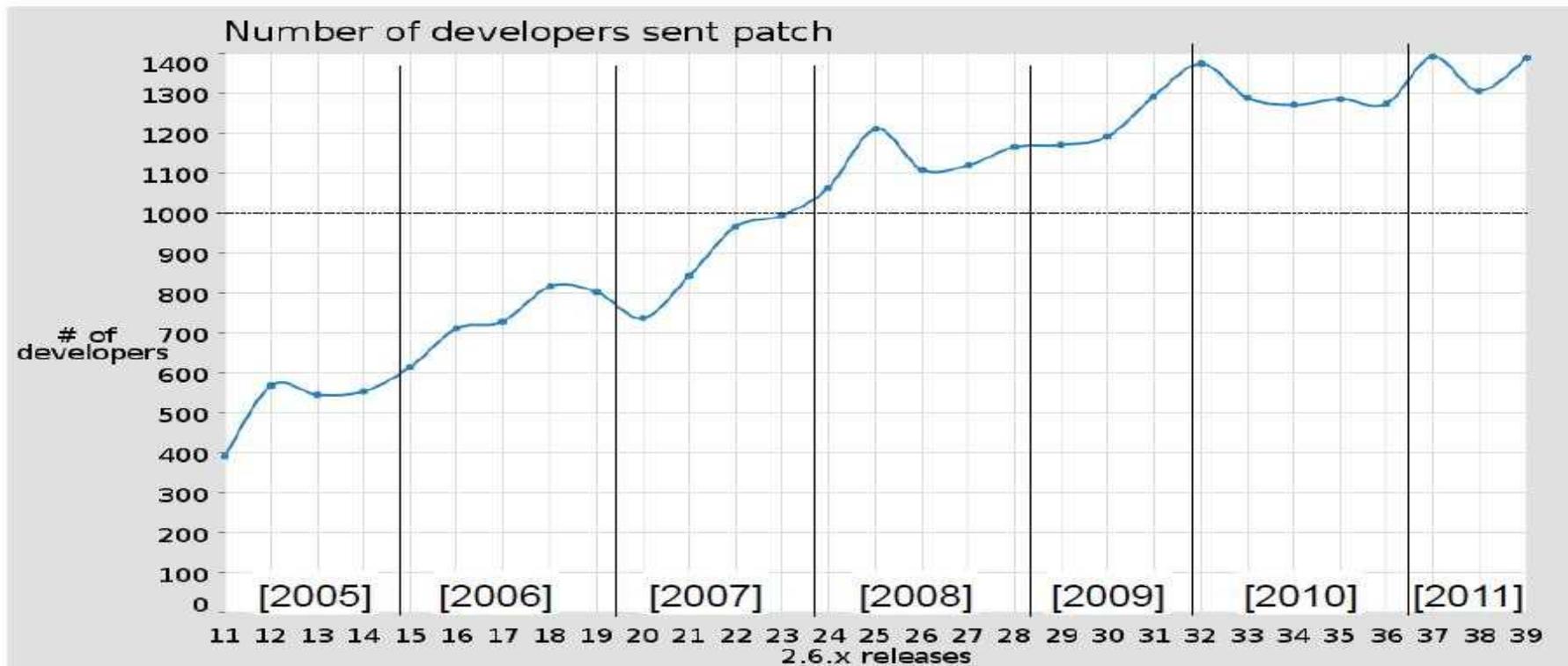
- 0.8ML/year from 2004-2008, 1.8ML/year since 2009,
- Large code removal happened 2.6.23 and 2.6.36



Kernel developer (patch submitter) trends

Number of developers sent patches

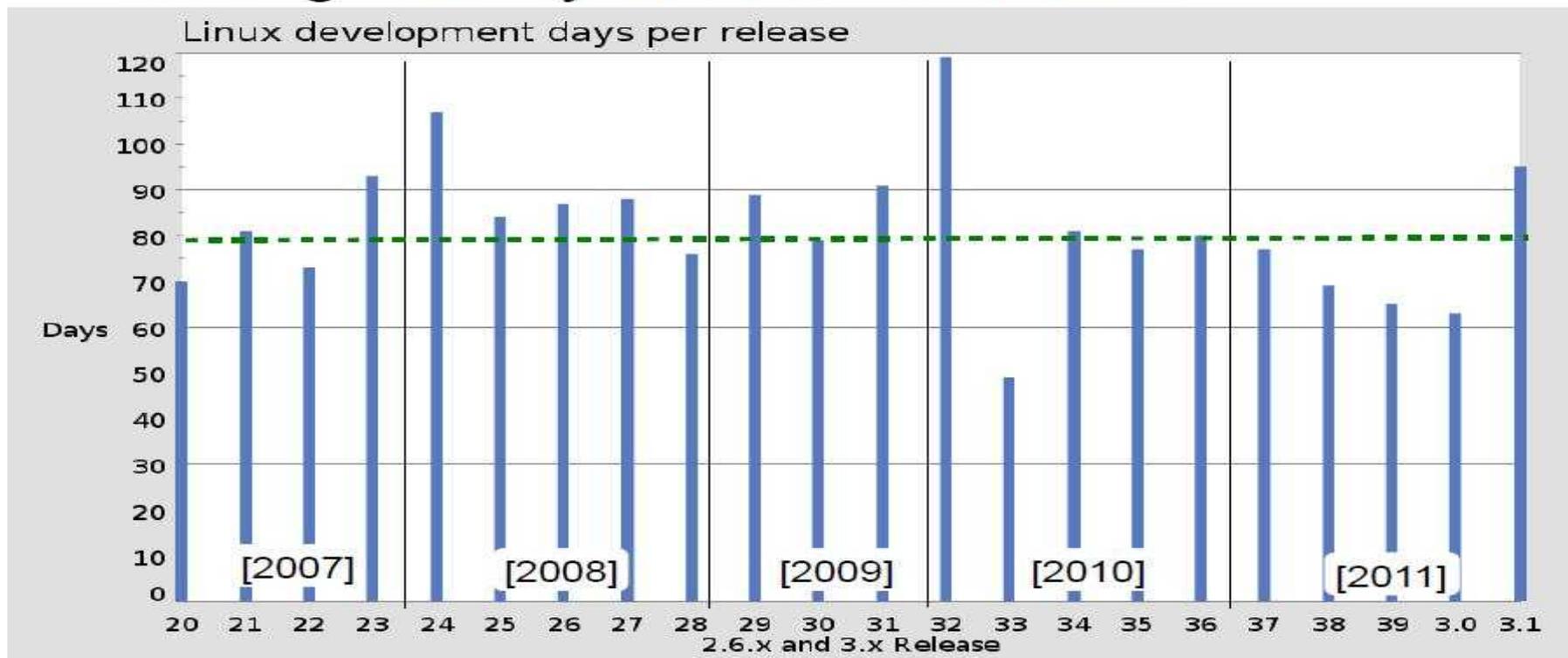
- More than 1000 of developers since 2.6.23
- Become average 1,300 developers this year



Kernel release interval record history

Days to develop each releases

- Average 82 days in last 4 releases
- Average 84 days in last 10 releases



“www.kernel.org” is Linux master repository

The Linux Kernel Archives

Welcome to the Linux Kernel Archives. This is the primary site for the Linux kernel source, but it has much more than just Linux kernels.

[Frequently Asked Questions](#)

Protocol	Location
HTTP	http://www.kernel.org/pub/
FTP	ftp://ftp.kernel.org/pub/
RSYNC	rsync://rsync.kernel.org/pub/

Latest Stable Kernel: ②



[3.1.8](#)

development target code ①

linux-next: next-20120106	2012-01-06							[Gitweb]
mainline: 3.2	2012-01-04	[Full Source]	[Patch]	[View Patch]				[Gitweb]
stable: 3.1.8	2012-01-06	[Full Source]	[Patch]	[View Patch]	[View Inc.]	[Gitweb]	[Changelog]	
stable: 3.0.9	2011-11-11	[Full Source]	[Patch]	[View Patch]	[View Inc.]	[Gitweb]	[Changelog]	
stable: 2.6.39.4	2011-08-03	[Full Source]				[Gitweb]		
stable: 2.6.35.14	2011-08-01					[Gitweb]		
stable: 2.6.33.20	2011-11-07					[Gitweb]		
stable: 2.6.32.53	2012-01-06					[Gitweb]		

long-term stable code ③

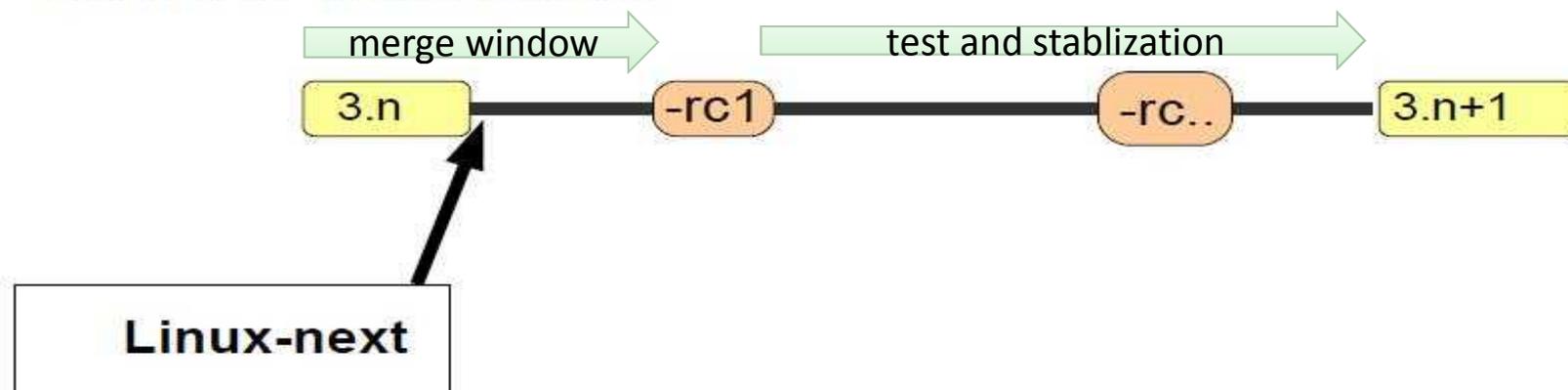
Changelogs are provided by the kernel authors directly. Please don't write the webmaster about them.

