Best practice of Open Source Operating system adoption

How to realize high-quality, availability and maintainability

Hisao Munakata
Automotive Information Solution division,
Renesas Electronics Corp.

Who am I

- From automotive target SoC provider company Renesas.
- Leading Linux kernel upstream development team.
- AGL (Automotive Grade Linux) advisory board member.
- LTSI (longterm kernel) and yocto (build tool) project board member.

Most active 4.6 employers

By changesets			By lines changed		
Intel	2009	15.0%	Intel	131992	20.0%
(Unknown)	1358	10.2%	(Unknown)	43271	6.5%
Red Hat	1043	7.8%	Red Hat	40589	6.1%
(None)	647	4.8%	(None)	29098	4.4%
Linaro	588	4.4%	Linaro	19259	2.9%
Outreachy	413	3.1%	IBM	17455	2.6%
Samsung	390	2.9%	Outreachy	17445	2.6%
SUSE	364	2.7%	Omnibond Systems	17122	2.6%
Renesas Electronics	336	2.5%	Texas Instruments	16779	2.5%
IBM	312	2.3%	Code Aurora Forum	14484	2.2%
AMD	307	2.3%	Renesas Electronics	14222	2.2%
ARM	253	1.9%	Google	14183	2.1%
Google	247	1.8%	AMD	13581	2.1%
(Consultant)	238	1.8%	SUSE	12304	1.9%
Texas Instruments	230	1.7%	Samsung	12107	1.8%
Oracle	191	1.4%	Oracle	11441	1.7%
Code Aurora Forum	175	1.3%	Mellanox	10971	1.7%
Atmel	166	1.2%	ARM	10811	1.6%
NVidia	159	1.2%	Facebook	9642	1.5%
Huawei Technologies	147	1.1%	(Consultant)	8086	1.2%

Latest Linux kernel contributor company ranking

https://lwn.net/Articles/686393/

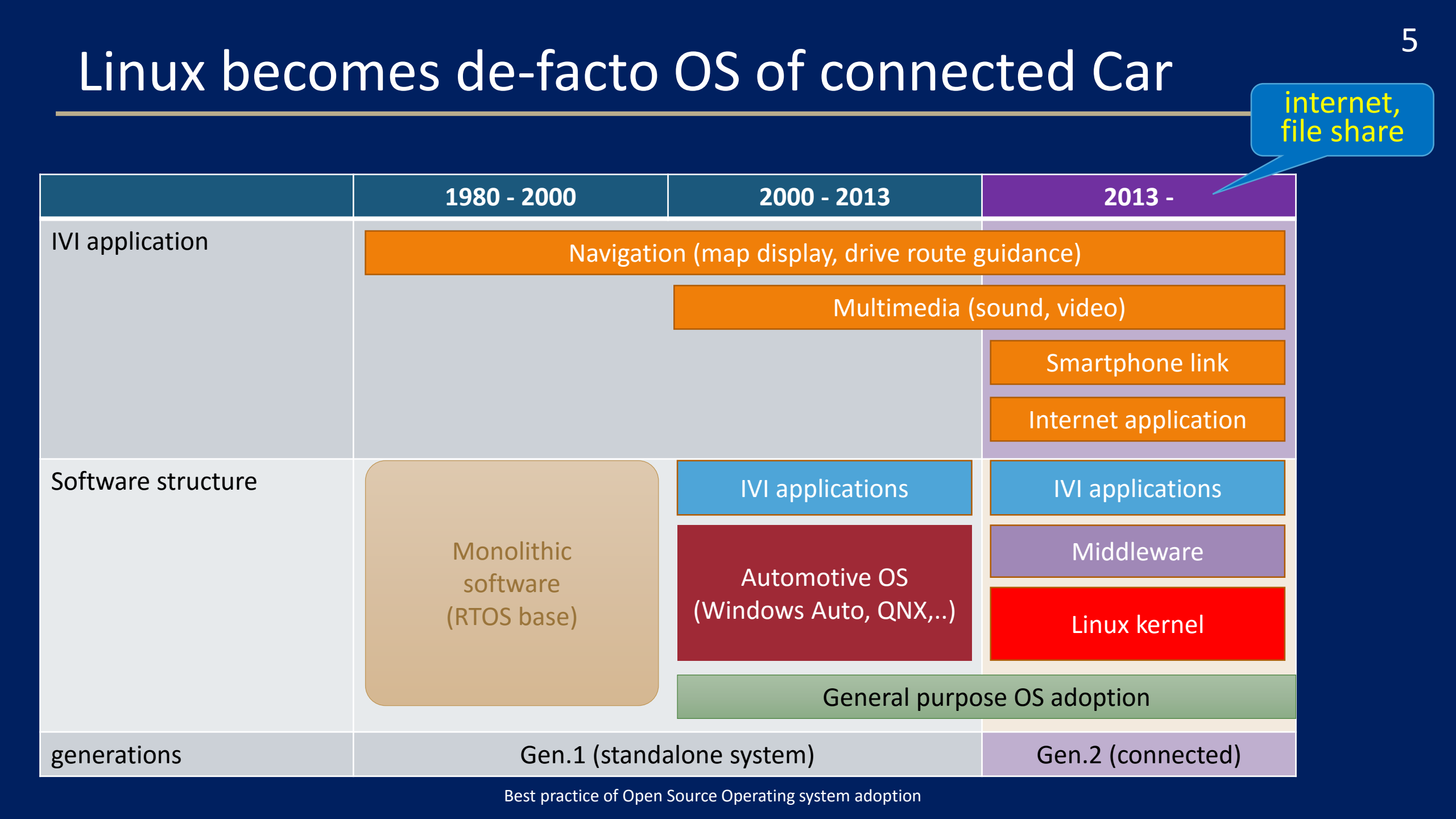
Agenda

- Software structure of IVI (In Vehicle Information) system
  - IVI software trend and architecture
  - RTOS vs. Linux philosophy comparison
  - Challenge of connected car software
- The fact of Linux kernel
  - Linux kernel summary
  - development community
  - Transparency and openness of OSS development process
- OSS adoption challenge for IVI system
  - Code quality
  - Security risk management

