

# CLOSE ENCOUNTERS OF THE UPSTREAM RESOURCE

## 商品開発とオープンソース活動のバランスのとり方

ルネサスエレクトロニクス株式会社

オートモーティブソリューション事業本部 カスタマーエンゲージメント統括部

シニアダイレクター 宗像尚郎

# 自己紹介

2

- ルネサス（=シリコンサプライヤ）で働いています
  - 15年以上 組み込み Linux 業界 にいます
  - セットメーカー向けに Linux BSP（無償、故に無保証）を提供してきました
  - （最近は減りましたが）Linux 経験値の少ない お客様の導入支援 なども行ってきました
- 一方で、10年以上にわたって Linux 開発コミュニティ と関わってきました
- Linux Foundation ボードメンバ、AGL ボードメンバー、yocto ボードメンバー などなど
- ルネサス社内で（上記の Linux BSP 開発チームとは別に）Linux 要素技術開発チーム を運営し 開発コミュニティ連携を図ってきました。このチームにはコミュニティで著名なデベロッパーも含まれます

*We are learning how to work with the Linux & OSS community*

# 私の発見

# “製品開発エンジニア” vs. “コミュニティ デベロッパー”

4

開発コミュニティ, 長期保守, コードの再利用性,  
開発デッドライン (納期) は定められていない

“ユートピア志向”

upstream  
community

“赤い人”  
製品開発エンジニア

“青い人”  
コミュニティ・デベロッパー



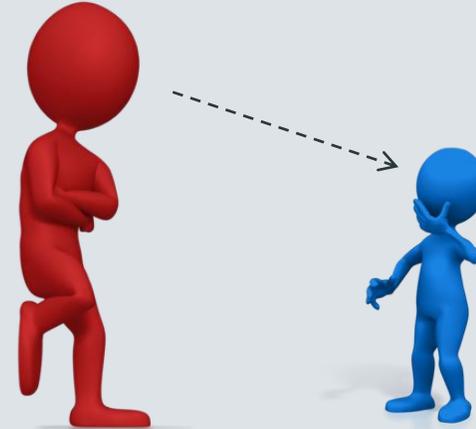
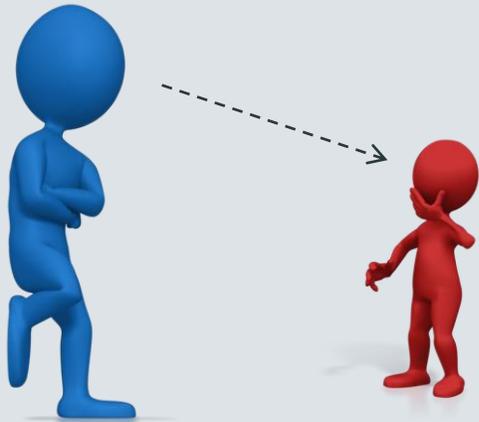
“ビジネス縛り”

予算、量産の為の開発、市場での競争



# 通常、お互い良く思っていないかも..... 目標とするものが違うから

5



- 目先の問題解決だけのための、**安易**な考え方
- **将来**そのコードが**再利用**される事を全く考慮していない
- オープンソース開発への**貢献**なんて全く考えていない
- ソフトウェアの機能要件やバグ情報などが共有されない
- 要求内容に一貫性が無く、**ランダム**に色々言うてる

- そもそも、開発に**時間**がかかりすぎ
- **品質**確保のための活動を行っているように見えない
- 商品の**差異化**（競争優位性）が作れない
- **今日**困っている**問題解決**に何も貢献してくれない
- オープンソースの**サポート**体制が確立されていない

*A red and blue party seems to aim a different goal, and it seems it is very hard to share it. However, it is obvious that red party's (future) work depends on blue party anyway.*

# クローズドソース時代の 製品開発エンジニア の 開発スキーム



認定ディストリビューター



バイナリーコードを  
有償でライセンス



基本は  
契約に基づく  
有償サポート

*A red engineer has no chance to access the original source code. Alternatively, they could ask paid support to get workaround, but it was not a true solution.*

# コンシューマー機器開発に Linux が導入された頃のサポートスキーム

7



コミュニティのソースコードを引用



バージョン  
ギャップ

製品開発エンジニアはコミュニティ  
デベロッパーとはつながっていなかった



embedded Linux  
distribution

有償で BSP として  
ソースコードを入手

*Industry developers were mainly supported by embedded Linux distribution and/or integrator company. Due to kernel version gap, there were limited connection to the upstream community.*

# 最近、開発コミュニティは製品開発トレンドとつながっているのだが...

8



*Industry developers have started to adopt relatively new kernel that have been integrated in software platform. However they are not familiar with working with the upstream community.*

# OSS ベース商品開発現場の“あるある”

# kernel を改造して似て非なる独自 OS を作ってしまう

10



upstream  
community

もはや原型をとどめていないので  
**X** ←  
メインラインコードには戻せない



**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	<a href="http://www.kernel.org/pub/">http://www.kernel.org/pub/</a>
FTP	<a href="ftp://ftp.kernel.org/pub/">ftp://ftp.kernel.org/pub/</a>
RSYNC	<a href="rsync://rsync.kernel.org/pub/">rsync://rsync.kernel.org/pub/</a>