[Customize the patch viewer](#)

Continuous development process

Linux-next for next merge window

- Do automated build test to solve problem for next merge window
- Merging 190 of subsystem trees
- Maintainers get notification about build problems on almost daily basis
- Started from 2008



multipoint & concurrent development with "git"

gitk: linux-2.6

ファイル 編集 ビュー

Revert "mmc: enable runtime PM by default"
mmc: sdhci: remove "state" argument from sdhci_suspend_host
SELinux: Fix RCU deref check warning in sel_netport_insert()
Merge branch 'evm-fixes' of git://git.kernel.org/pub/scm/linux/kernel/git/kasatkin/linux-digsig into for-linus
evm: prevent racing during tfm allocation
evm: key must be set once during initialization
Merge branch 'for-3.2-fixes' of git://git.kernel.org/pub/scm/linux/kernel/git/tj/cgroup
cgroups: fix a css_set not found bug in cgroup_attach_proc
Merge branch 'x86-urgent-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/tip/tip
x86, dumpstack: Fix code bytes breakage due to missing KERN_CONT
Merge branch 'timers-urgent-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/tip/tip
time/clocksource: Fix kernel-doc warnings
rtc: m41t80: Workaround broken alarm functionality
rtc: Expire alarms after the time is set.
Merge branch 'perf-urgent-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/tip/tip
oprofile: Fix uninitialized memory access when writing to writing to oprofilefs
Merge branch 'stable/for-linus-fixes-3.2' of git://git.kernel.org/pub/scm/linux/kernel/git/konrad/xen
Revert "xen/pv-on-hvm kexec: add xs_reset_watches to shutdown watches from old kernel"
Merge branch 'sh-fixes-for-linus' of git://github.com/pmundt/linux-sh
sh: fix build warning in board-sh7757lcr
Merge branch 'rmobile-fixes-for-linus' of git://github.com/pmundt/linux-sh
Merge branches 'rmobile/core', 'rmobile/kota2' and 'rmobile/ag5' into rmobile-fixes-for-linus
ARM: mach-shmobile: AG5EVM GIC Sparse IRQ fix
ARM: mach-shmobile: Kota2 TPU LED platform data
ARM: mach-shmobile: Kota2 GIC Sparse IRQ fix
ARM: mach-shmobile: Kota2 PINT fix
ARM: mach-shmobile: SH73A0 external Ethernet fix
Merge branch 'bugfixes' of git://git.linux-nfs.org/projects/trondmy/linux-nfs
NFSv4: Do not accept delegated opens when a delegation recall is in effect
NFSv4: Ensure correct locking when accessing the 'lock_states' list
NFSv4.1: Ensure that we handle _all_ SEQUENCE status bits.
NFSv4: Don't error if we handled it in nfs4_recovery_handle_error
SUNRPC: Ensure we always bump the backlog queue in xprt_free_slot
SUNRPC: Fix the execution time statistics in the face of RPC restarts
Merge branch 'drm-fixes' of git://people.freedesktop.org/~airlied/linux
vmwgfx: Clip cliprects against screen boundaries in present and dirty
vmwgfx: Resend the cursor after legacy modeset
vmwgfx: Do better culling of presents
vmwgfx: Refactor kms code to use vmw_user_lookup_handle helper

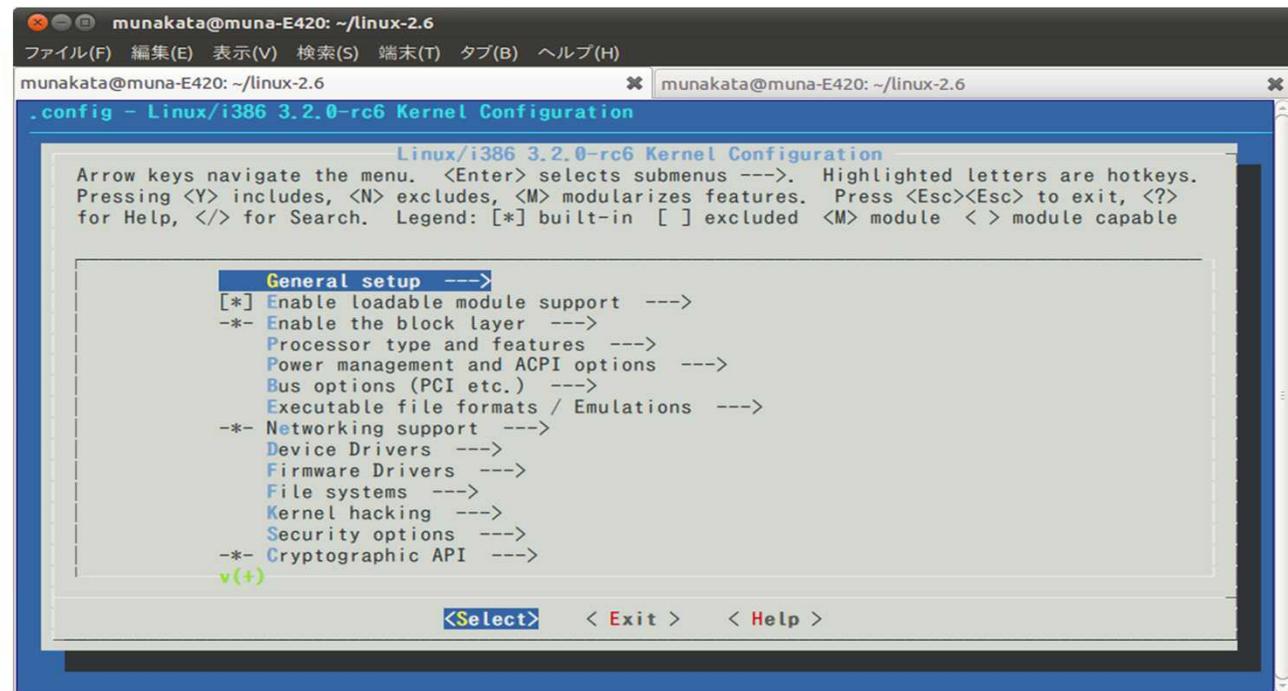
Ohad Ben-Cohen <ohad@wizery.com>
Manuel Lauss <manuel.lauss@googlemail.com>
David Howells <dhowells@redhat.com>
James Morris <jmorris@namei.org>
Dmitry Kasatkin <dmitry.kasatkin@intel.com>
Dmitry Kasatkin <dmitry.kasatkin@intel.com>
Linus Torvalds <torvalds@linux-foundation.org>
Mandeep Singh Baines <msb@chromium.org>
Linus Torvalds <torvalds@linux-foundation.org>
Clemens Ladisch <clemens@ladisch.de>
Linus Torvalds <torvalds@linux-foundation.org>
Kusanagi Kouichi <slash@ac.auone-net.jp>
John Stultz <john.stultz@linaro.org>
NeilBrown <neilb@suse.de>
Linus Torvalds <torvalds@linux-foundation.org>
Robert Richter <robert.richter@amd.com>
Linus Torvalds <torvalds@linux-foundation.org>
Konrad Rzeszutek Wilk <konrad.wilk@oracle.com>
Linus Torvalds <torvalds@linux-foundation.org>
Yoshihiro Shimoda <yoshihiro.shimoda.uh@renesas.com>
Linus Torvalds <torvalds@linux-foundation.org>
Paul Mundt <lethal@linux-sh.org>
Magnus Damm <damm@opensource.se>
Linus Torvalds <torvalds@linux-foundation.org>
Trond Myklebust <Trond.Myklebust@netapp.com>
Linus Torvalds <torvalds@linux-foundation.org>
Jakob Bornecrantz <jakob@vmware.com>
Jakob Bornecrantz <jakob@vmware.com>
Jakob Bornecrantz <jakob@vmware.com>
Jakob Bornecrantz <jakob@vmware.com>

2011-12-20 08:51:38
2011-11-03 19:09:45
2011-12-13 23:49:04
2011-12-21 09:28:16
2011-12-05 20:17:42
2011-12-05 20:17:41
2011-12-21 04:44:18
2011-12-16 04:36:43
2011-12-21 04:43:08
2011-12-20 06:07:58
2011-12-21 04:42:38
2011-12-19 18:13:19
2011-12-13 06:57:52
2011-12-09 07:39:15
2011-12-21 04:41:17
2011-12-20 00:38:30
2011-12-21 04:40:48
2011-12-19 23:30:35
2011-12-21 04:39:21
2011-11-18 16:32:34
2011-12-21 04:32:18
2011-11-24 17:32:59
2011-11-22 15:31:40
2011-11-22 15:44:58
2011-11-22 15:29:54
2011-11-22 15:15:57
2011-11-22 15:23:17
2011-12-21 04:31:56
2011-12-16 08:38:10
2011-12-10 09:05:58
2011-12-10 06:31:52
2011-12-02 06:37:42
2011-12-02 06:31:34
2011-12-02 04:16:17
2011-12-02 04:00:15
2011-12-21 04:31:44
2011-11-28 21:19:15
2011-11-28 21:19:14
2011-11-28 21:19:13
2011-11-28 21:19:12

Fully configurable flexible kernel (one source)

