



CIVIL  
INFRASTRUCTURE  
PLATFORM

OSAKA NDS Embedded Linux Cross Online Forum #11  
超長期メンテナンスLinuxを支える  
CIPのテスト基盤とコアパッケージ群

藤田 和寛

Civil Infrastructure Platform/ルネサス エレクトロニクス株式会社  
10th July 2020

# 自己紹介

## 藤田 和寛

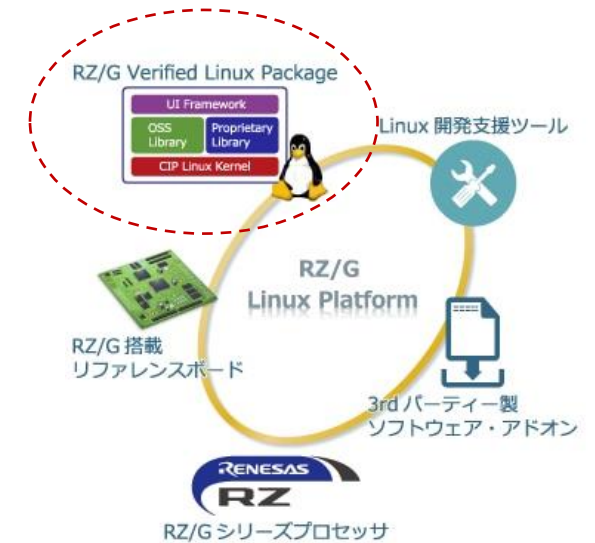
ルネサス エレクトロニクス株式会社  
IoT・インフラ事業本部 ソフトウェア開発統括部  
ソフトウェアプラットフォーム開発第二部  
Linuxプラットフォーム第一課

2007年頃～組み込みLinuxドライバ開発やkernel調査などを担当（前職）

2018年よりRZ/G Linux Package開発を担当し、CIPの活動にも参加開始

2019年頃よりRZ/G2向けを中心にVerified Linux Package開発の全体リーダー

2020年現在、CIP Linuxをルネサスの産業向け製品全体に展開するべく活動中



# アジェンダ

- CIPとは
- CIP Testing
- CIP Core
- Q&A



# CIPとは



# この社会を実現する様々な産業/インフラ機器

## Transport



Rail automation



Vehicle control



Automatic ticket gates

## Energy



Power Generation



Turbine Control



Turbine Control

## Others



Building automation



Broadcasting



Healthcare

## Industry



Industry automation



CNC control



Industrial communication





# 共通の課題

## 産業グレードの 品質の確保

- Reliability
- Functional Safety
- Real-time capabilities

## 長期間の メンテナンス

- Product life-cycles of decades
- Backwards compatibility
- Standards

## セキュリティの 担保

- Security & vulnerability management
- Firmware updates
- Minimize risk of regressions

# CIPが解決



— CIVIL —  
**INFRASTRUCTURE**  
— PLATFORM —



CIVIL  
INFRASTRUCTURE  
— PLATFORM —



# Civil Infrastructure Platform (CIP) とは？



## ■ CIP Project

- 産業用機器に共通した課題を解決するべく設立された、Linux FoundationのProject

## ■ Open Source Base Layer (OSBL) の確立を目指す

- 産業機器が求める要件を満たすLinuxの提供
- 高い信頼性・安定性・長期間のメンテナンス  
(同一バージョンソフトウェアの**10年超メンテナンス**)
- Upstream Communityとの密な連携