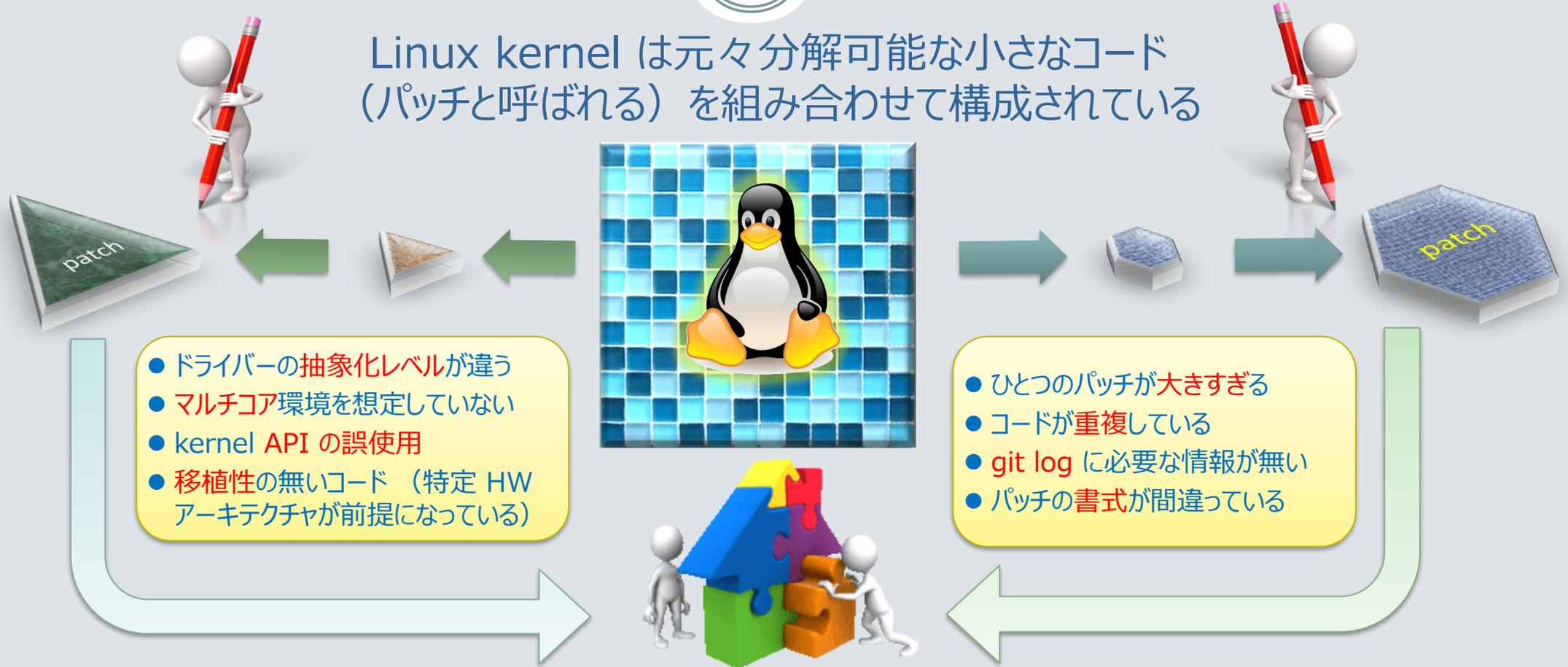
Latest Stable Kernel:  
 3.2.2

確かにライセンス文書には「コード改変可能」と書かれているのだが、しかし …

# OS コードの記述ルールを知らない、正しいパッチが書けない



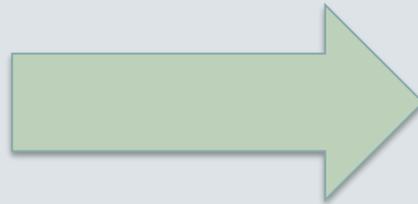
Linux kernel は元々分解可能な小さなコード (パッチと呼ばれる) を組み合わせて構成されている



*Locally developed large & irregular patch can not be integrated into the mainline kernel*

# 最新の OS のデバッグメッセージを読み取ることが出来ない

12



毎晩深夜まで解析作業を  
やっているだが .....



```
Command Console
# kernel panic !!
# crush dump XXXX
# segmentation fault
```

*When they faced a Linux debug message, they can hardly understand what does it meant*

# 有償緊急サポートを要請をしたのに、全く要領を得ず金だけとられた

13

戦略なきログ収集（ただ集めるだけ）



本質的ではない的外れなアドバイス

# コミュニティで既にある対策コードを調査せず、独自の修正を適用

14



コミュニティの新しいバージョンのカーネルには検討中のバグの対策コードが含まれているのに



独自の対策パッチを  
開発して適用してしまう

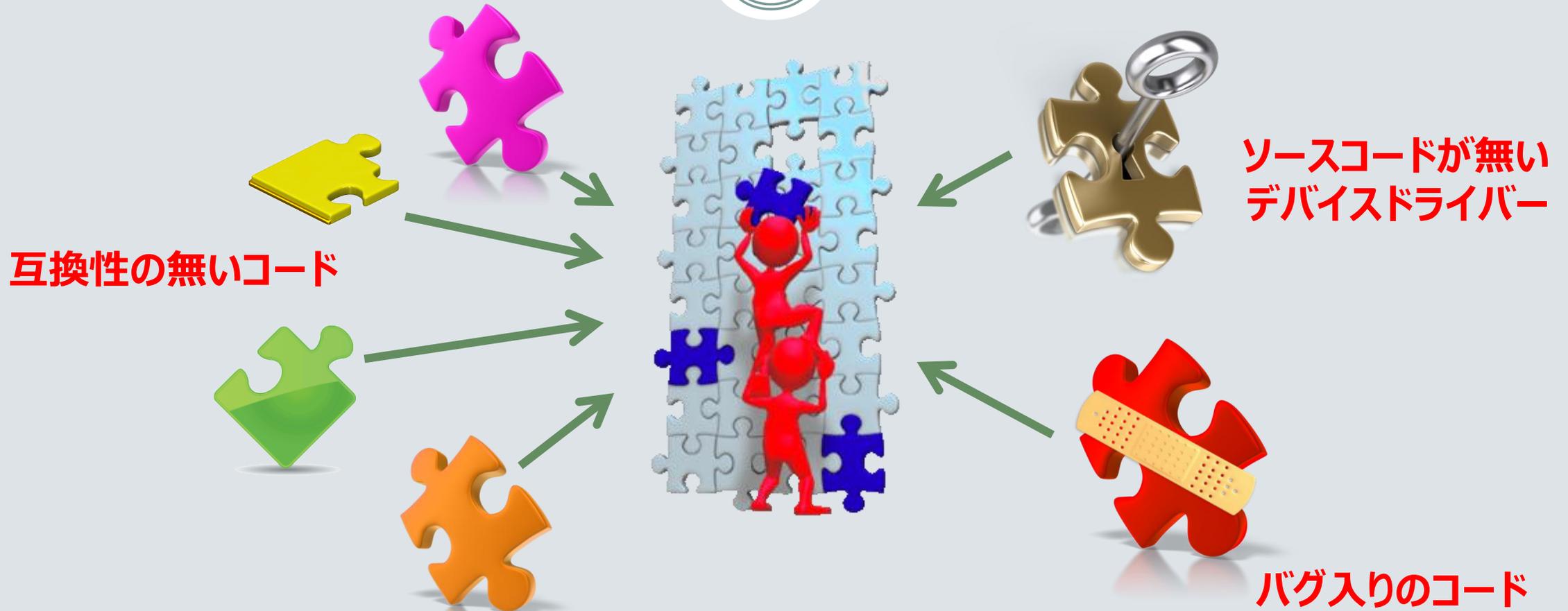


upstream  
community



# 必要なレビュープロセスを通さずに、何でもマージしてしまう

15



*They really do not understand the role of tree maintainer who need to block dirty piece*