Choosing feature of Linux

- Possible by using “make menuconfig” (or xconfig)
- There are 768 of Kconfig files in 3.1
- 11,662 of configuration items



```
munakata@muna-E420: ~/linux-2.6
munakata@muna-E420: ~/linux-2.6
.config - Linux/i386 3.2.0-rc6 Kernel Configuration

Linux/i386 3.2.0-rc6 Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus ---. Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?>
for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

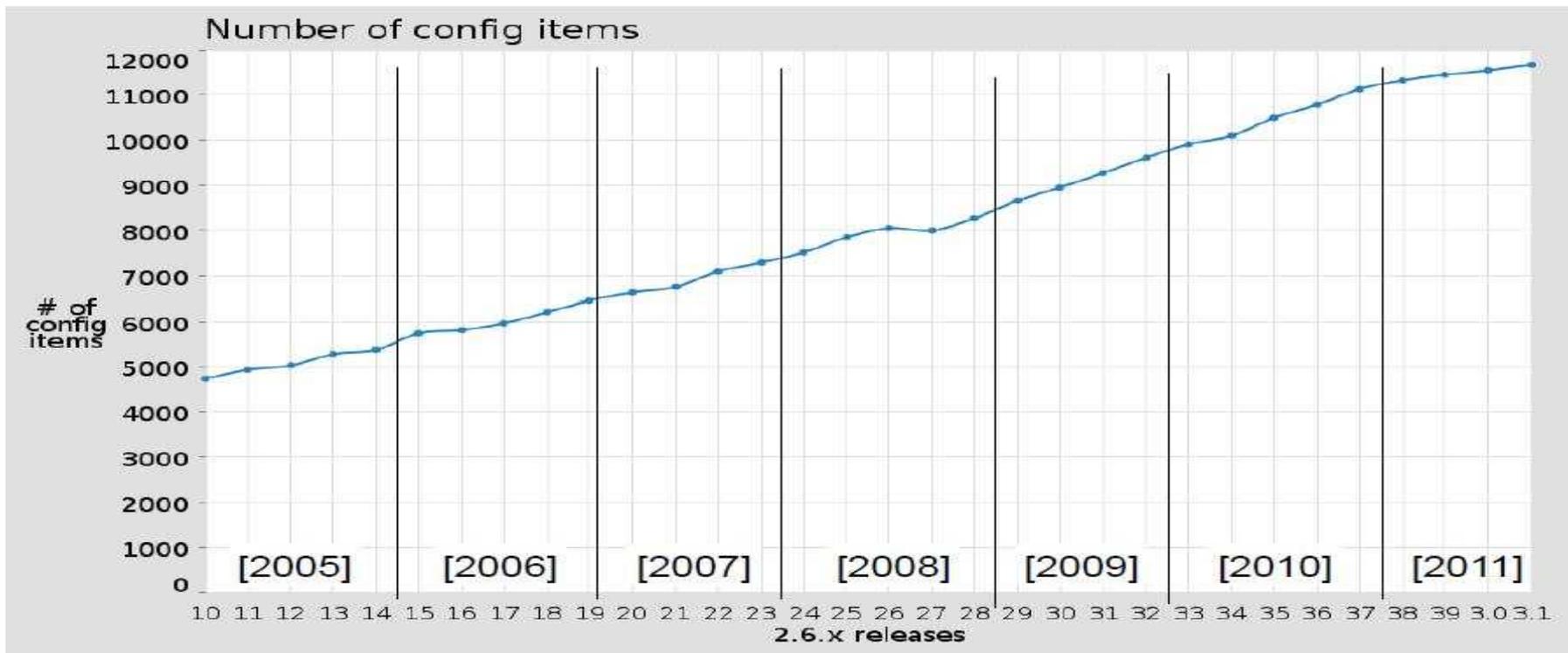
General setup --->
[*] Enable loadable module support --->
[*] Enable the block layer --->
Processor type and features --->
Power management and ACPI options --->
Bus options (PCI etc.) --->
Executable file formats / Emulations --->
[*] Networking support --->
Device Drivers --->
Firmware Drivers --->
File systems --->
Kernel hacking --->
Security options --->
[*] Cryptographic API --->
v(+)

<Select> < Exit > < Help >
```

Still growing to support wider coverage

of configuration items

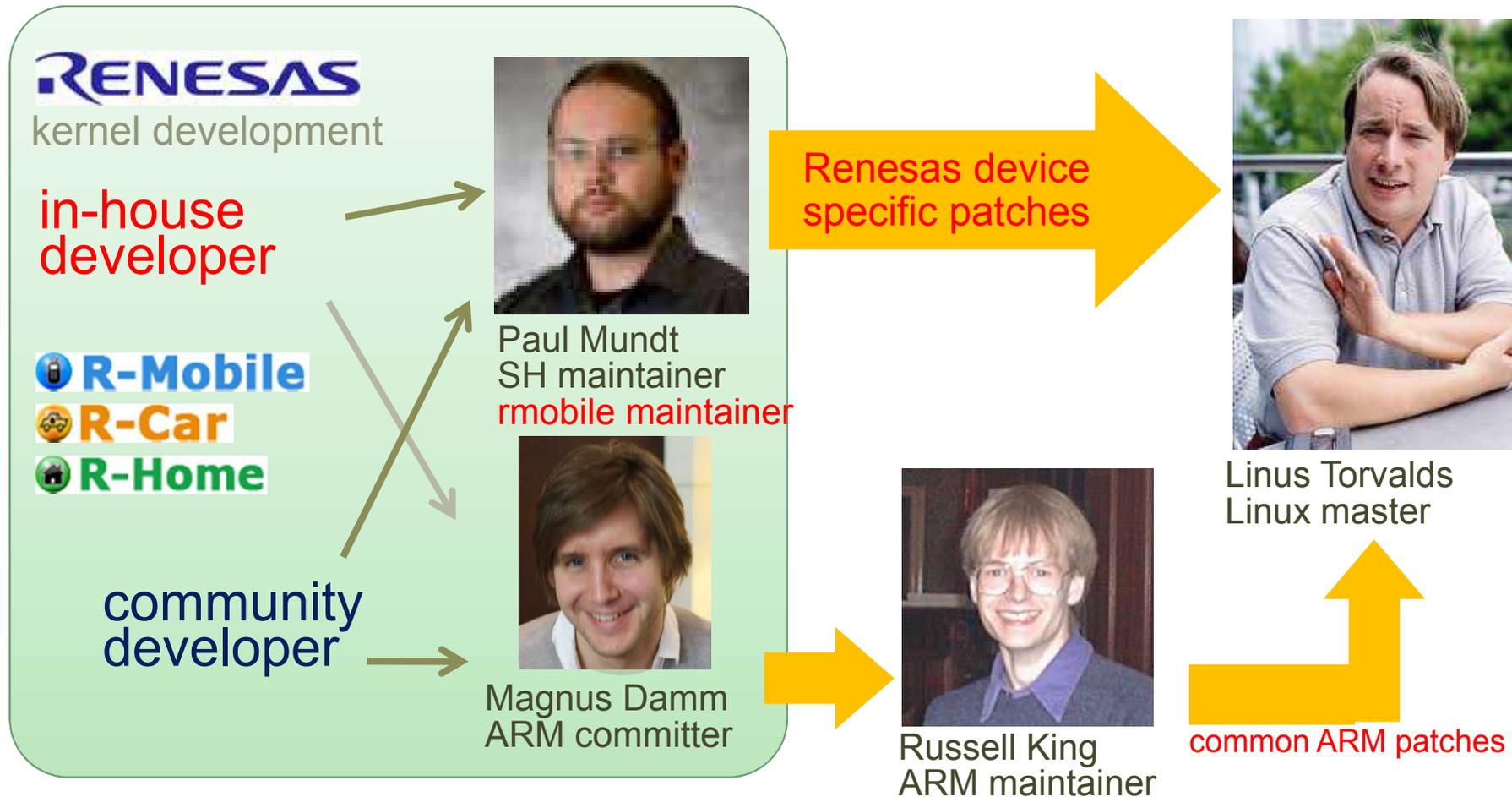
- 11,662 of configuration items in 3.1



Renesas connection to the upstream

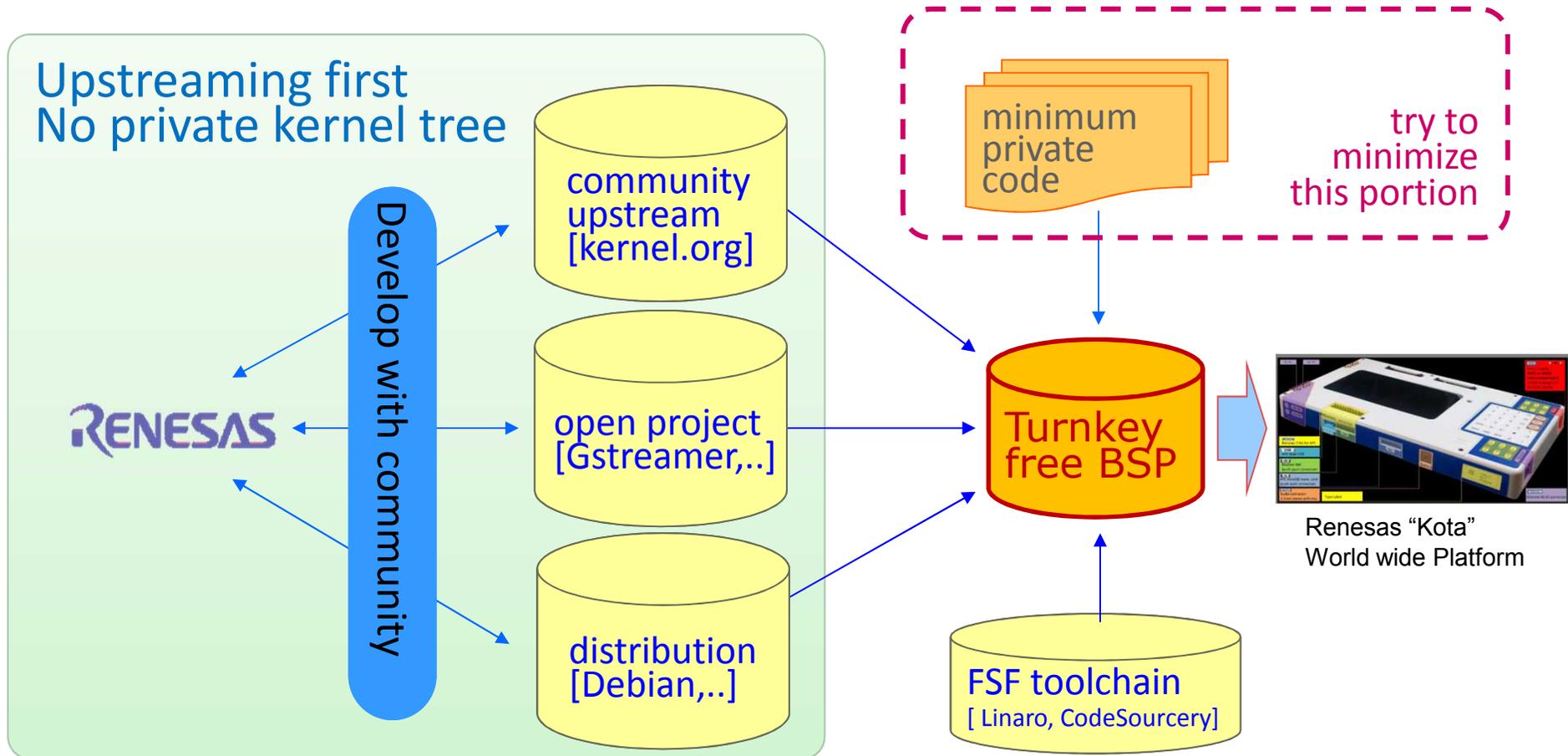
Renesas has ARM kernel submission direct path