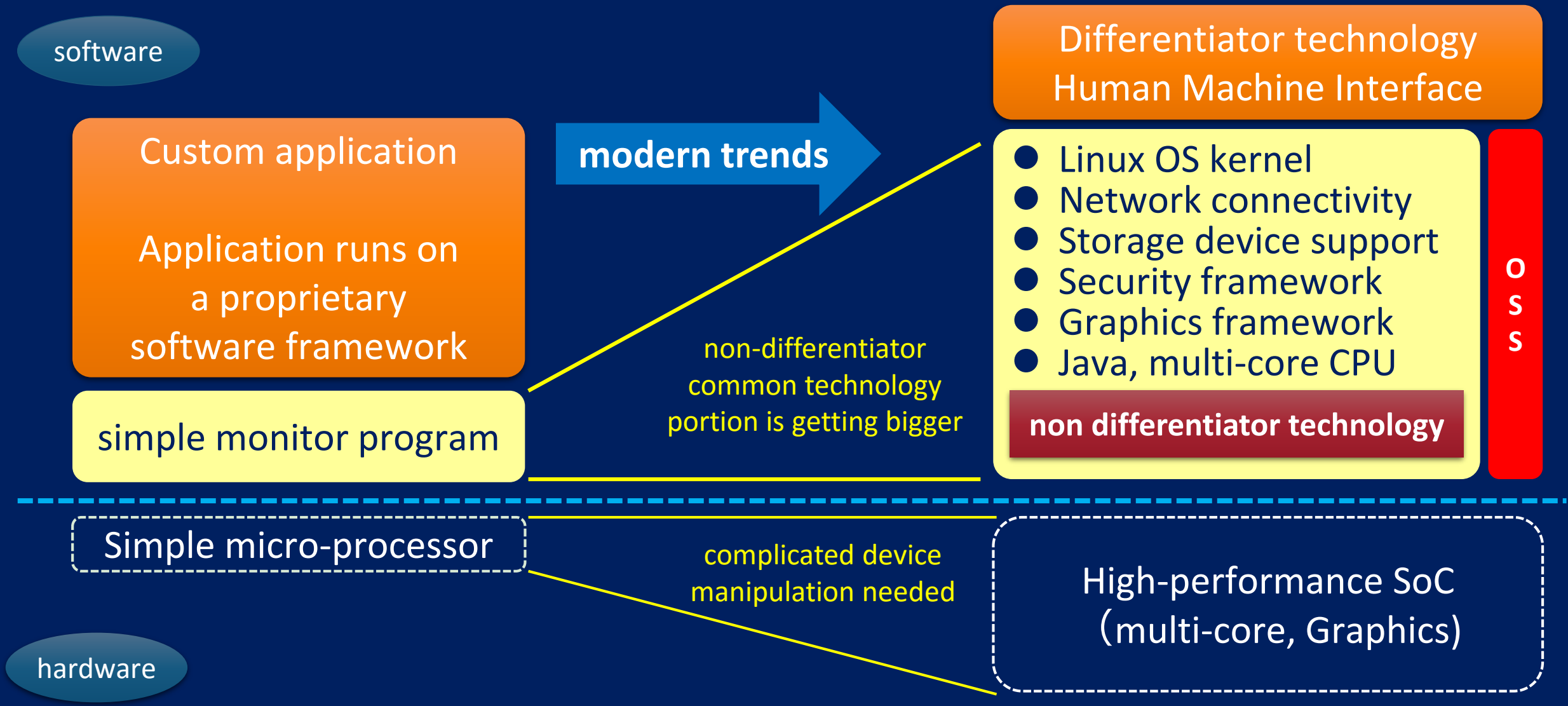
Agenda

- Software structure of IVI (In Vehicle Information) system
  - IVI software trend and architecture
  - RTOS vs. Linux philosophy comparison
  - Challenge of connected car software
- The fact of Linux kernel
  - Linux kernel summary
  - development community
  - Transparency and openness of OSS development process
- OSS adoption challenge for IVI system
  - Code quality
  - Security risk management

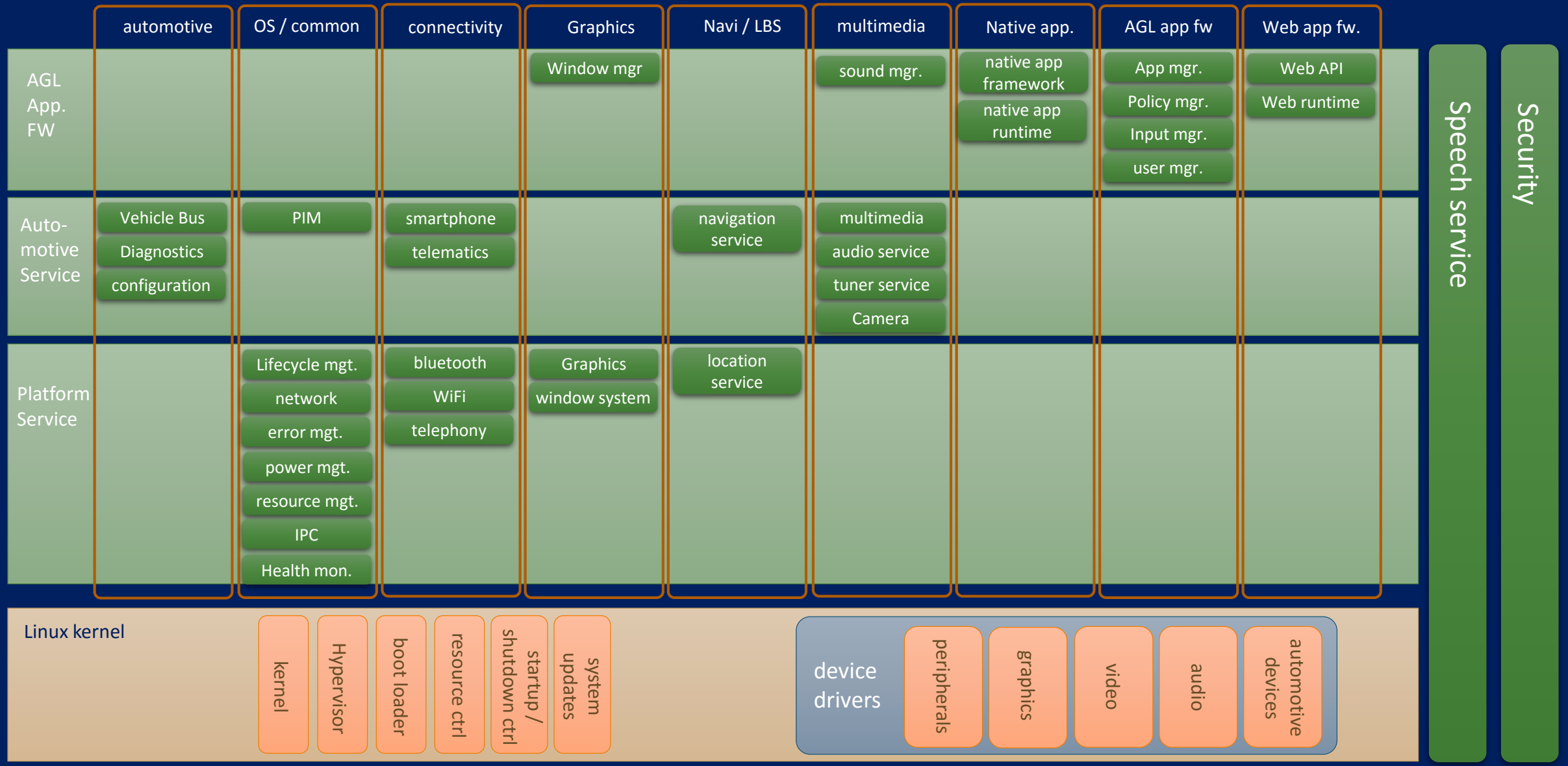
Linux becomes de-facto OS of connected Car