## 自家製カーネルのメンテに行き詰まり、不良債権化

16



ローカルパッチの塊りと化し、Upstreamとは大きく乖離してしまった、“保守不可能”な不良資産で身動きが取れない

*It costs enormous (unrealistic) cost for maintaining in-house private kernel tree*

# 製品開発エンジニアのコミュニティとの付き合い方

できるだけ最新のカーネルを使うことで、コミュニティとの距離を縮めることができる

18



*Upstream community developer sit much closer to product development engineer now*

# カーネルデベロッパーでなくても、活用できるネタは沢山ある

19

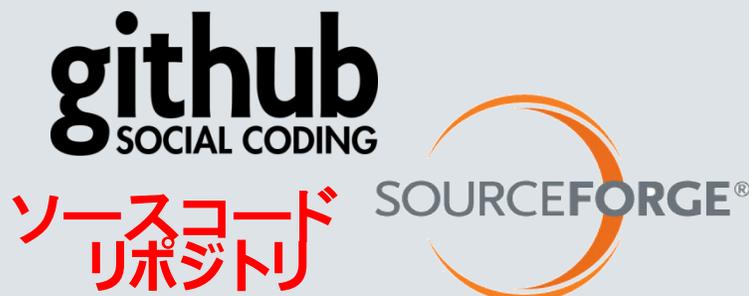
- 製品開発現場が **毎日がとても忙しい** のはわかっています。以下は無理ですね。
  - あなたの課題に関連したコミュニティ開発者の **メーリングリストに毎日目を通す**
  - コミュニティが開発中のカーネルバージョン向けに **パッチを作成して投稿** する
  - **メーリングリスト上の議論に参加** して、投稿したパッチをマージしてもらえらるまで持っていく
- それでも コミュニティが今おこなっている **git、バグトラッキング情報（BTS）、メーリングリスト上の議論** などの活動に少しでも注目してもらえればきっとあなたの **開発をもっと効率的、かつ健全** な形にすることができる可能性があるので。いまこそ、オープンソース開発の現場に飛び込んでみませんか？



# 開発コミュニティの活動を覗いてみよう

20

デベロッパーコミュニティ



メーリングリスト・アーカイブ



コンソーシアム活動



ソースコード・アーカイブ

*Even without direct contribution, red team developer can utilize various "open resources" !!*

# ともかく、徹底的に git を使いこなそう

21

git は非常に強力なコードマネージメントツールなので、git を正しくかつ徹底的に有効活用できれば、あなたの オープンソースの利用をずいぶんと助けてくれるでしょう。

- .git という通常は見えない隠しディレクトリの中に、ひとつひとつの パッチに関する付帯情報がたくさん保存できるようになっています。この 付帯情報を検索 することで解析（デバッグ）やメンテナンスの作業を大幅に効率化できます。
- あなたが git を使ってコード（パッチ）を登録する時には、後で参照できるような log 情報や適切な tag の登録 などを行ってください。コミュニティ開発に参加する時には、このような厳密な git の運用ルールを理解し従うことが求められます。
- ローカスバグ修正などの 実験コード を作る場合には、適切な ブランチを切ってマスターコードと分けてコードが管理できるような工夫も重要です。



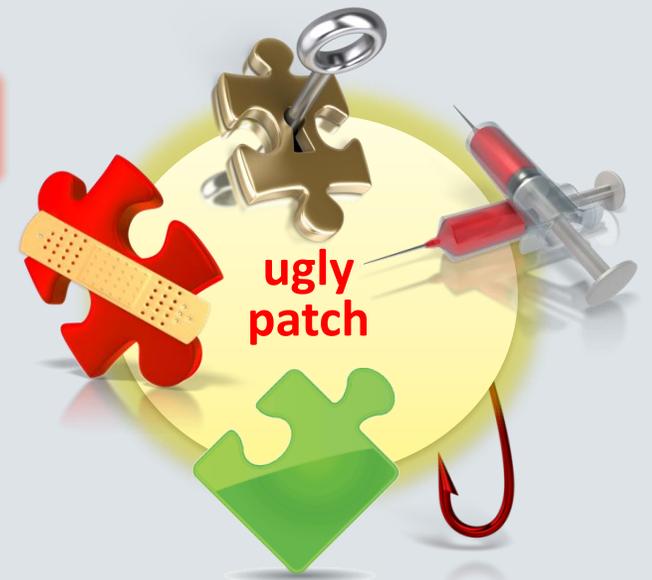
## 強い権限をもったゲートキーパー（＝インハウス メンテナー）

22

- マスターコードにパッチをマージする時には、例外なくメンテナーによるコードレビューを義務づける必要があります。メンテナーには **いかなるコードでもマージを拒否する権限** を持たせます。
- メンテナーはサーバー管理者とは別です。  
**新人の仕事ではありません。**  
SW に関する深い知見と権限をもったシニアプログラマーの役割です。



社内コードメンテナー

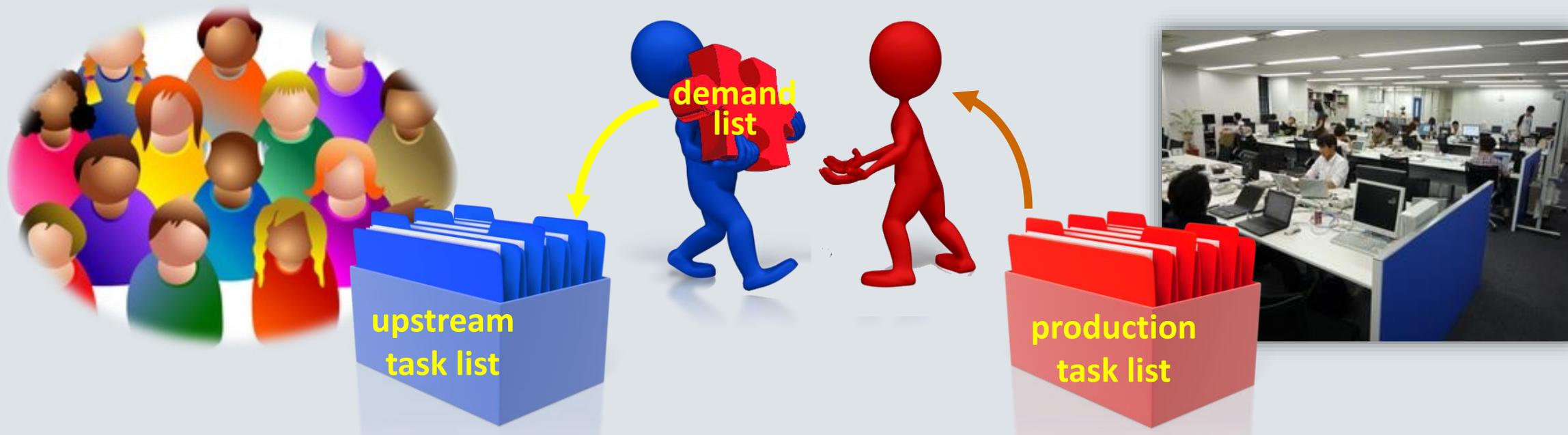


***Maintainer should retain the authority to reject any vulnerable nor ugly shape code merge.***

## 製品開発エンジニアとコミュニティデベロッパーのコラボレーション

23

- 製品開発エンジニアとコミュニティデベロッパーは 別々の課題リストを持っているのですが 意図的に製品開発現場の課題を定期的にコミュニティデベロッパーと共有する機会を設けるのはとても重要（有効。今日解がもらえなくても、中長期の根本解決につながっていきます。