- "R-Mobile" is now approved as new ARM sub architecture.
- Renesas utilize dual path for kernel code submission path.



We follow “upstream first” strategy

- Renesas eliminate local fork, not to make Galápagos



We do not have any vendor closed in-house development tree

Latest (= kernel 3.1) contribution result

Of the 182 employers identified as contributing to the 3.1 kernel, the most active were:

Most active 3.1 employers

By changesets			
(None)	1111	13.1%	
Red Hat	882	10.4%	
(Unknown)	749	8.8%	
Intel	616	7.3%	
Broadcom	428	5.1%	
Novell	380	4.5%	
IBM	301	3.6%	
Texas Instruments	276	3.3%	
(Consultant)	223	2.6%	
Freescale	182	2.2%	
Linaro	170	2.0%	
Samsung	162	1.9%	
Google	150	1.8%	
Wolfson Microelectronics	142	1.7%	
Fujitsu	131	1.5%	
Renesas Electronics	100	1.2%	
Oracle	82	1.0%	
MITAC	80	0.9%	
Nokia	79	0.9%	
(Academia)	73	0.9%	

By lines changed			
Novell	162583	19.8%	
(None)	90119	11.0%	
Broadcom	76810	9.4%	
Red Hat	58262	7.1%	
Intel	43505	5.3%	
(Unknown)	27109	3.3%	
Metzler Brothers Systementwicklung GbR	23681	2.9%	
Samsung	23238	2.8%	
Rising Tide Systems	23090	2.8%	
IBM	22231	2.7%	
Texas Instruments	21130	2.6%	
Freescale	17270	2.1%	
Brocade	16587	2.0%	
Realsil Microelectronics	15868	1.9%	
Wolfson Microelectronics	14004	1.7%	
(Consultant)	13710	1.7%	
South Pole AB	12087	1.5%	
Linaro	11129	1.4%	
Oracle	9390	1.1%	
Nokia	7450	0.9%	

Broadcom's extensive work to move its wireless driver out of staging caused it to move to a higher than usual position on both lists. Also notable is the continued slow climb by companies like Texas Instruments and Samsung; Nokia, instead, appears to be about to fall out of the top 20. The handling of Linaro deserves an explanation: contributions by Linaro assignees is normally credited back to their home companies. Nonetheless, Linaro makes an appearance on its own here as the result of the work of an increasing number of engineers employed by the organization itself.

<http://lwn.net/Articles/460597/>

Connected to commercial quality BSP creation

If you build Linux BSP with 3.0 LTS kernel, you can always cherry-pick code from newer kernel that Renesas keep maintaining. This can be perfect resource for commercial BSP.