Why OSS becomes common choice?



Construction of Linux based software (AGL example)



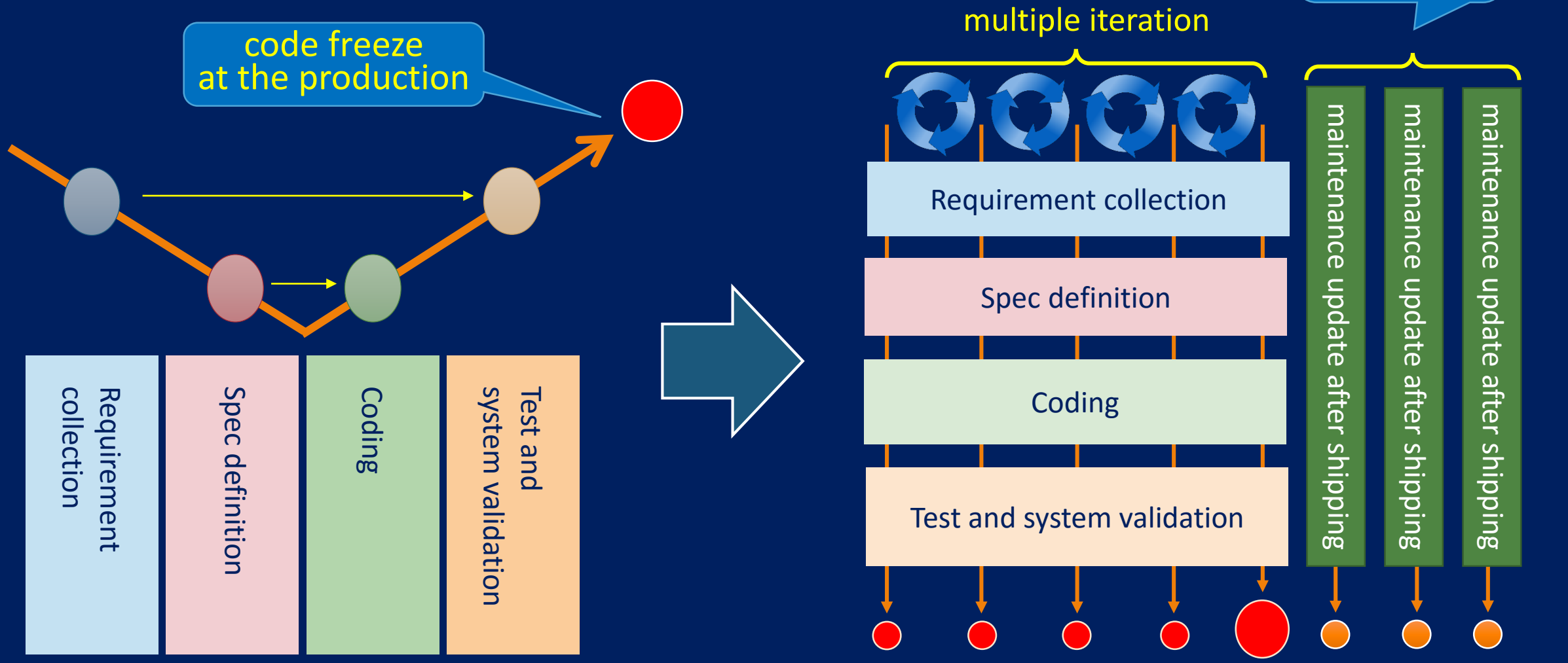
https://wiki.automotivelinux.org/agl-distro

RTOS vs. Linux : completely different philosophy

Traditional RTOS programing manner does not work for Linux environment.

RTOS (monitor program)	comparison items	Linux
sequential execution	task execution order	runtime auto coordination
fixed by coding (deterministic & predictable)	scheduling policy	smart & fair CPU allocation (heuristics)
static allocation (pre-assigned)	resource allocation	dynamic (runtime allocation)
absolute address	address description	relocation address
No	address translation	Yes (by MMU)
Yes	realtime management	No (various cache mechanism)
No	multi processor	Yes (SMP, AMP, HMP)
No	virtualization	Yes
Yes	FUSA support	No

Evolution of software development model



Agenda

- Software structure of IVI (In Vehicle Information) system
  - IVI software trend and architecture
  - RTOS vs. Linux philosophy comparison
  - Challenge of connected car software
- The fact of Linux kernel
  - Linux kernel summary
  - development community
  - Transparency and openness of OSS development process
- OSS adoption challenge for IVI system
  - Code quality
  - Security risk management

Fact : Linux kernel version 4.6 (released 2016.5.15)

- Linux kernel is huge, the work result of 25 years continuous development effort.
- No single company can compete with this development power (speed and quality) .

item	data
SLOC (true Source Lines Of Code)	14,398,866 line
v4.5 to v4.6 SLOC delta	279,351 line
v4.5 to v4.6 update period	73 days
Average update cycle	around 70 days
Source code file number	42,236 files
merged commits	13,354 pcs
Contributed developers count	1,661 mans
Value estimation (by sloccount)	$627,894,993 (≒ 690 oku-yen)

kernel 4.6 release info : https://lwn.net/Articles/686393/
kernel 4.6 feature intro : http://kernelnewbies.org/Linux_4.6

www.kernel.org : heart of Linux kernel development

Protocol	Location
HTTP	https://www.kernel.org/pub/
GIT	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/