*Community developers eager to know what production engineer are struggling now*

## 製品開発チームの BTS (バグ管理情報) を公開しよう

24

- コミュニティ開発は、関係者との調整に時間を要するので時間のかかるプロセスです
- 一方製品開発現場では、待たなしの問題解決が求められるのが普通ですね
- なので **コミュニティ開発者は 商品開発現場のバグ管理票を解決してはくれない** でしょう

しかし、商品開発現場での課題解決を オープンソースのマスターコードに反映できなければ、問題が **確実に将来再発** します。 開発 現場の課題と対策をコミュニティデベロッパーと共有していく事がとても大事なのです。



# ベストプラクティス ? (事例紹介)

# 社内 コミュニティブリッジ チーム

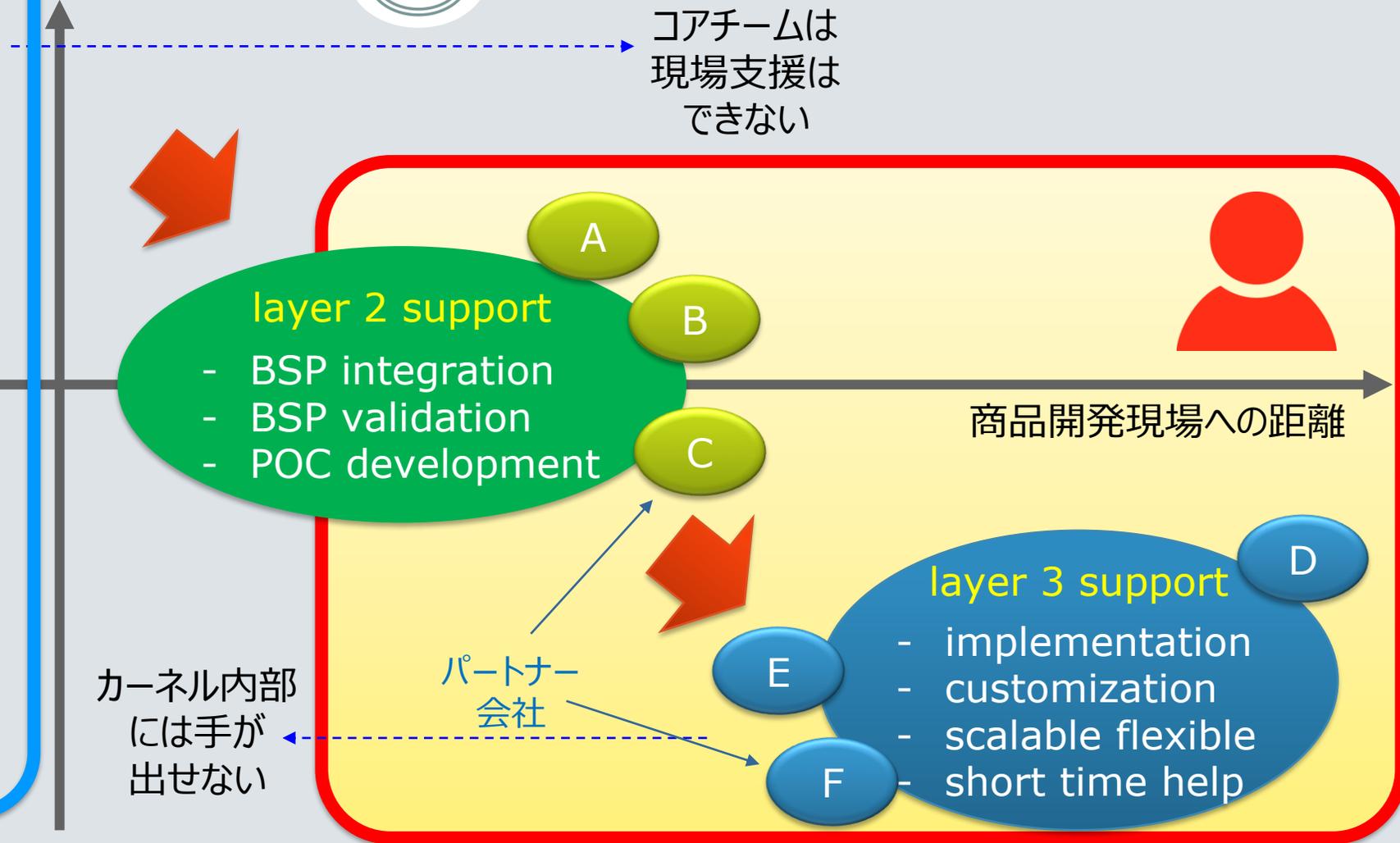
26

layer 1 support

- code upstream
- innovation
- long term vision



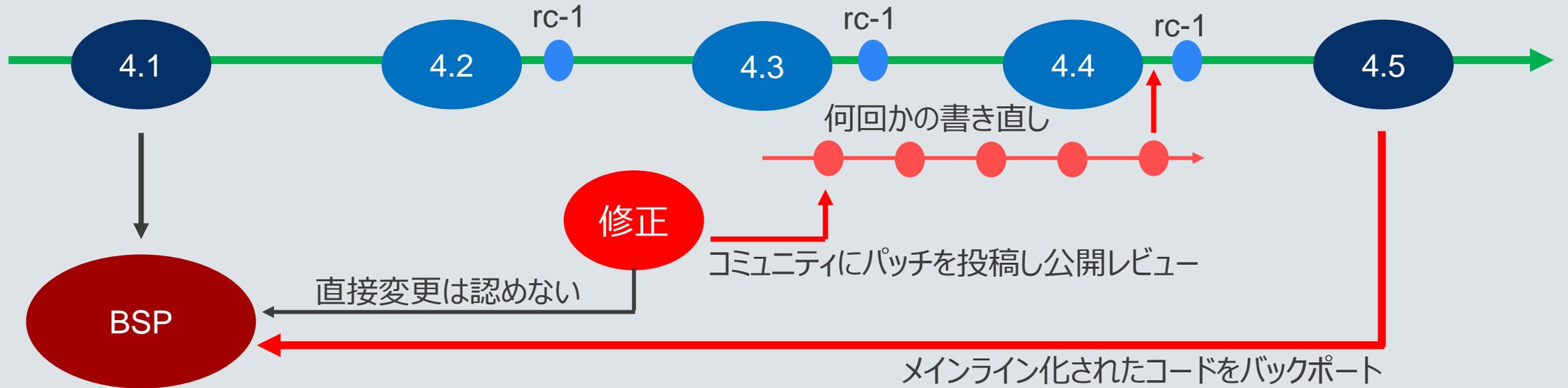
we have internal  
**core team**





# アップストリーム・ファースト

27



メリット	デメリット
公開コードレビューによってコードの品質が高まる	時間がかかる (イメージ的には半年程度の時間が必要)
変更はマスターコードに取り込まれる、ローカルパッチ不要	公開プロセスなので、秘密にしておく事はできない
社内コードの独自管理が不要になる (ローカルパッチ削減)	何回かのコード書き直しが必要になる



# コントリビューション(5.3)

実績

28

## Most active 5.3 employers

### By changesets

Intel	1545	10.7%
AMD	966	6.7%
(Unknown)	963	6.7%
Red Hat	925	6.4%
(None)	865	6.0%
Google	638	4.4%
Renesas Electronics	638	4.4%
Linaro	548	3.8%
IBM	460	3.2%
Mellanox	440	3.0%
(Consultant)	410	2.8%
Huawei Technologies	372	2.6%
SUSE	337	2.3%
Samsung	311	2.2%
NXP Semiconductors	309	2.1%
Linux Foundation	291	2.0%
ARM	280	1.9%
Facebook	188	1.3%
Oracle	170	1.2%
BayLibre	170	1.2%

### By lines changed

AMD	432537	37.0%
Linaro	115926	9.9%
Intel	76950	6.6%
Mellanox	46732	4.0%
Samsung	37869	3.2%
Google	31666	2.7%
(Unknown)	31310	2.7%
Red Hat	31089	2.7%
IBM	29449	2.5%
(None)	27803	2.4%
SUSE	18928	1.6%
Cirrus Logic	16509	1.4%
(Consultant)	14192	1.2%
Facebook	13723	1.2%
Linux Foundation	13405	1.1%
ARM	12755	1.1%
Pengutronix	12398	1.1%
NXP Semiconductors	12202	1.0%
Renesas Electronics	11345	1.0%
Huawei Technologies	10165	0.9%

## Most active 5.3 developers

### By changesets

😊 Kuninori Morimoto	271	1.9%
Christoph Hellwig	262	1.8%
Mauro Carvalho Chehab	205	1.4%
Nishka Dasgupta	162	1.1%
Chris Wilson	160	1.1%
Greg Kroah-Hartman	145	1.0%
Yue Haibing	137	0.9%
Masahiro Yamada	130	0.9%
Gustavo A. R. Silva	128	0.9%
😊 Geert Uytterhoeven	120	0.8%
Takashi Sakamoto	113	0.8%
😊 Wolfram Sang	101	0.7%
Arnaldo Carvalho de Melo	99	0.7%