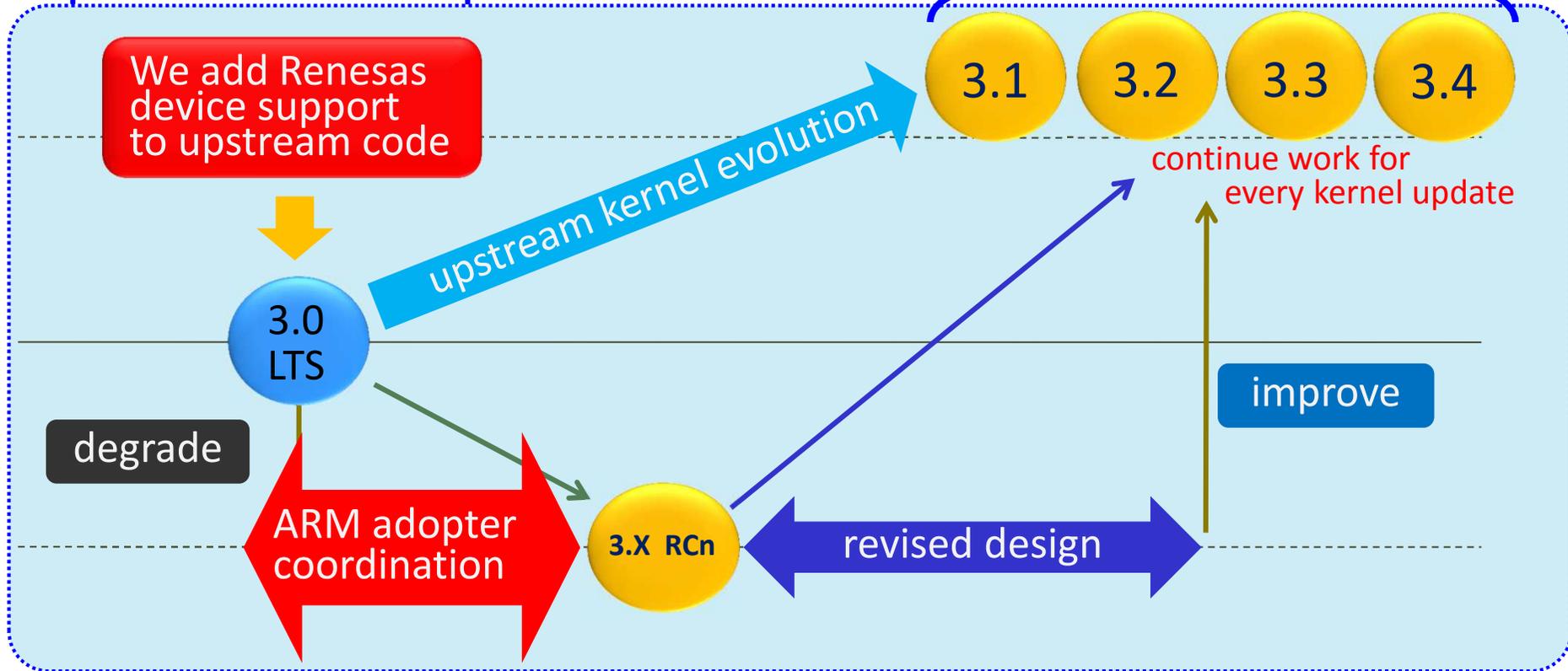
commercial quality BSP

cherry-pick appropriate version

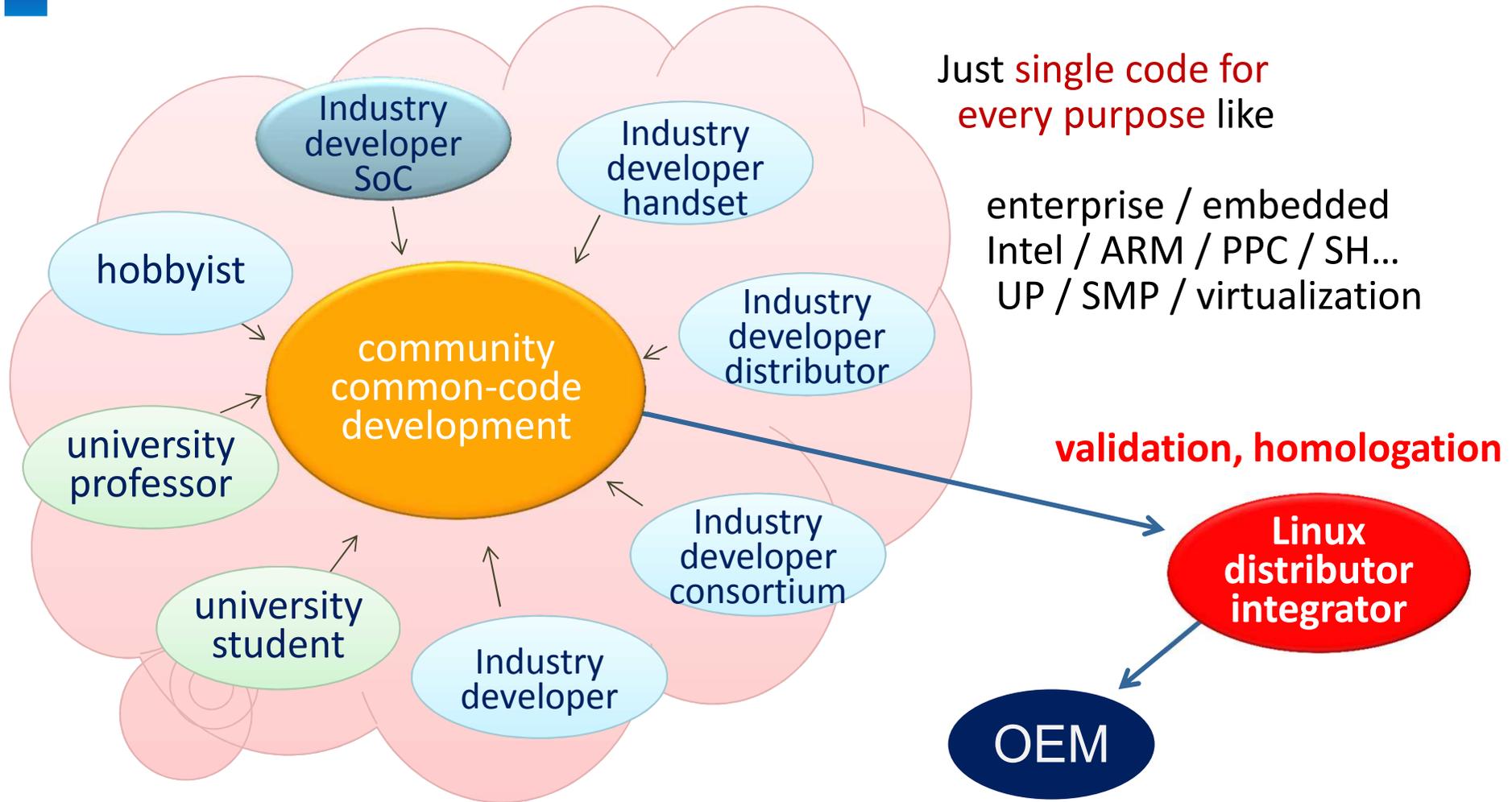


validation, stress test regression db

Upstream development

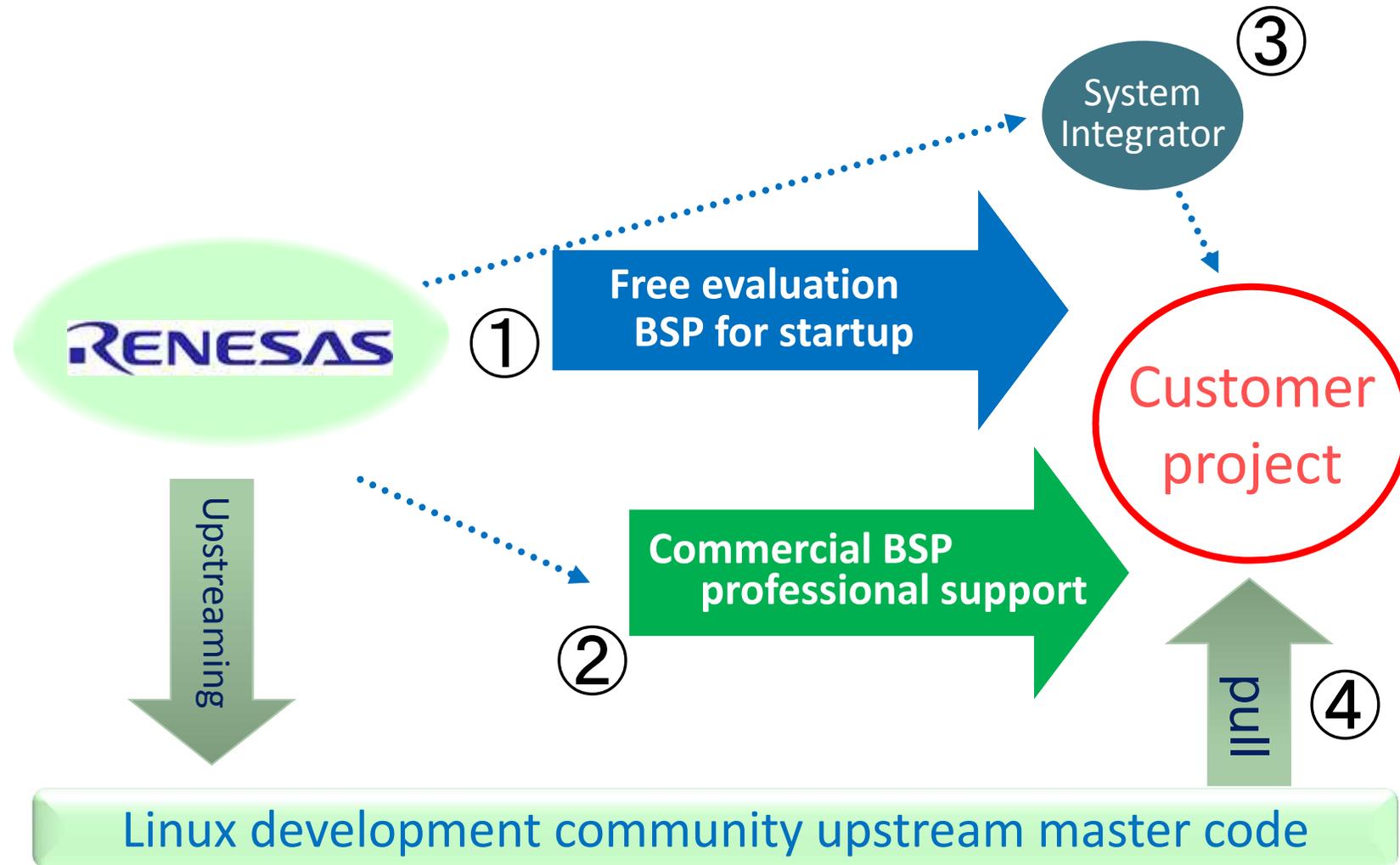


We work with the dedicated Linux integrator

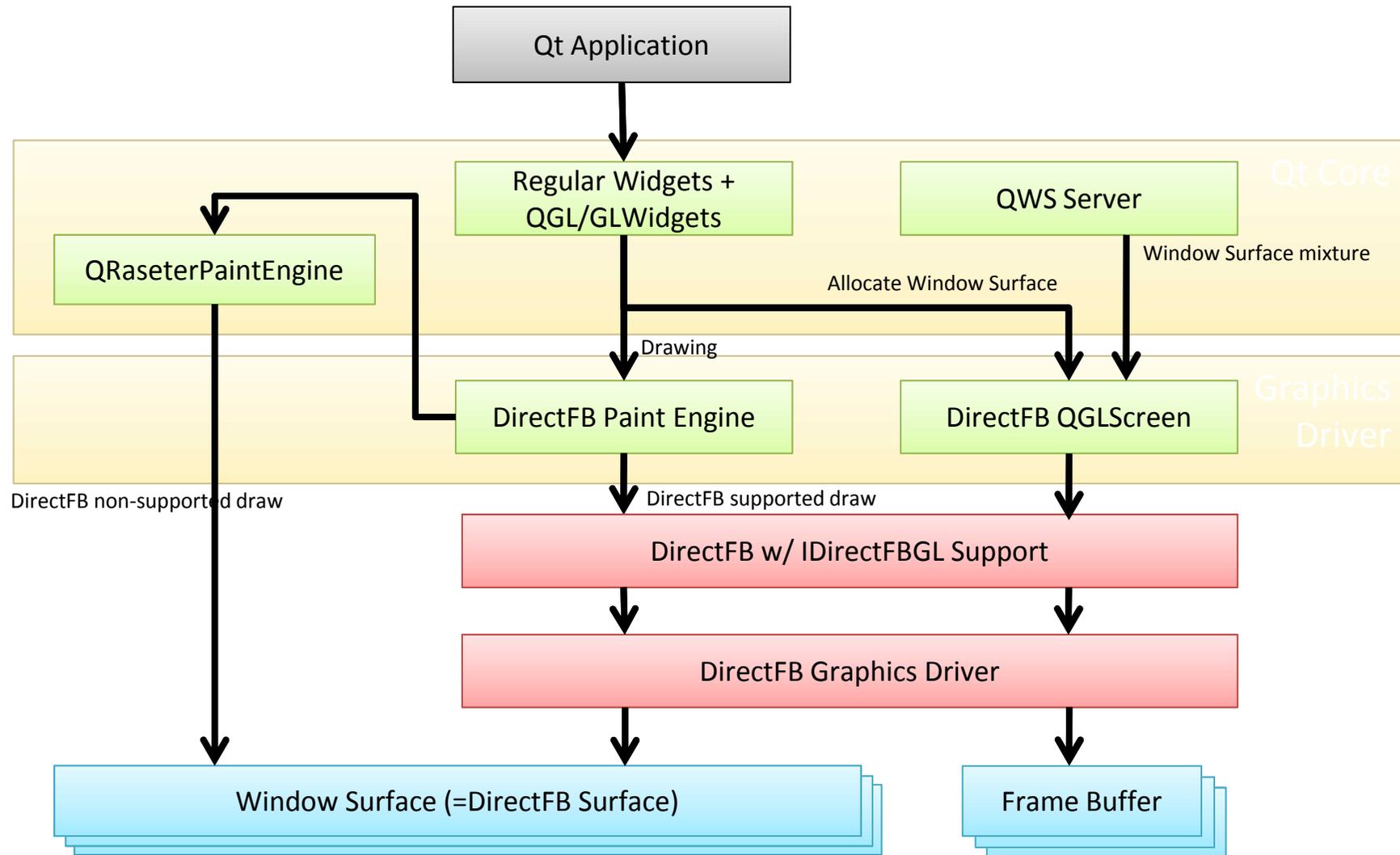