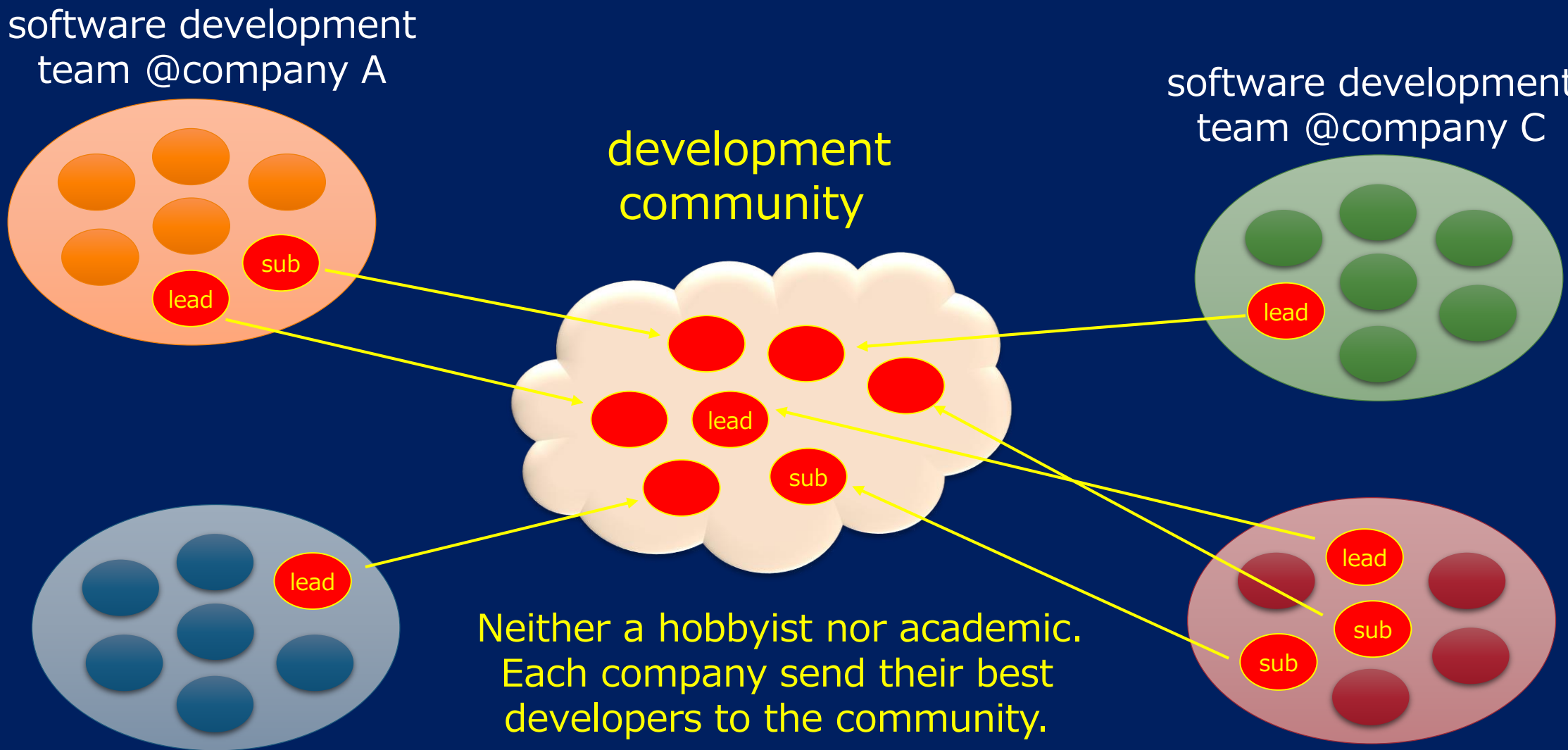
① Latest Stable Kernel: 4.6

② ③ ④

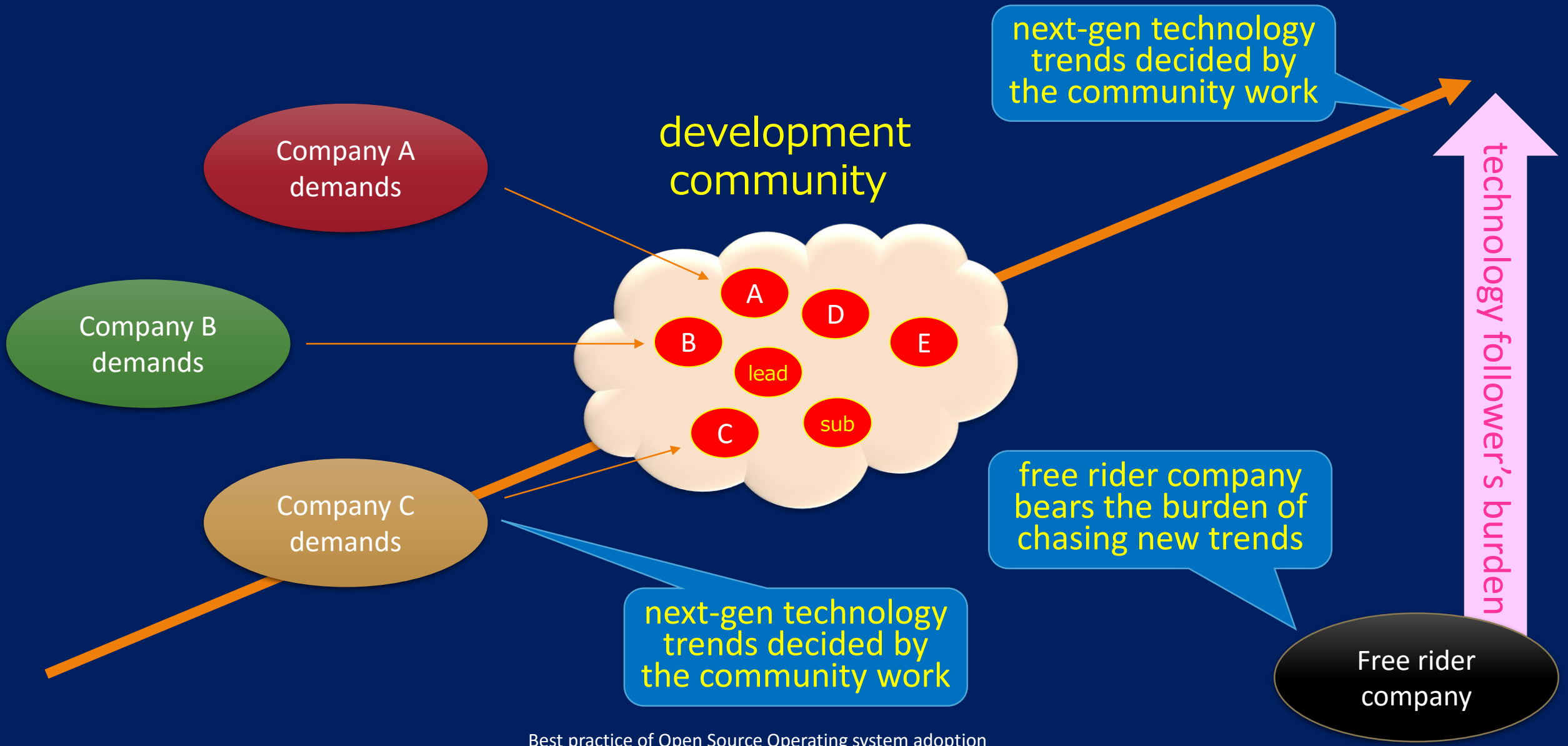
mainline:	4.6	2016-05-15	[tar.xz] [pgp] [patch]	[view diff] [browse]
stable:	4.5.5	2016-05-19	[tar.xz] [pgp] [patch] [inc. patch]	
longterm:	4.4.11	2016-05-19	[tar.xz] [pgp] [patch] [inc. pat	
longterm:	4.1.24	2016-05-11	[tar.xz] [pgp] [patch] [inc. pat	
longterm:	3.18.33	2016-05-11	[tar.xz] [pgp] [patch] [inc. pat	
longterm:	3.16.35	2016-04-30	[tar.xz] [pgp] [patch]	
longterm:	3.14.70	2016-05-19	[tar.xz] [pgp] [patch] [inc. pat	
longterm:	3.12.59	2016-04-27	[tar.xz] [pgp] [patch] [inc. pat	
longterm:	3.10.101	2016-03-16	[tar.xz] [pgp] [patch] [inc. pat	
longterm:	3.4.112	2016-04-27	[pgp] [patch] [inc. pat	
longterm:	3.2.80	2016-04-30	[pgp] [patch] [inc. pat	
linux-next:	next-20160520	2016-05-20		[browse]