Colin Ian King	98	0.7%
Arnd Bergmann	96	0.7%
Adrian Hunter	96	0.7%
Russell King	94	0.7%
David Howells	92	0.6%
Hawking Zhang	90	0.6%
Maxime Ripard	84	0.6%

### By changed lines

Hawking Zhang	364371	31.2%
Arnd Bergmann	84993	7.3%
Mauro Carvalho Chehab	34041	2.9%
Harry Wentland	33568	2.9%
Jason Gunthorpe	22421	1.9%
Chris Wilson	17382	1.5%
Richard Fitzgerald	15791	1.4%
Linus Walleij	13392	1.1%
Greg Kroah-Hartman	11713	1.0%
Bernard Metzler	10963	0.9%
Christoph Hellwig	10844	0.9%
Pawel Laszczak	7702	0.7%
Hannes Reinecke	7255	0.6%
Andrii Nakryiko	7143	0.6%
Maxime Ripard	6709	0.6%
David Ahern	6557	0.6%
Daniele Ceraolo Spurio	5781	0.5%
Darrick J. Wong	5776	0.5%
😊 Kuninori Morimoto	4717	0.4%
Thomas Zimmermann	4706	0.4%

<https://lwn.net/Articles/798505/>



# 透明性 ( Upstream 經過共有 )

社内  
事例

29

<https://git.kernel.org/pub/scm/linux/kernel/git/geert/renesas-devel.git/>



## index : kernel/git/geert/renesas-devel.git

Renesas Kernel ARM SoC Development Tree

master

Geert Uytterhoeven

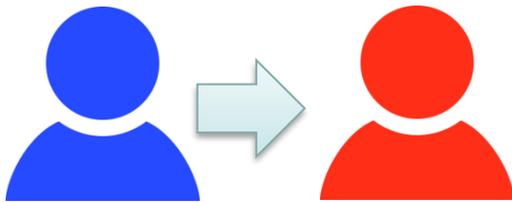
about **summary** refs log tree commit diff stats

log msg

Branch	Commit message	Author	Age
master	Merge branches 'renesas-arm-defconfig-for-v5.6' and 'renesas-arm-dt-for-v5.6'...	Geert Uytterhoeven	3 days
renesas-arm-defconfig-for-v5.6	ARM: shmobile: defconfig: Enable support for panels from EDT	Fabrizio Castro	3 days
renesas-arm-dt-for-v5.6	ARM: dts: iwg20d-q7-common: Add LCD support	Fabrizio Castro	3 days
next	Merge branches 'renesas-arm-defconfig-for-v5.5', 'renesas-arm64-defconfig-for-...	Geert Uytterhoeven	4 weeks
renesas-drivers-for-v5.5	soc: renesas: rcar-sysc: Add R8A77961 support	Geert Uytterhoeven	4 weeks
renesas-arm-defconfig-for-v5.5	ARM: shmobile: defconfig: Refresh for v5.4-rc1	Geert Uytterhoeven	4 weeks
topic/renesas-defconfig	arm64: renesas_defconfig: Enable R8A77961 SoC	Geert Uytterhoeven	4 weeks
renesas-arm64-defconfig-for-v5.5	arm64: defconfig: Enable R8A77961 SoC	Geert Uytterhoeven	4 weeks
renesas-arm64-dt-for-v5.5	arm64: dts: renesas: Add support for Salvator-XS with R-Car M3-W+	Geert Uytterhoeven	4 weeks