Renesas deliver mainlined common code to Linux integrator

4-way Linux support channel

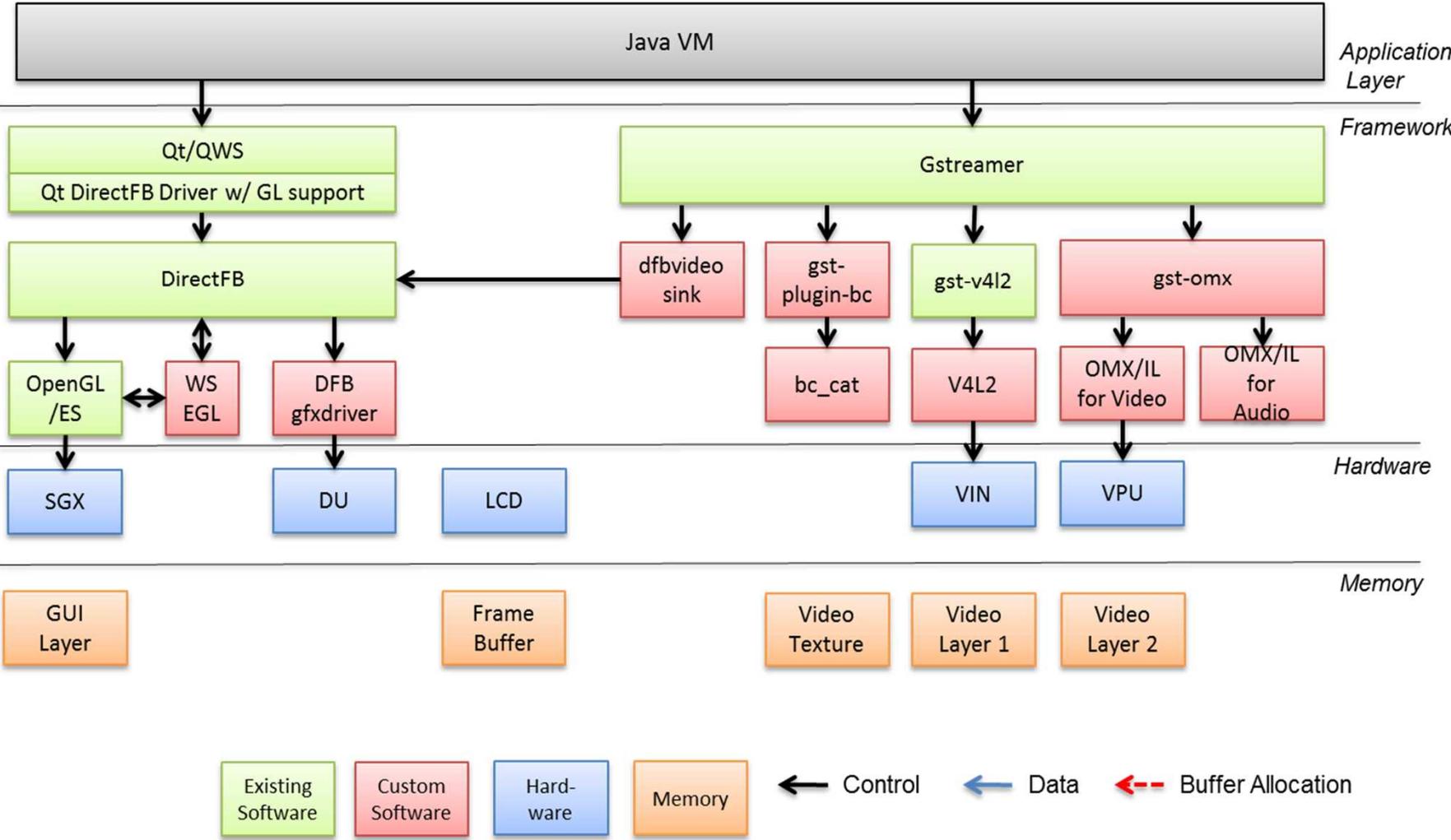


Qt + SGX integration guide (from Nokia)

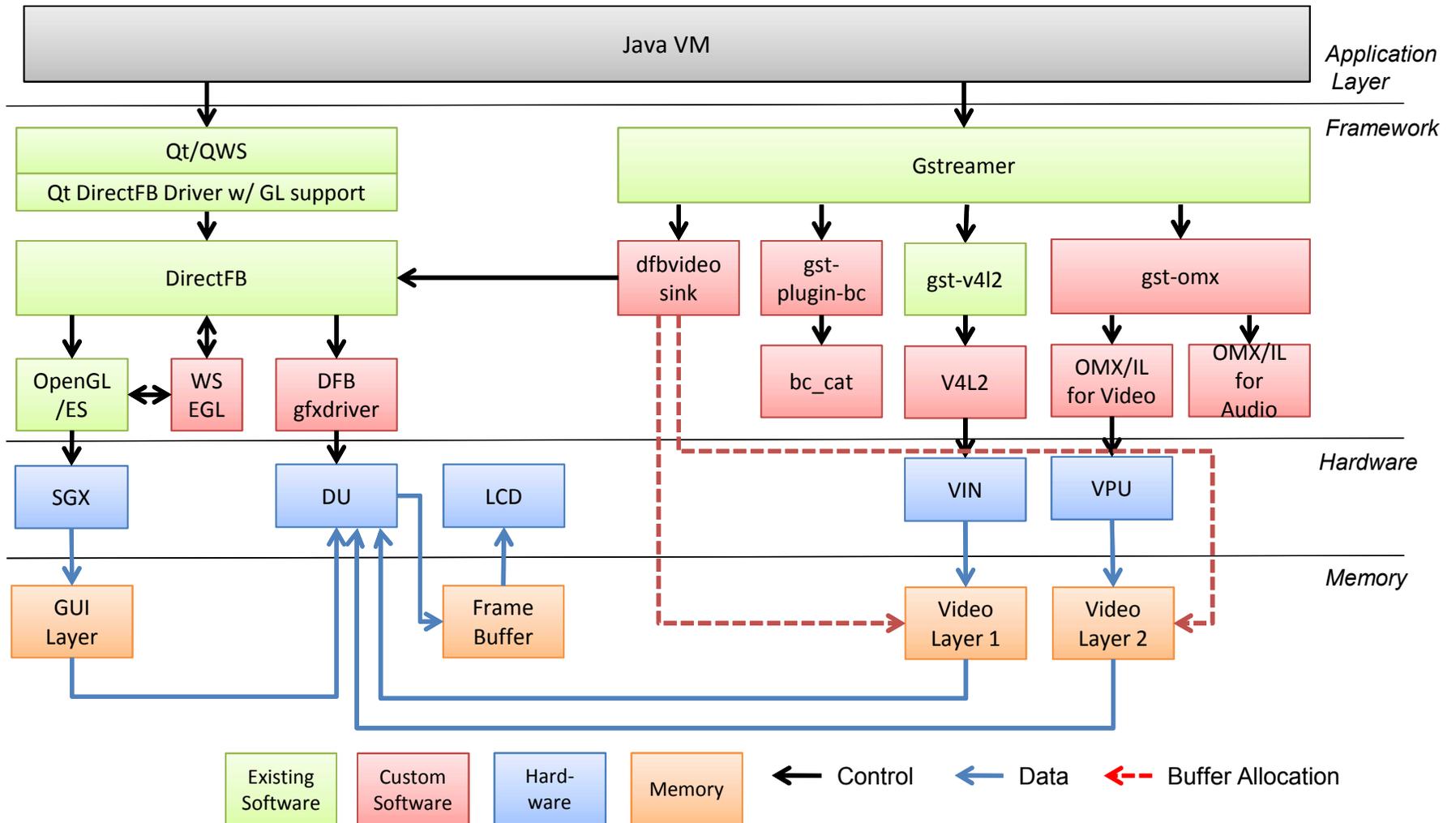


<http://doc.qt.nokia.com/latest/qt-embeddedlinux-directfb.html>

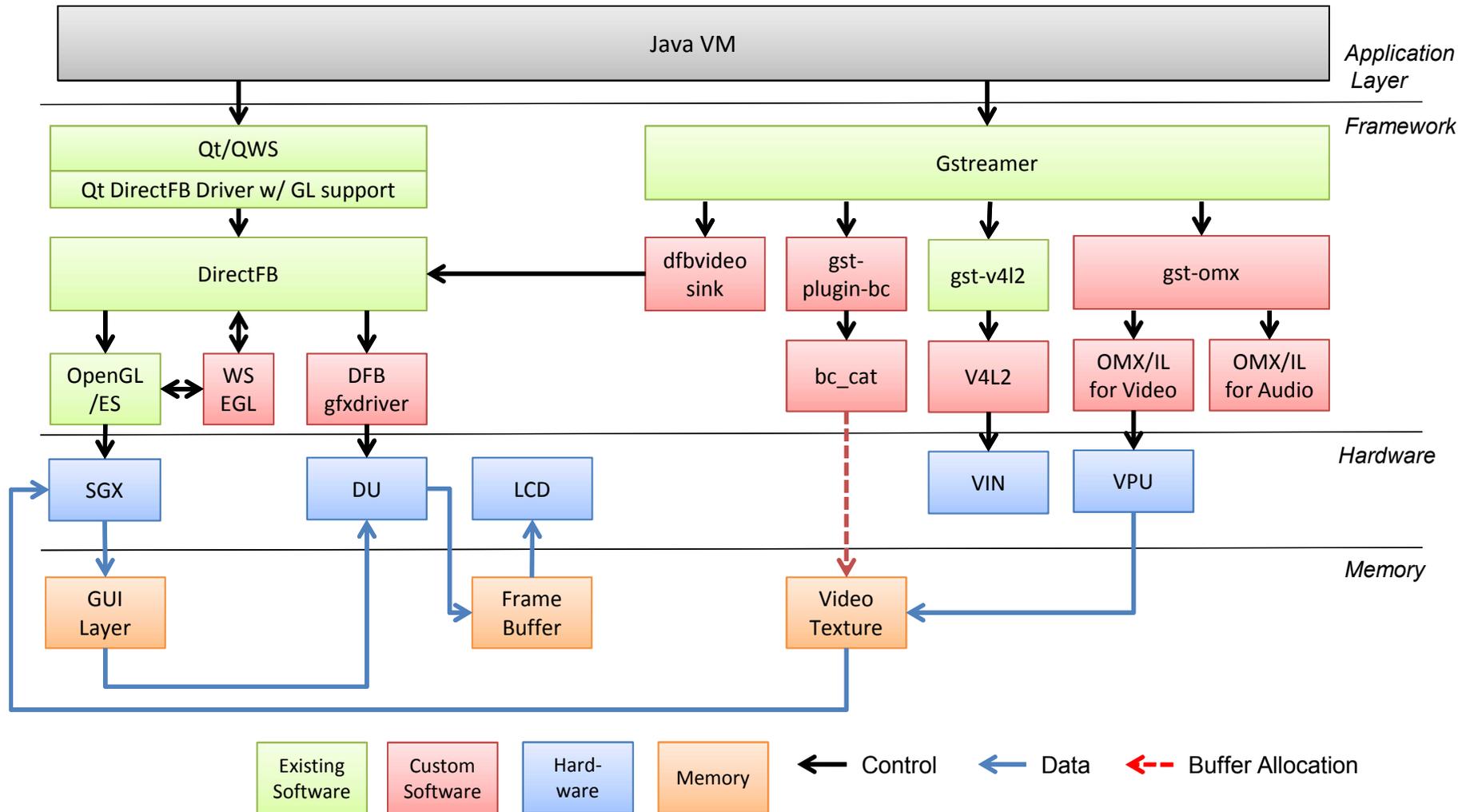
Renesas Qt + SGX integration architecture