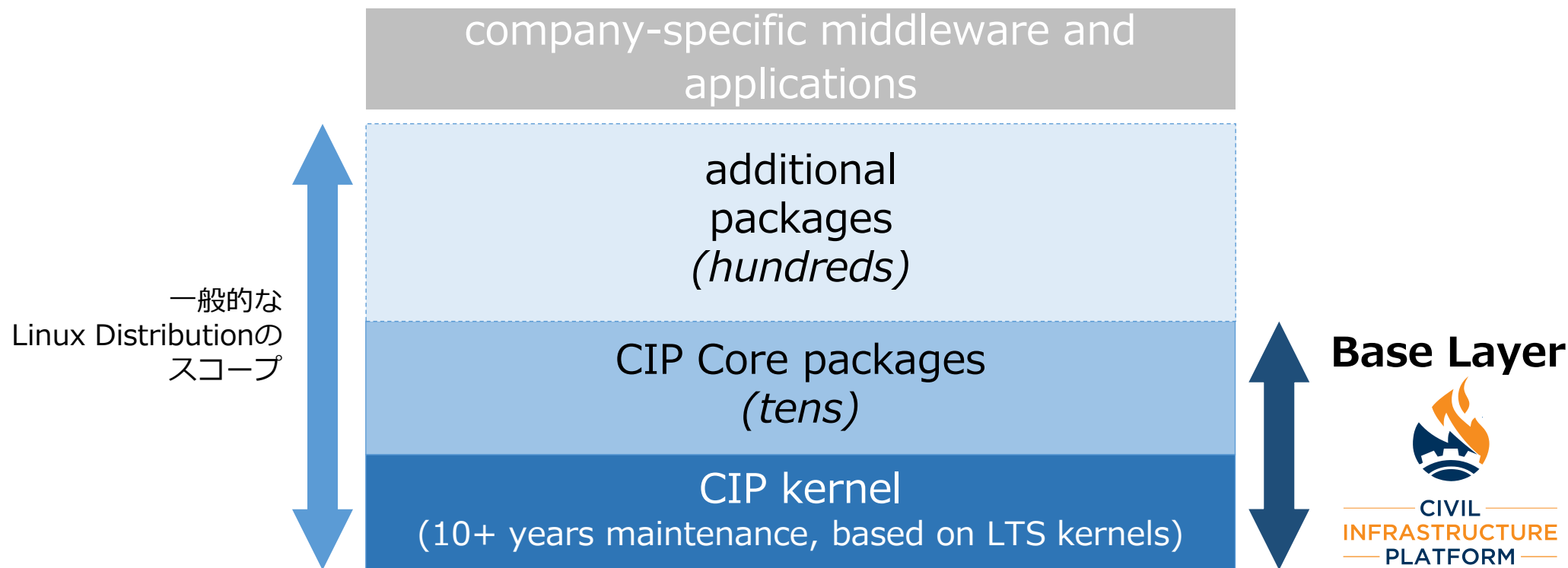


# “Open Source Base Layer (OSBL)” とは?



## ■ OSBL ≠ 新ディストリビューション

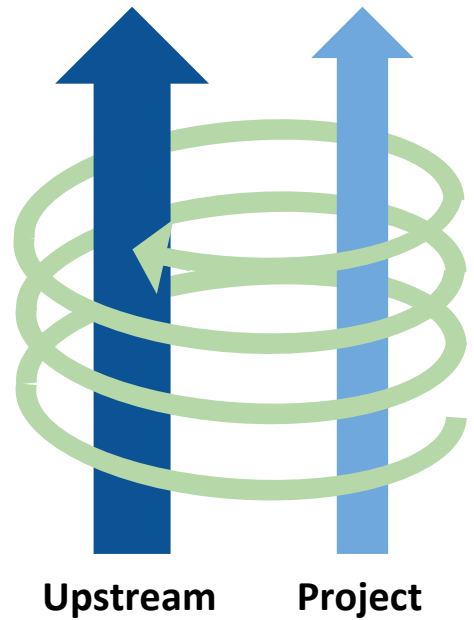
- OSBLとは、関連するオープンソースプロジェクトを利用する（またそれらに影響を与える）産業機器向けLinuxディストリビューションの一階層



# Development Models

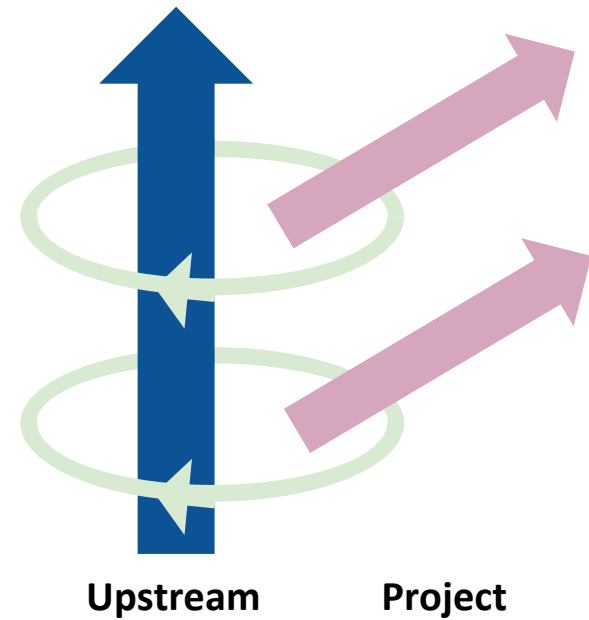


## “Upstream First” Project



- The project can share its results with the Upstreams
- The project can fulfill longer time maintenance