Age	Commit message	Author	Files	Lines
3 days	Merge branches 'renesas-arm-defconfig-for-v5.6' and 'renesas-arm-dt-for-v5.6'... <b>HEAD</b> <b>renesas-devel-2019-11-25-v5.4</b> <b>master</b>	Geert Uytterhoeven	3	-1/+88
3 days	Merge tag 'v5.4' into renesas-devel	Geert Uytterhoeven	94	-317/+695
3 days	ARM: shmobile: defconfig: Enable support for panels from EDT <b>renesas-arm-defconfig-for-v5.6</b>	Fabrizio Castro	1	-0/+3
3 days	ARM: dts: iwg20d-q7-common: Add LCD support <b>renesas-arm-dt-for-v5.6</b>	Fabrizio Castro	2	-1/+85
3 days	Linux 5.4 <b>v5.4</b>	Linus Torvalds	1	-1/+1
3 days	Merge branch 'fixes' of git://git.kernel.org/pub/scm/linux/kernel/git/viro/vfs	Linus Torvalds	1	-2/+2
4 days	cramfs: fix usage on non-MTD device	Maxime Bizon	1	-2/+2
4 days	Merge tag 'for_linus' of git://git.kernel.org/pub/scm/linux/kernel/git/mst/vhost	Linus Torvalds	3	-24/+28
5 days	Merge branch 'for_linus' of git://git.kernel.org/pub/scm/linux/kernel/git/dto...	Linus Torvalds	1	-1/+0
5 days	Revert "Input: synaptics - enable RMI mode for X1 Extreme 2nd Generation"	Lyude Paul	1	-1/+0



# 解説 (説明責任)

社内  
事例

30

Linux v5.4 がリリースされました。  
LWN の記事はこちら

Statistics from the 5.4 development cycle  
<https://lwn.net/Articles/804119/>

2019 年最後のリリースとなります  
また、Linux v5.4 は次期 LTS となり 2021/12 までサポートされる予定です

## ■ Renesas からの貢献は？

今回 TOP20 に入ったのは私だけでした。  
後は、Tested-by / Reported-by で Geert が TOP20 入りしています。

Renesas は 12 位でした。

## ■ Renesas Linux v5.4 の状況は？

v5.4 の Renesas Linux としては、

- I2C ダイナミックアドレス API 準備
- R-Car Gen3 SDHI/eMMC ホストで IOMMU を使う事でパフォーマンス向上
- R-Car VIN で、ARGB 出力可能. Alpha は V4L2 コントロールを使う事で調整可能
- Sound フレームワークのクリーンアップ

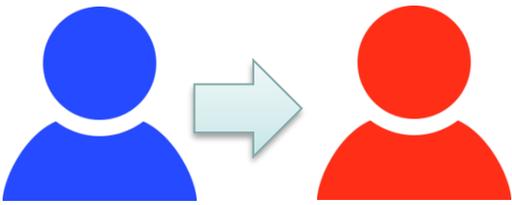
## ■ Linux v5.4 でのトピックは？

Linux v5.4 は次期 LTS となり 2021/12 までサポートされる予定です

また v5.4 には新たに「ロックダウン」セキュリティ機能が追加されたようです  
更に強力にセキュリティを向上するもので、root でさえも制限をうけるようになる  
ようですが、それ故にデフォルトでは OFF になっているようです。  
有効化するには CONFIG\_SECURITY\_LOCKDOWN\_LSM を選択する必要があります

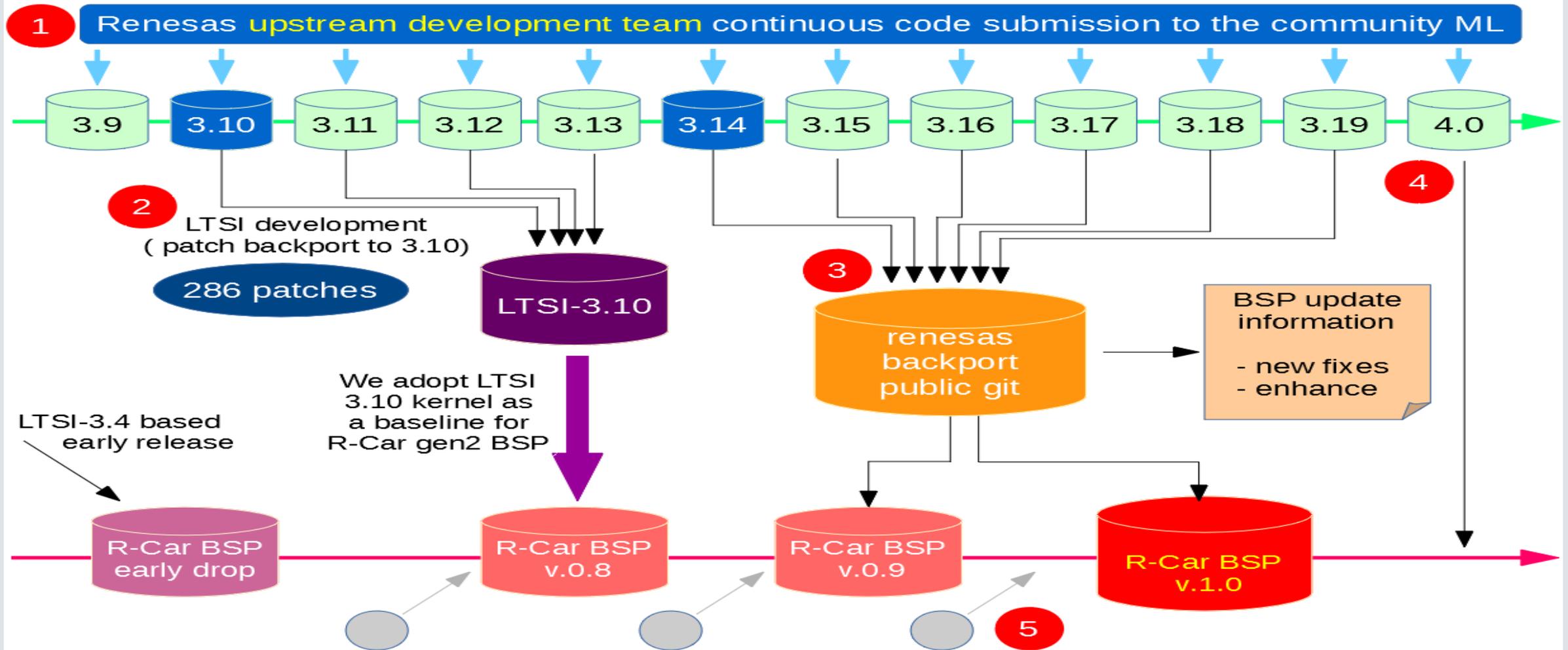
Linuxにカーネル「ロックダウン」セキュリティ機能追加へ  
<https://japan.zdnet.com/article/35143344/>