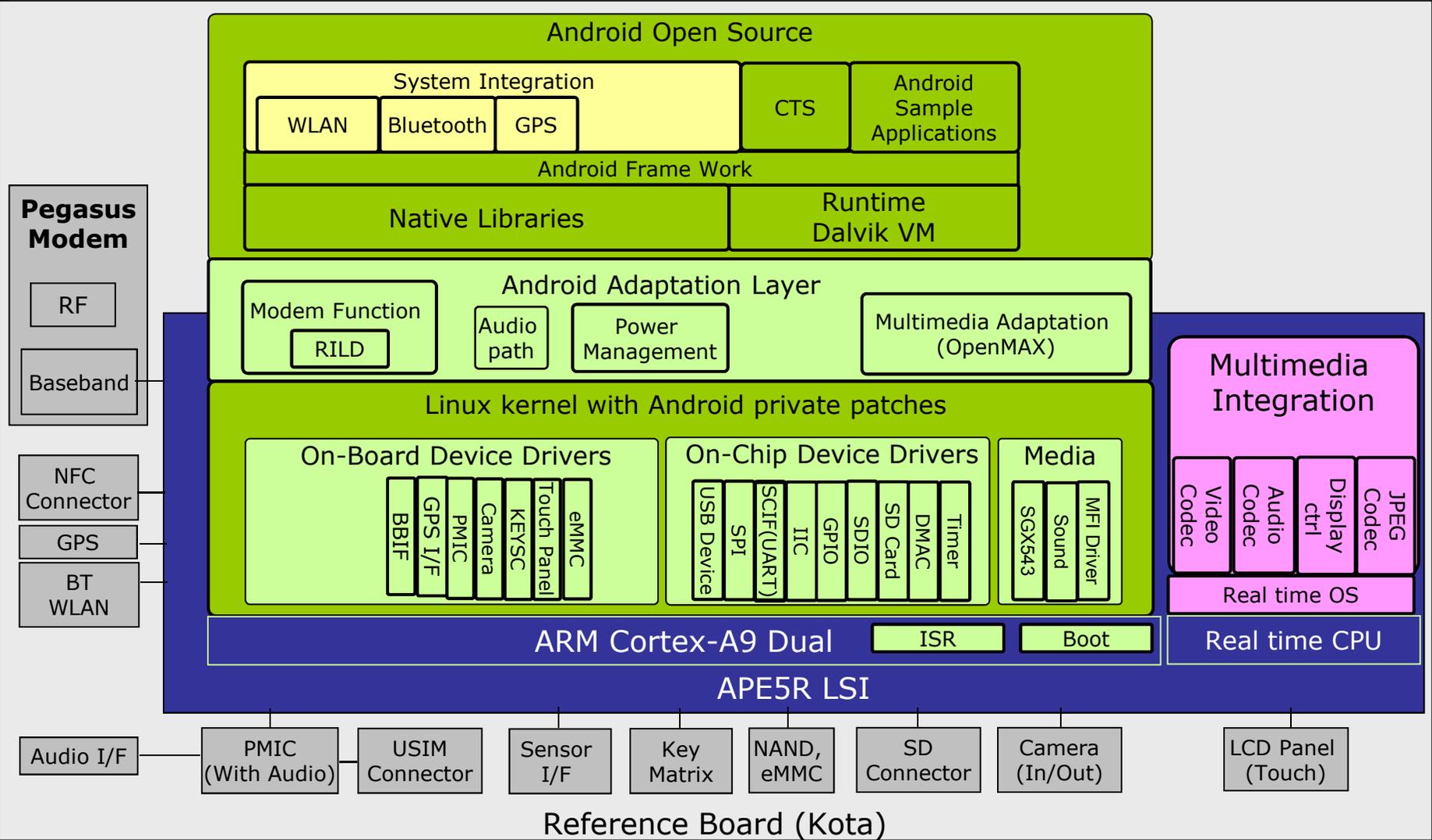
Data & control flow : Full Screen Video Playback



Data & control flow : Video as Texture

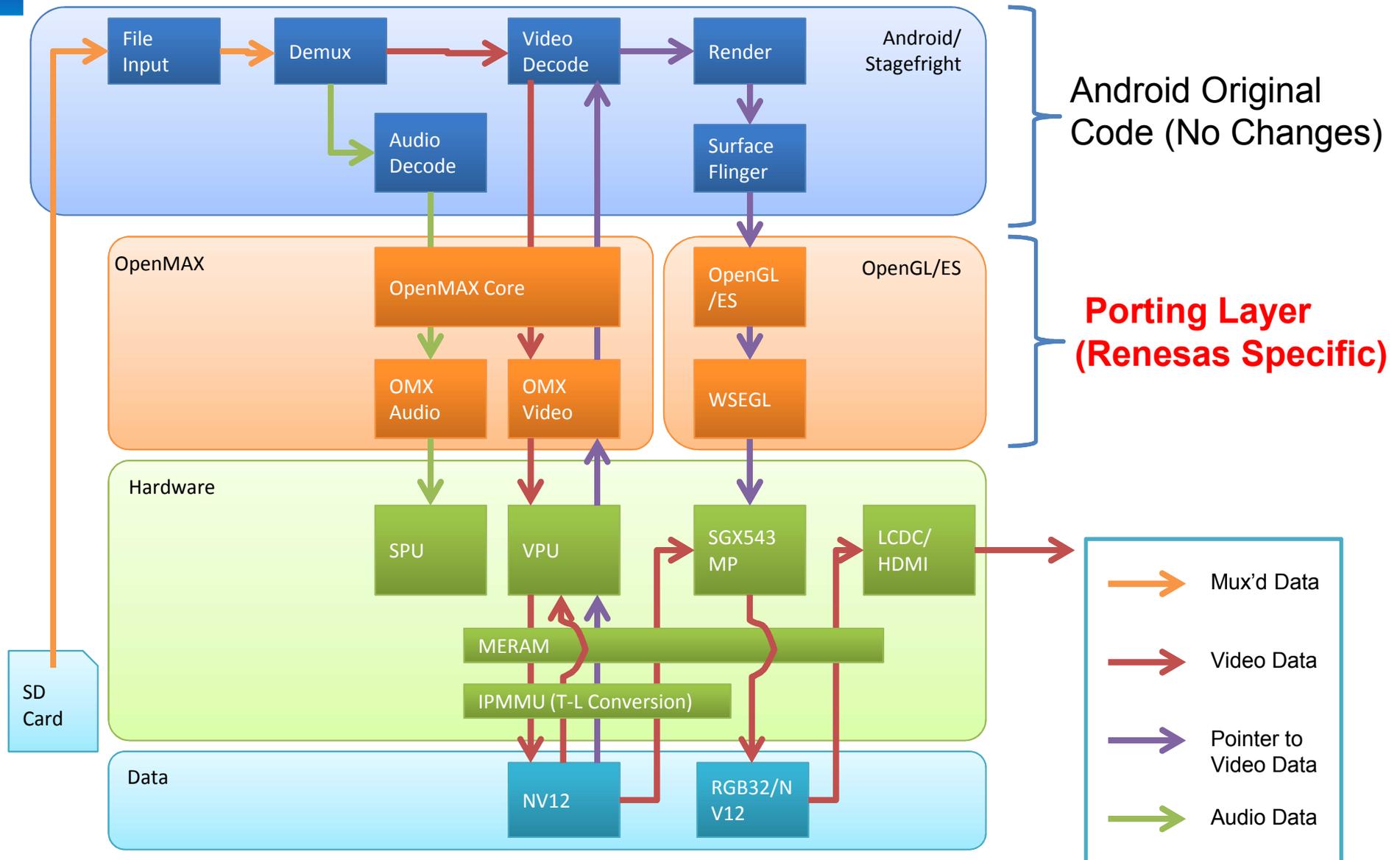


Renesas ODIN software architecture (Android)



Open Source
 Renesas SW (Application domain)
 Renesas SW (Real time domain)
 Partner SW

Renesas Android multi-media integration



Tentative logo



LTSI : Long Term kernel support initiative

The Linux Foundation Announces Consumer Electronics Long Term Stable Kernel Initiative

By Linux_Foundation - October 25, 2011 - 9:48pm

New kernel tree addresses electronics makers' need for regular updates and long-term support

PRAGUE (LinuxCon Europe) October 26, 2011 – The Linux Foundation, the nonprofit organization dedicated to accelerating the growth of Linux, today announced it is hosting a new project created by its Consumer Electronics (CE) workgroup.

<http://www.linuxfoundation.org/news-media/announcements/2011/10/linux-foundation-announces-consumer-electronics-long-term-stable-ke>

LTSI : long term kernel support initiative



- Home
- What is LTSI?
- Developers
- Users
- Downloads
- Participate

The LTSI is an ecosystem-wide collaborative project hosted at the Linux Foundation to create and maintain a common Linux base for the use in a variety of CE products and to enable faster contributions upstream and better alignment with the mainline kernel.

LTSI Blog

The latest from LTSI.

Introducing the LTSI Project

Submitted by brian on December 26, 2011 - 00:42

On October 25, 2011, [The Linux Foundation](#), the nonprofit organization dedicated to the growth of Linux, announced it is hosting a new project created by its Consumer Electronics Workgroup.