① : Latest release
② : under development
③ : long-term
④ : queued for the next

Community gathers best of best talented developer



Serious battle to get next-gen technology standards



Layered structure of maintainer : trusted link

Linus Torvalds

pull request

Review and merge request

sub-system maintainer

sub-system maintainer

sub-system maintainer

about 1,800 maintainers

driver / file maintainer

driver / file maintainer

driver / file maintainer

driver / file maintainer

user / developer

user / developer

user / developer

user / developer

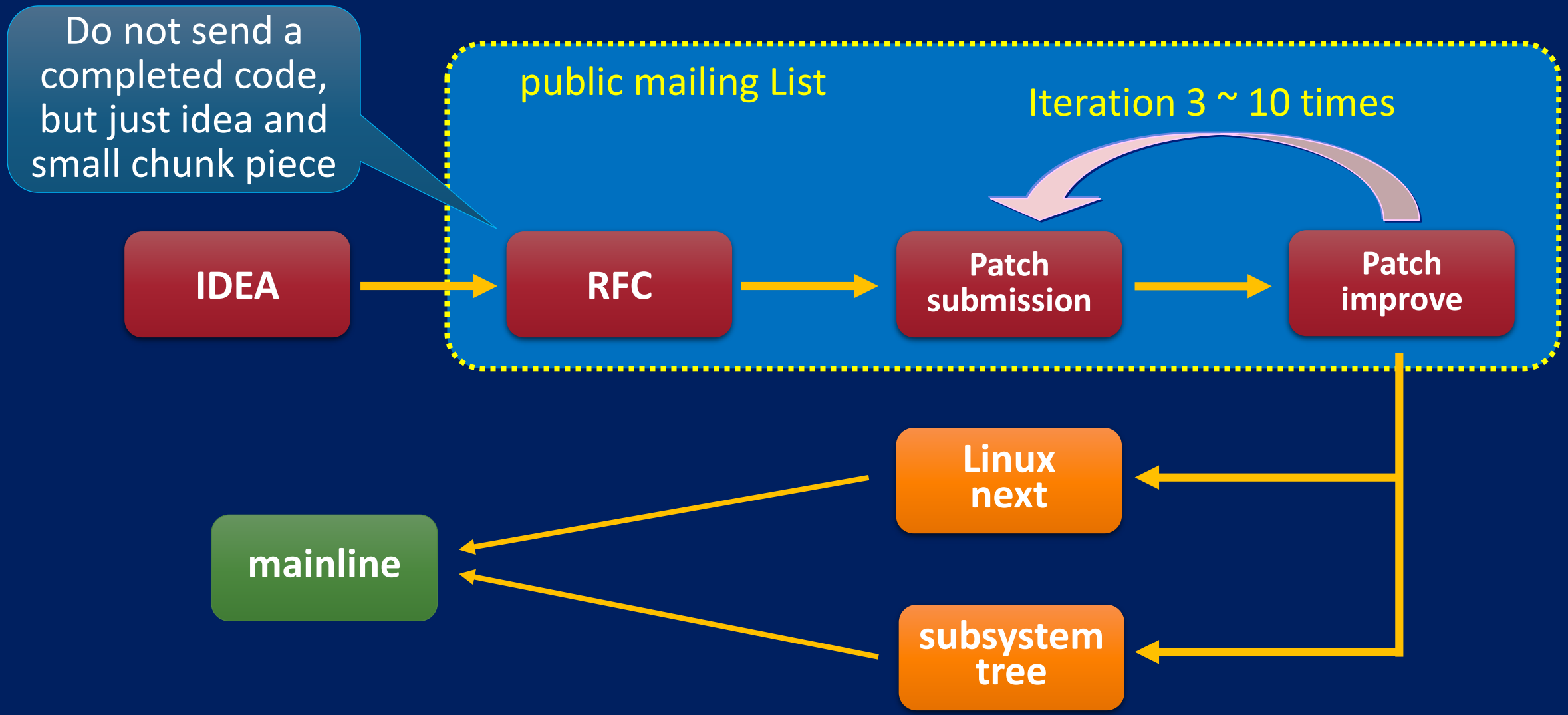
user / developer

git : advanced code management system (= CMS)