## “Own Community” Project

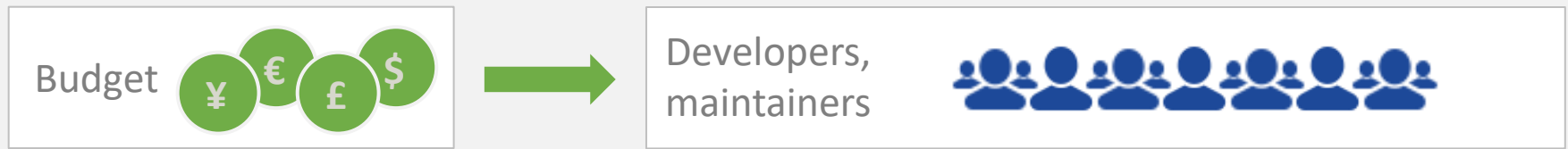
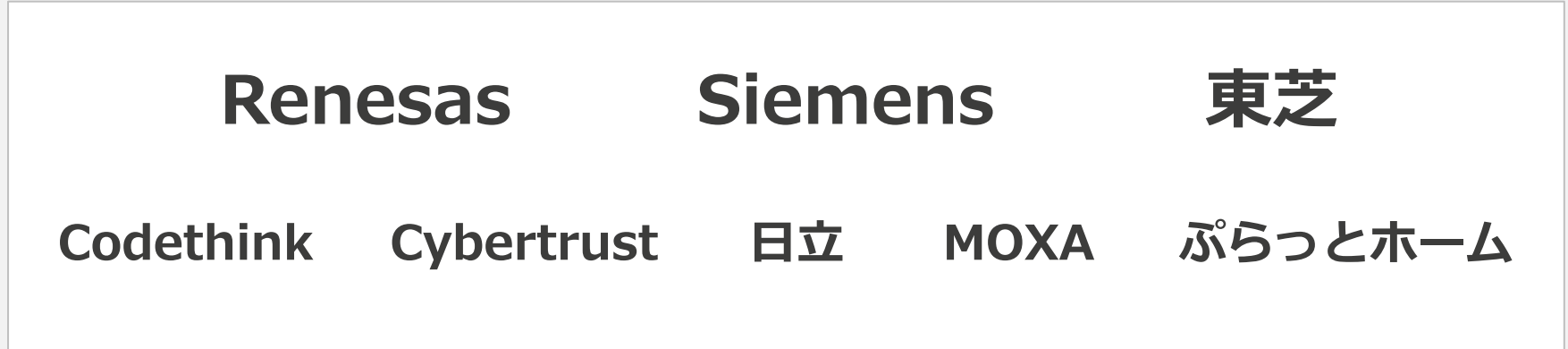


- The project can develop their code very quickly
- The project faces difficulties to backport Upstream patches due to conflicts as time goes by