The new project, the Long Term Support Initiative (LTSI), provides a regular release of a Linux kernel suitable for supporting the lifespan of consumer electronics products and regular updates of those releases for two years.

[more blogs](#)

LTSI Releases

LTSI Trees.

- Current LTSI Tree: [3.0.0](#)
- Current LTSI Staging Tree: [3.0.0](#)
- Current Industry Staging Tree: [3.0.0](#)

Participating Organizations

See who is involved in LTSI.

HITACHI



NEC

Panasonic



RENESAS



SONY

Tentative web site image

Community stable kernel support policy change

Possible changes to longterm kernel maintenance

[Posted August 13, 2011 by corbet]

Greg Kroah-Hartman has posted [a proposal](#) for some changes to how the stable and (especially) longterm kernels are maintained. The changes are being driven by users other than the enterprise distributors. "Now that 2.6.32 is over a year and a half, and the enterprise distros are off doing their thing with their multi-year upgrade cycles, there's no real need from the distros for a new longterm kernel release. But it turns out that the distros are not the only user of the kernel, other groups and companies have been approaching me over the past year, asking how they could pick the next longterm kernel, or what the process is in determining this." The core idea is to pick a new longterm kernel once a year; that kernel would then be maintained for two years thereafter. There is [some discussion](#) on Google+; it should move to the mailing list around August 15.

<http://lwn.net/Articles/454886/>

LTSI vs. LTS (community stable kernel)

LTSI : long term support kernel

- Back-port **newly added device / platform** support from new kernel.
- Back-port **new kernel feature** to improve code functionality like PM.
We can apply **newly added (to 3.2 and later) Android code** to 3.0.
- SoC vender specific new code that are still in Linux-next can be added.
- Some embedded target kernel enhancement also can be added.

Community LTS

- Safe bug fix already verified and merged new kernel.
- Serious security fix already verified and merged new kernel.

Upstream vanilla kernel

Renesas will add our latest SoC support into stable kernel with LTSI

You are special and unique, just like everyone else.



<http://elinux.org/images/c/ca/Gregkh-android-presentation-kernel.pdf>

Resolution of past Android isolation issue

```
commit b0a0ccfad85b3657fe999805df65f5cfe634ab8a
Author: Greg Kroah-Hartman <gregkh@suse.de>
AuthorDate: Mon Oct 5 16:29:29 2009 -0700
Commit: Greg Kroah-Hartman <gregkh@suse.de>
CommitDate: Fri Dec 11 12:23:21 2009 -0800
```

Staging: android: delete android drivers

These drivers are no longer being developed and the original authors seem to have abandoned them and hence, do not want them in the mainline kernel tree.

So sad :(

```
Cc: Brian Swetland <swetland@google.com>
Signed-off-by: Greg Kroah-Hartman <gregkh@suse.de>
```

Android discussion @kernel summit 2011



Kernel Summit surprise

- Android
 - Android was not on the agenda
 - I was gearing up for a big fight over mainlining Android features
 - Ingo Molnar, Ted Tso, Linus Torvalds, Paul McKenny and others advocated re-examining the Android patches
 - Someone still needs to submit them (Google is in no mood)
- We need to keep fighting bloat
 - New logging system proposed which added minimal overhead of 16 bytes per message!!

<http://elinux.org/images/7/70/Status-of-Embedded-Linux-2011-12-JJ39.pdf>

Android Mainlining Project @elinux wiki

Patch/Feature Status Chart

Feature/Patch	Description	Status	Part of core?	Owner/Interested parties	Notes
logger	kernel support for Android system logging	not mainlined (but see linux-next staging as of 12/19/11)	yes	Tim Bird	should be non-controversial (though I'm always surprised) See Mainline Android logger project for a list of ideas, issues and a project plan for this feature Also see this LKML discussion thread
wakelocks	Power management locking mechanism to prevent opportunistic suspend	not mainlined	yes	Rafael Wysocki	Is important due to impact on board support and drivers by 3rd parties
Android alarm timers	Timers that count down during suspended operation, and can wake from suspend	Partial: Posix alarm timers were mainlined in kernel version 2.6.38 - see https://lwn.net/Articles/429925/	yes	John Stultz	Mending patches to convert Android Alarm Timers to utilize the upstreamed alarm timer work are still pending.
ashmem	Shared memory implementation that allows unpinned pages to be marked, which can be dropped by the kernel under memory pressure	not mainlined	yes	John Stultz	Working on fadvise volatile alternative implementation that handles part of the ashmem functionality. However, there are additional aspects of ashmem design that need to be addressed(no tmpfs mounts, atomic create/unlink behavior,etc).
network security	special permission checks for secure access to network operations	not mainlined	? (can run without it, but network security won't	no one	May be very difficult to mainline, as the code is extremely Android-specific with hardcoded GIDs and capabilities.

Target feature set list is continued

http://elinux.org/Android_Mainlining_Project

Renesas is leading Runtime PM integration to Android now

Runtime PM (Power Management) update

Better power management to achieve longer battery operation time is getting essential requirement for modern smartphone as its CPU clock and core number is becoming bigger and bigger. At the same time PM code is most touchy part of Android as it has adopted non standard way of power-state control. Renesas is working hard to solve this issue using already mainlined “Runtime-PM” concept.

Linux (and Android) key power control factors

■ CPU

- CPU hot-plug (per core of/off control)
- CPI idle (cut down unused core power)

These control can be integrate to DVFS control.

■ Device

- Device clock control
- per on-chip IP/on-board-device control
- **per power-domain** power control

Runtime PM is designed to manage all these control points.

Trial Runtime PM integration on Renesas SoC

Power Management Using PM Domains on SH7372

Rafael J. Wysocki

Faculty of Physics U. Warsaw / Renesas Electronics / SUSE Labs

October 27, 2011

Rafael J. Wysocki (rjw@sisk.pl)

Power Management on SH7372

October 27, 2011

1 / 21

Renesas work with community leading PM developer

Trial Runtime PM integration on Renesas SoC

Runtime Power Management Power Management QoS

Power Management Quality Of Service (PM QoS)

Timing information is important for power management

- 1 When a device is suspended or the CPU is idle, the kernel has to decide whether or not to remove power from its domain (or its master).
- 2 All domain and device PM operations take time to complete.
- 3 CPUidle needs to know how much time that is.
- 4 There may be requirements regarding the time devices can spend in the non-working state.
- 5 The device suspend code should take those requirements into account.

Device PM QoS

Framework allowing kernel subsystems to specify wakeup latency constraints for I/O devices (scheduled for inclusion into the 3.2 kernel).

Rafael J. Wysocki (rjw@sisk.pl) Power Management on SH7372 October 27, 2011 14 / 21

“PM QoS” is very important mechanism to better efficiency

Trial Runtime PM integration on Renesas SoC

Runtime Power Management | I/O Device Power Management

I/O Runtime PM Framework

Turning devices off (when idle) and on

- The core:
 - ① Handles concurrency (locking etc.).
 - ② Takes care of device dependencies (parents vs children).
 - ③ Provides reference counting facilities (detection of idleness).
 - ④ Provides common helpers (e.g. `pm_runtime_suspend()`).
- Subsystems and drivers:
 - ① Provide callbacks.
 - ② Handle wakeup events (remote wakeup).

Power domains have to be taken into account.

Subsystem callbacks may be overridden by power management domain callbacks (representation via `struct dev_pm_domain`).

Rafael J. Wysocki (rjw@sisk.pl) | Power Management on SH7372 | October 27, 2011 | 7 / 21

ARM Linux updates

Historically ARM corporation provides CPU core portion only and each SoC company designed own Interrupt Controller, Timer and DMA to make their products. Due to this reason Linux ARM code becomes pretty duplicated and confused status compared to other major CPU architecture. 2012 will be the first year to re-organizing ARM code in Linux kernel.

ARM Linux kernel issues

- Current confusion
 - vendor private tree integration issue.
 - upstreaming path confusion issue.
 - Linaro start fixing this with **ARM-SoC** tree
- No common platform like x86 PC
(No single binary kernel image for ARM)
 - **DT (Device Tree)** would be the solution.
- New core (**CA15, bigLITTLE** etc.) support

1st ARM maintainer workshop @kernel summit

Events/Kernel Summit 2011 ARM Subarch Maintainership Workshop

We're currently planning an ARM Subarchitecture Maintainership Workshop for the first day of Kernel Summit in Prague, October 23, 2011. Watch this page for details, or contact Grant Likely, Nicolas Pitre, or Arnd Bergmann.

There is limited space available for this meeting. Seats were provided on a first-come, first-served basis and we have now surpassed our self-imposed limit of 24 people. However, if you still would like to attend, or would like to nominate someone who you think is an important participant, then email Grant Likely <grant.likely@secretlab.ca>. We may be able to open up additional seats as needed.

We're also extending an open invite to all kernel summit invitees. If you've been invited to kernel summit and you'd like to attend the ARM workshop, then email Grant so that we know you are coming. Otherwise we'll make you sit on the floor.



http://elinux.org/Events/Kernel_Summit_2011_ARM_Subarch_Maintainership_Workshop

Renesas is a part of future ARM kernel consolidation discussion

Working with Linaro



Linaro is a membership organization to develop common ARM Linux kernel code as an open project. Renesas is not a member of Linaro so far, but we can work with Linaro project from the upstream community support connection.

- We plan to adopt Linaro toolchain for CA15,..
- Linaro will follow LTSI project.
- LinaroConnect 12Q1 and Embedded Linux Conference [ELC2012] is co-located in February 2012 @Redwood city CA. USA.

Conclusion

- Renesas had been contributing a lot for Linux and opensource development. Therefore we understand how to collaborate with community.
- We are in the circle of latest advanced software discussion like advanced PM integration or ARM kernel renovation. So we can add most advanced software technology into our BSP delivery.
- Linux and Android software platform is huge resources that requires broad range of support. We are flexible to collaborate with any distributor which handset vendor prefers to achieve that.