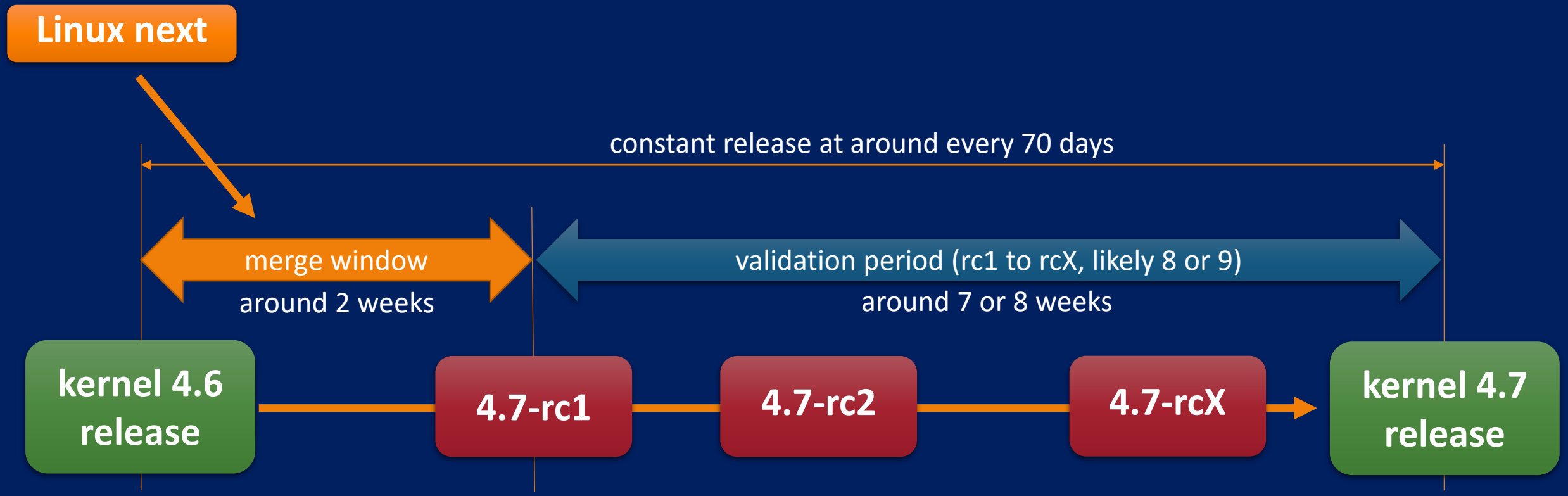
Linux kernel consist of small code chunk named “patch” submitted by the developers.

git support distributed code development and automated code integration & review mechanism. Each developer can send small code chunk named “patch” to create Linux

“public patch review” is the key success procedure



Constant kernel version migration process



- Patch merge period starts right after new version kernel release, and last 2 weeks.
- After merge windows close, multiple validation release (rc1 to rc8/9) continues
- Linus keeps same release interval of around 65 days.

Agenda

- Software structure of IVI (In Vehicle Information) system
  - IVI software trend and architecture
  - RTOS vs. Linux philosophy comparison
  - Challenge of connected car software
- The fact of Linux kernel
  - Linux kernel summary
  - development community
  - Transparency and openness of OSS development process
- OSS adoption challenge for IVI system
  - Code quality
  - Security risk management

Software vulnerability (CVE list) risk

CVE LIST | COMPATIBILITY | NEWS — MAY 18, 2016 | SEARCH

Common Vulnerabilities and Exposures

The Standard for Information Security Vulnerability Names

HOME > CVE LIST

About CVE
FAQs
CVE List
Search & Downloads
Updates & Feeds
Coverage Goals
Request a CVE-ID
CVE Numbering Authorities (CNAs)
CVE In Use
Scoring (via NVD)
Fix Info (via NVD)
CVE-Compatible Products
News
Free Newsletter
Community
CVE Editorial Board
Board Discussion Archives
Search the Site
Site Map

CVE List Main Page

CVE® is a publicly available and free to us
vulnerabilities and exposures.

IMPORTANT: CVE-IDs have a new numbe

National Vulnerability Dat

Full database functionality for the CV
provided through MITRE's partnership
the U.S. National Vulnerability Databa
(NVD).

- Data feeds of NVD's CVE content
- Scoring for CVE-IDs
- Fix information for CVE-IDs
- Statistics for NVD's CVE content
- Advanced searching of NVD's CVE

You may download the CVE List, copy it,
modify CVE itself as per our Terms of U
Cybersecurity and Communications at th

Search Results

There are **1444** CVE entries that match your search.

Name	Description
CVE-2016-4951	The tipc_nl_publ_dump function in net/tipc/socket.c in the Linux kernel through 4.6 does not verify socket existence, which allows local users to cause a denial of service (NULL pointer dereference and system crash) or possibly have unspecified other impact via a dumpit operation.
CVE-2016-4913	The get_rock_ridge_filename function in fs/isofs/rock.c in the Linux kernel before 4.5.5 mishandles NM (aka alternate name) entries containing \0 characters, which allows local users to obtain sensitive information from kernel memory or possibly have unspecified other impact via a crafted isofs filesystem.
CVE-2016-4805	Use-after-free vulnerability in drivers/net/ppp/ppp_generic.c in the Linux kernel before 4.5.2 allows local users to cause a denial of service (memory corruption and system crash, or spinlock) or possibly have unspecified other impact by removing a network namespace, related to the ppp_register_net_channel and ppp_unregister_channel functions.
CVE-2016-4794	Use-after-free vulnerability in mm/percpu.c in the Linux kernel through 4.6 allows local users to cause a denial of service (BUG) or possibly have unspecified other impact via crafted use of the mmap and bpf system calls.
CVE-2016-4581	fs/pnode.c in the Linux kernel before 4.5.4 does not properly traverse a mount propagation tree in a certain case involving a slave mount, which allows local users to cause a denial of service (NULL pointer dereference and OOPS) via a crafted series of mount system calls.
CVE-2016-4580	The x25_negotiate_facilities function in net/x25/x25_facilities.c in the Linux kernel before 4.5.5 does not properly initialize a certain data structure, which allows attackers to obtain sensitive information from kernel stack memory via an X.25 Call Request.
CVE-2016-4578	sound/core/timer.c in the Linux kernel through 4.6 does not initialize certain r1 data structures,

https://cve.mitre.org/cve/index.html

Longterm kernel contains many CVE fix patces

Selected kernel version named “longterm-stable (LTS)” kept maintained with a collection of security fixes.