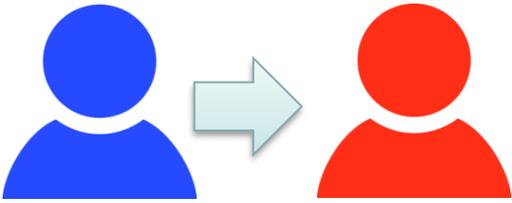
Linux 5.4-rc1がリリース, 議論となった「ロックダウン」機能も実装へ  
[http://gihyo.jp/admin/clip/01/linux\\_dt/201910/02](http://gihyo.jp/admin/clip/01/linux_dt/201910/02)



# バックポート (メインライン → BSP)

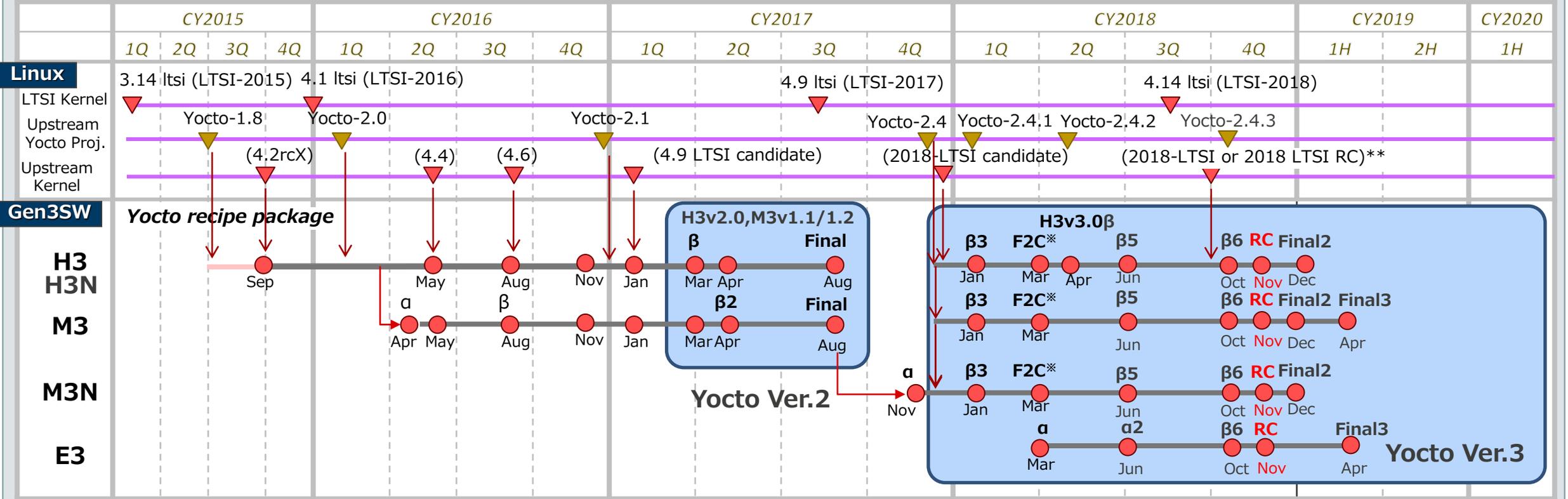
31





# バックポート (メインライン → BSP)

社内事例

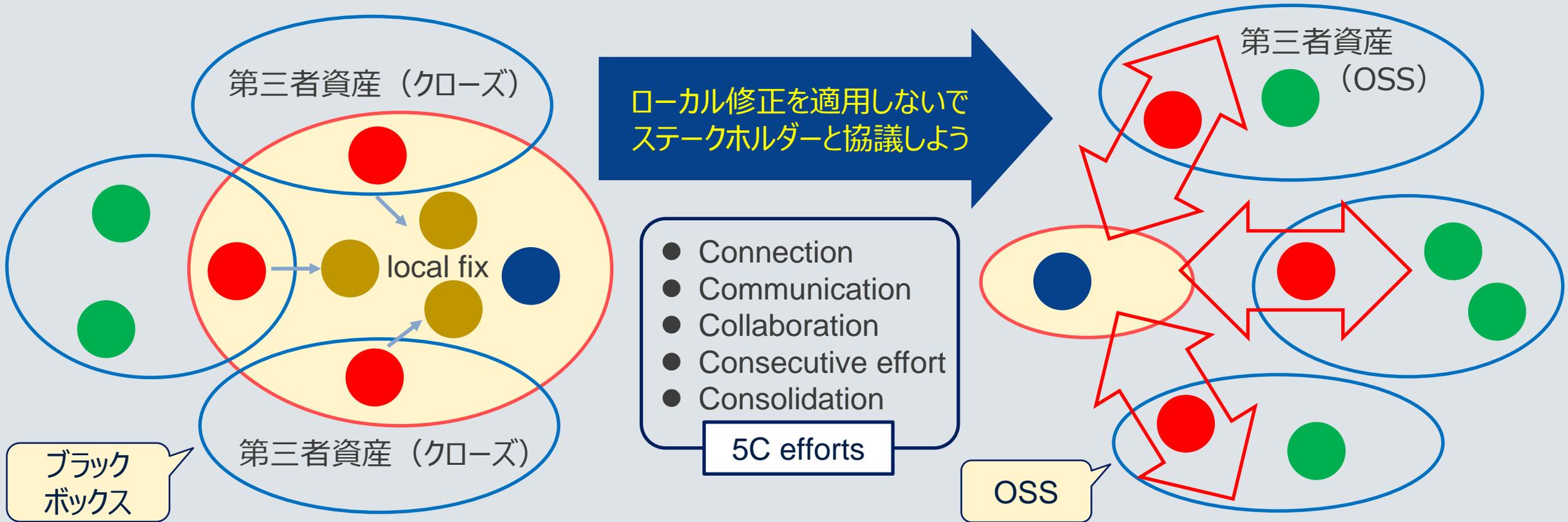


*BSP development process relies on upstream release ( LTS/LTSI and yocto schedule )*

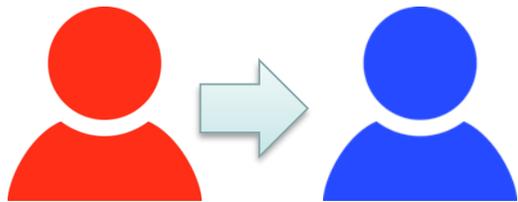


# ローカルな修正をしない

33



*You can access to the code author and request to include your workaround (if that is correct)*