# CIPの活動はメンバー企業によって支えられている



CIVIL  
INFRASTRUCTURE  
PLATFORM



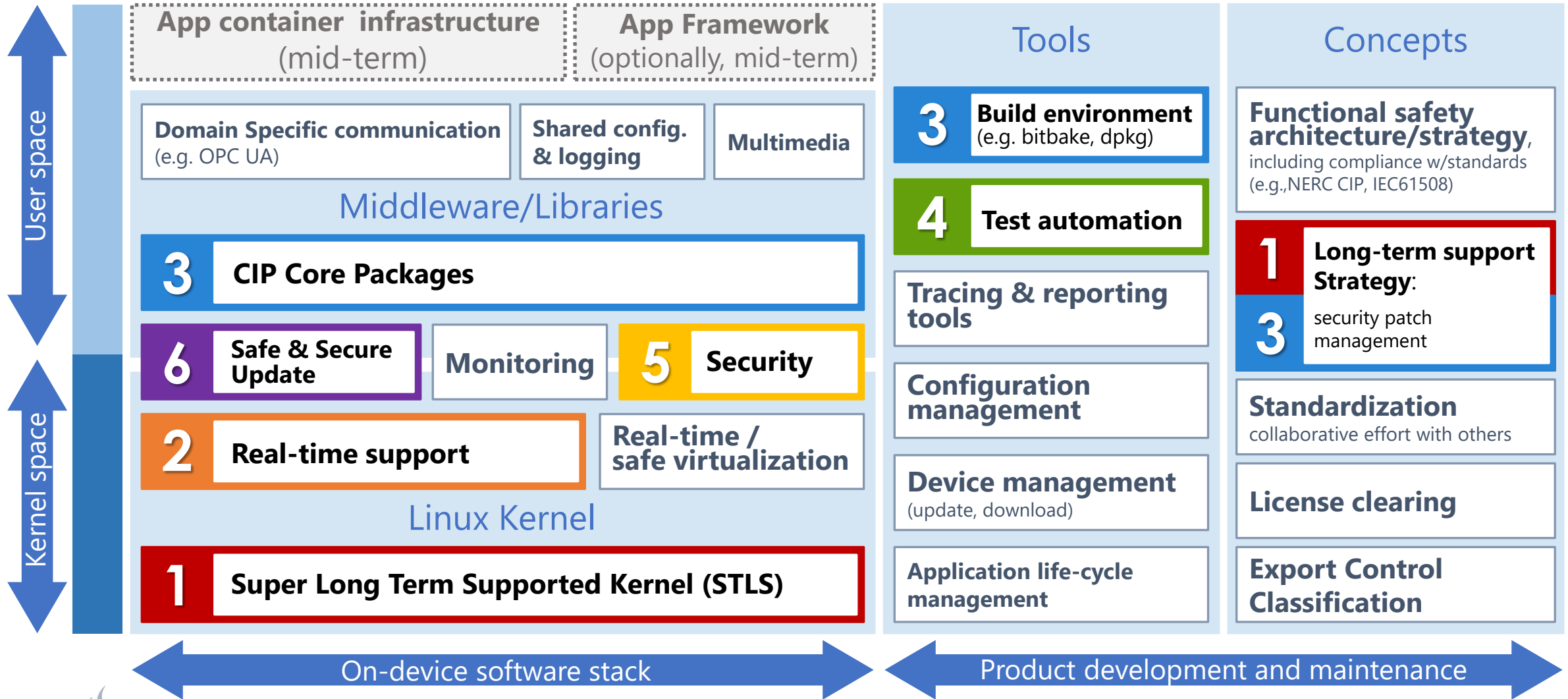
↓ オプション：特定のプロジェクトへ  
ファンディング

↑ ↓ コントリビューション & 利用/インテグレーション





# CIPのスコープ

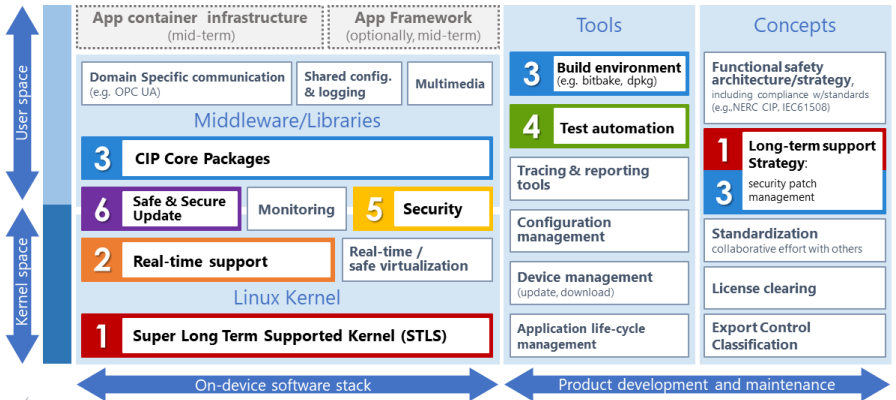


# CIPガバナンス構造とプロジェクト



Governing Board (GB)

Technical Steering Committee (TSC)