Longterm release kernels

Version	Maintainer	Released	Projected EOL
4.4	Greg Kroah-Hartman	2016-01-10	Feb, 2018
4.1	Sasha Levin	2015-06-21	Sep, 2017
3.18	Sasha Levin	2014-12-07	Jan, 2017
3.16	Ben Hutchings	2014-08-03	Apr, 2020
3.14	Greg Kroah-Hartman	2014-03-30	Aug, 2016
3.12	Jiri Slaby	2013-11-03	Jan, 2017
3.10	Willy Tarreau	2013-06-30	Oct, 2017
3.4	Li Zefan	2012-05-20	Sep, 2016
3.2	Ben Hutchings	2012-01-04	May, 2018

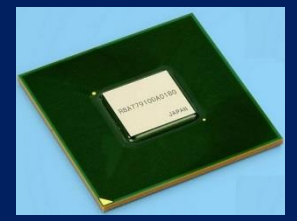
https://kernel.org/category/releases.html

LTS	kernel	release	latest	patch
	3.9	04.28.13	3.9.11	746
*	3.10	06.30.13	3.10.101	5,388
	3.11	09.03.13	3.11.10	677
(*)	3.12	11.15.13	3.12.60	6,412
	3.13	01.21.14	3.13.11	903
*	3.14	03.30.14	3.14.70	4,696
	3.15	06.08.14	3.15.10	703
	4.0	04.12.15	4.0.9	757
*	4.1	06.21.15	4.1.25	2,499
	4.2	09.01.15	4.2.8	903
	4.3	11.01.15	4.3.6	618
*	4.4	01.10.16	4.4.11	1,145

aiming 2017 LTS version kernel (4.7?)