# アップポート (ローカルな修正を戻す)

社内  
事例

34

## Folder

- [projects](#)
  - [linux](#)
    - [core](#)
    - [io](#)
      - [done](#)
      - [maybe](#)
    - [mm](#)
  - [uboot](#)

## Assignee

- [BSP](#)
- [Geert](#)
- [Jacopo](#)
- [Kaneko](#)
- [Kieran](#)
- [Laurent](#)
- [Magnus](#)
- [Marek](#)
- [Morimoto](#)
- [Niklas](#)
- [Shimoda](#)
- [Simon](#)
- [Ulrich](#)
- [Wolfram](#)
- [NoAssignee](#)

## Status

### Active

file	team	assignee	title
<a href="#">rcar_dmac_priority</a>	Core	Magnus	RCAR-DMAC; Priority handling
<a href="#">rcar_dmac_slow_mode</a>	Core	Magnus	RCAR-DMAC; Slow mode trial (for e.g. SCIF hardening)
<a href="#">bsp392_iommu_ipmmu-vmsa</a>	Core	Geert	From bsp392, upport iommu/ipmmu-vmsa:
<a href="#">clk_renesas_determine_rate</a>	Core	Geert	CLK; Implement .determine_rate() callback
<a href="#">bsp392_arm64_dts_renesas_r8a77990-ebisu</a>	Core	NoAssignee	From bsp392, upport arm64: dts: renesas: r8a77990-ebisu:
<a href="#">virt_gpio_paravirtualization</a>	Core	Geert	virt; Paravirtualized GPIO
<a href="#">bsp396_pinctrl_sh-pfc_r8a77990</a>	Core	Geert	From bsp396, upport pinctrl: sh-pfc: r8a77990:
<a href="#">bsp392_clk_renesas_z_z2_clocks</a>	Core	Simon	From bsp392, upport clk: renesas: Z/Z2 clocks

## From bsp392, upport iommu/ipmmu-vmsa: ←

file	status	team	assignee	key
<a href="#">projects/linux/core/bsp392_iommu_ipmmu-vmsa.yaml</a>	Active	Core	Geert	cd3dde54-194a-408f-bb50-01ed151edb56

BSP	upstream
<ul style="list-style-type: none"> <li>• <a href="#">iommu/ipmmu-vmsa: Fix IMCTR_RTSEL_MASK for R-Car Gen3 and Gen2</a></li> <li>• <a href="#">iommu/ipmmu-vmsa: Do backup/restore for all IPMMU context</a></li> <li>• <a href="#">iommu/ipmmu-vmsa: Add override support for the actual number of MMU contexts</a></li> <li>• <a href="#">iommu/ipmmu-vmsa: Invalidate TLB when IPMMU translation error occurred</a></li> <li>• <a href="#">iommu/ipmmu-vmsa: Handle reserved bits correctly</a></li> <li>• <a href="#">iommu/ipmmu-vmsa: Clean up unnecessary micro-TLB invalidation</a></li> <li>• <a href="#">iommu/ipmmu-vmsa: Do backup/restore for each IP master</a></li> </ul>	<ul style="list-style-type: none"> <li>torvalds <a href="#">iommu/ipmmu-vmsa: Link IOMMUs and devices in sysfs</a></li> <li>torvalds <a href="#">iommu/ipmmu-vmsa: Prepare to handle 40-bit error addresses</a></li> <li>torvalds <a href="#">iommu/ipmmu-vmsa: Make IPMMU_CTX_MAX unsigned</a></li> <li>torvalds <a href="#">iommu/ipmmu-vmsa: Move num_utlbs to SoC-specific features</a></li> <li>torvalds <a href="#">iommu/ipmmu-vmsa: Extract hardware context initialization</a></li> <li>torvalds <a href="#">iommu/ipmmu-vmsa: Add suspend/resume support</a></li> </ul>

経営者を説得するために

# ソフトウェア品質

36

- OSS の“品質”について色々な意見がある。どちらかといえば **リスクを煽る見解** が多い？
- 実際に開発を行っている立場としては、**過剰に期待させるような言い方**には躊躇 がある

しかし 過去10年間の経験と（公開できないが）実績から、今あえて言いたいのは

社内製、社外製を問わず（多くの場合有償の）**プロプライエタリーなソフトウェア**よりも、**オープンソース・ソフトウェア**の方が明らかに**品質が高い**という強い印象を持っています

（参考）同じような見解をもっている第三者レポート

『OSS(オープン・ソース・ソフトウェア)のこれからと、静的解析の効果』 by SYNOPSIS

<https://www.synopsys.com/ja-jp/software-integrity/resources/reports/open-source-software-scan-report.html>

# コスト

37

- OSS は無償だからコストが下がる… という短絡的な考え方は全く NG ですが…
- コミュニティ連携（スケーラビリティ、サステナビリティ）
- カスタマーサポート（集合知活用、SW品質）
- 再利用可能な SW 資産形成（トレーサビリティ、モジュラリティ）
- 中長期的なメンテナンス性（セキュリティ、プライバシー）

# テクノロジー・ディレクション

38

## Today the Linux Foundation is Much More than Linux...

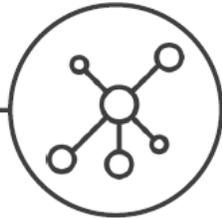
### Security



We securing the internet as home to world's largest certificate authority securing 100M web sites.



### Networking



We are home to 8 of the top 10 open source networking projects in the world backed by the majority of global network providers.



### Cloud



We are creating a portability layer for the cloud, driving standards and developing reference tools for cloud native development.



kubernetes



### Automotive



Our Automotive Grade Linux platform is backed by 12 automakers and is either in or slated for production in millions of vehicles worldwide.



HYPERLEDGER



We are creating a permanent, secure distributed ledger that makes it easier to create cost-efficient, decentralized business networks.



We are creating projects used in building the majority of embedded linux distributions and rationalizing edge computing.



We are providing the application development framework for next generation web, mobile, serverless, and IoT applications.



## まとめ

39

- 商品開発エンジニア と 開発コミュニティ の距離が以前より大幅に近く なっていることは、大変良い傾向と言えます。
- しかし、まだ商品開発エンジニアの多くは コミュニティのリソース（ソースコード、バグ管理情報、ML上での議論 等）を充分活用できていない かもしれません。
- 開発コミュニティの活動は、商品開発現場の今日の問題解決に直接寄与しない かもしれませんが、中長期にわたって利用可能な SW 資産を構築 していくには今以上に コミュニティ活動に目を向けることが重要 になるのではないのでしょうか。