Gen2 BSP

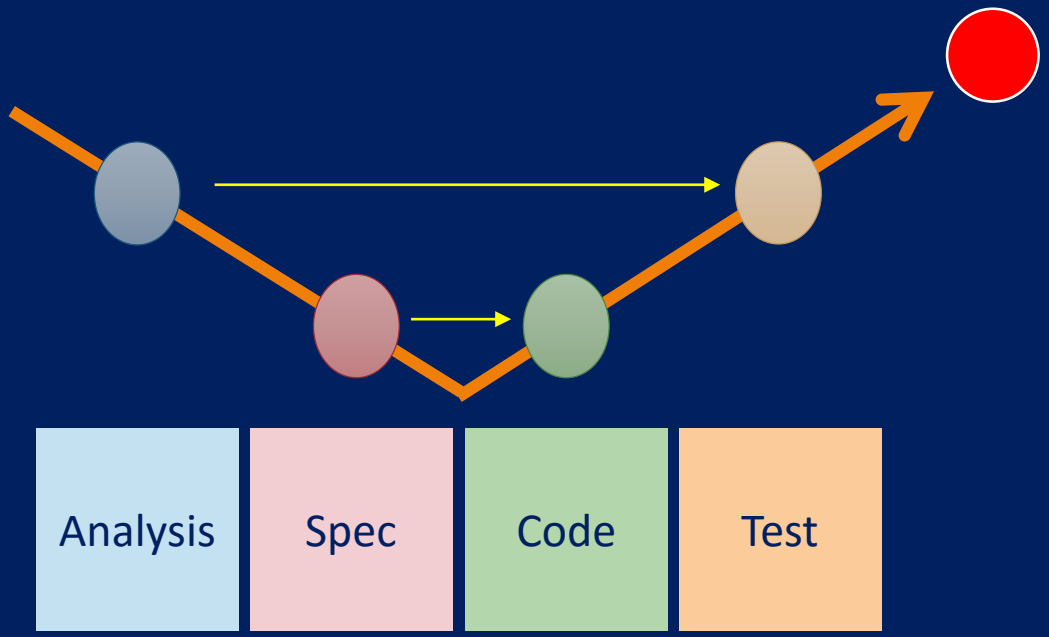
gen2 device

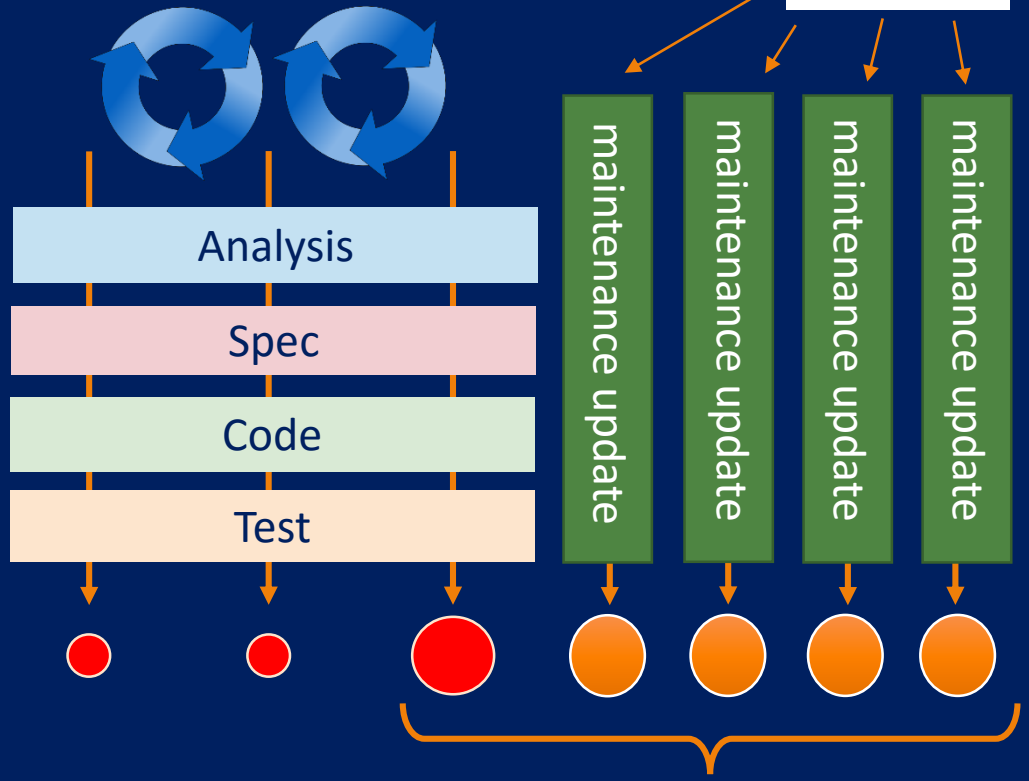


gen3 device

Gen3 BSP

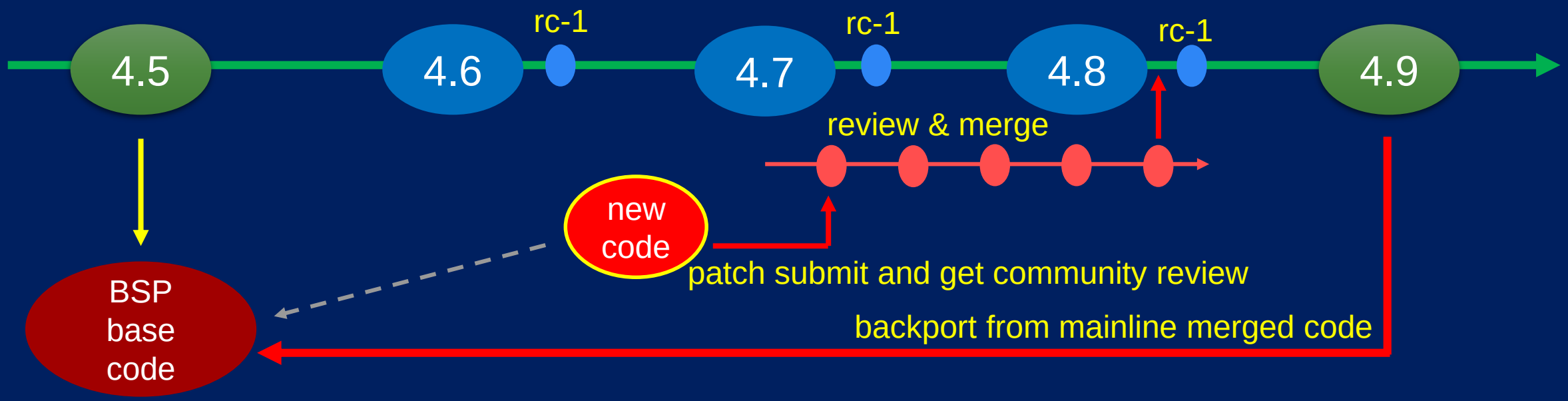
Connected era's quality : changing the scheme





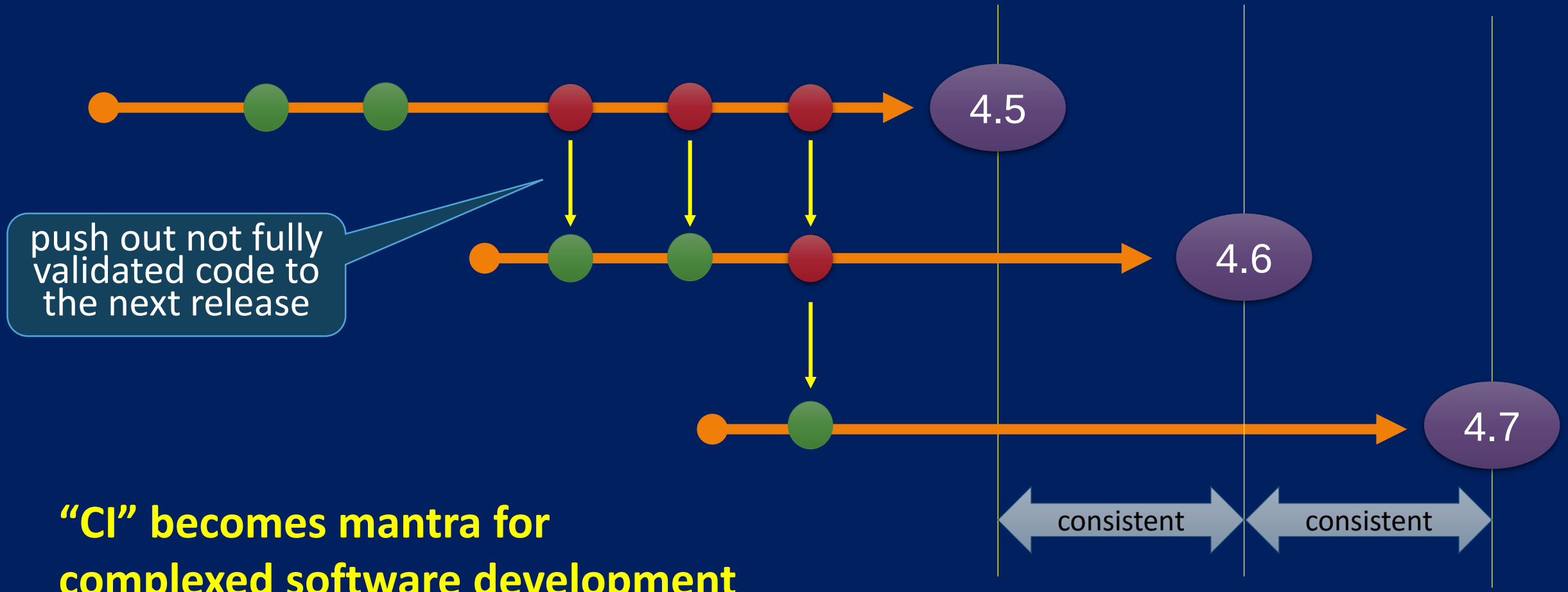
$$\text{Quality} = \frac{\text{Validation}}{\text{Requirement}} + \text{Quality} = \text{maintenance management}$$

Upstream first strategy



pros.	cons.
Can develop clean code by community review	It takes time (up to 6 month to merge)
Merged to the mainline, no more in-house code	Can not hide your secret (=differentiator)
Can apply upstream security fix patch	You may need to rewrite the patch by review

CI : Continuous Integration and many release



“CI” becomes mantra for complexed software development

Conclusion

- Linux and other OSS adoption become dominant for connected car IVI systems. (network, file system, security, graphics)

- Traditional RTOS development model does not work for Linux. You need to learn how OSS community develops the code. And write the appropriate code that complies with Linux philosophy.

- IVI system needs to have a field software update feature to apply CVE fix patches come from the community. If you heavily modify the code, it may cause security patch adoption troubles.