本日のテーマ

1	2	3	4	5	6	(*): Workgroup
SLTS kernel	Real-time	CIP Core	Testing	Security WG(*)	Software Update WG	
✓	✓	✓	✓	✓	✓	Industrial grade
✓		✓	✓		✓	Sustainability
✓		✓	✓	✓	✓	Security

CIP Projects and its scopes



# CIP Testing

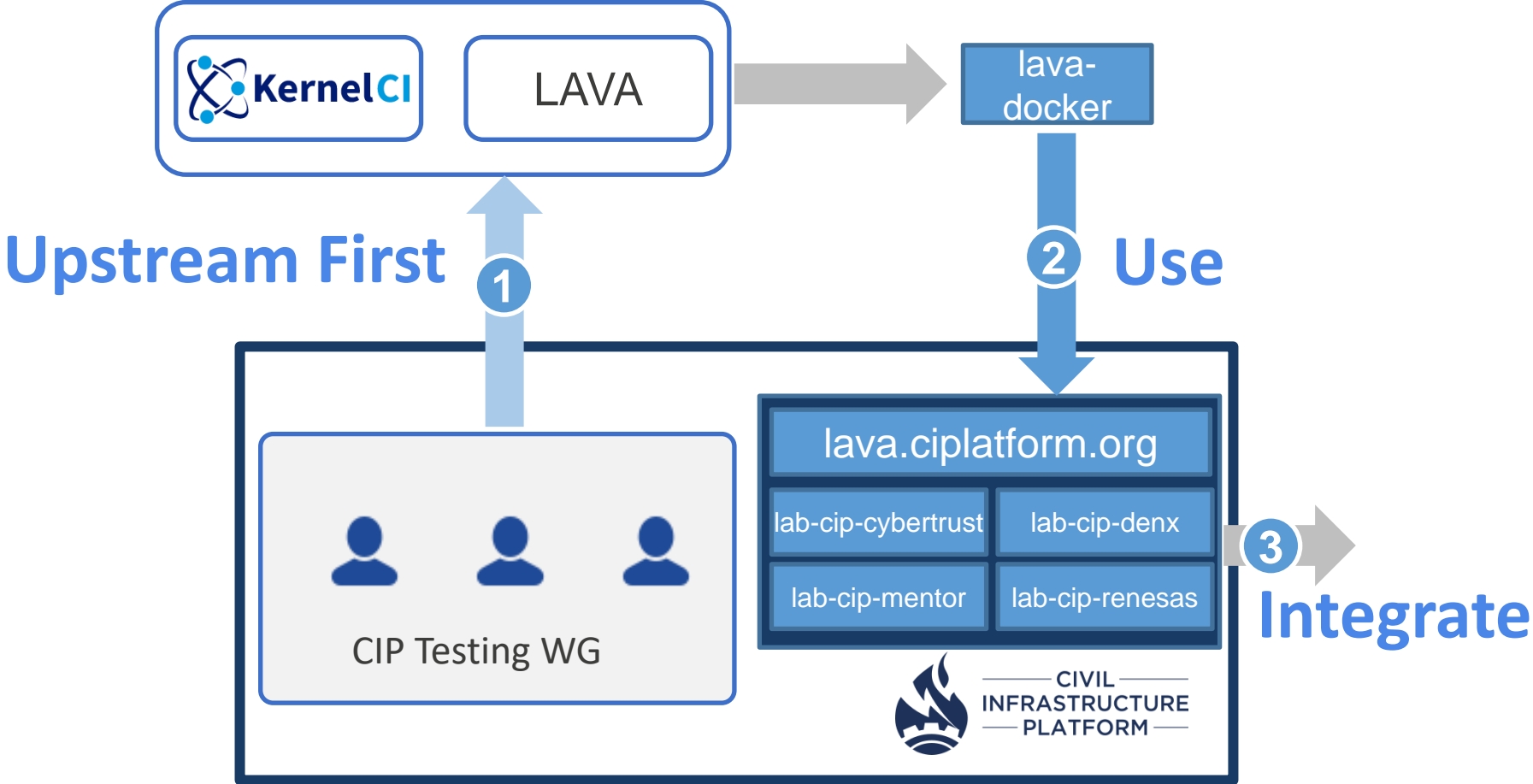


# CIP Testing Goals

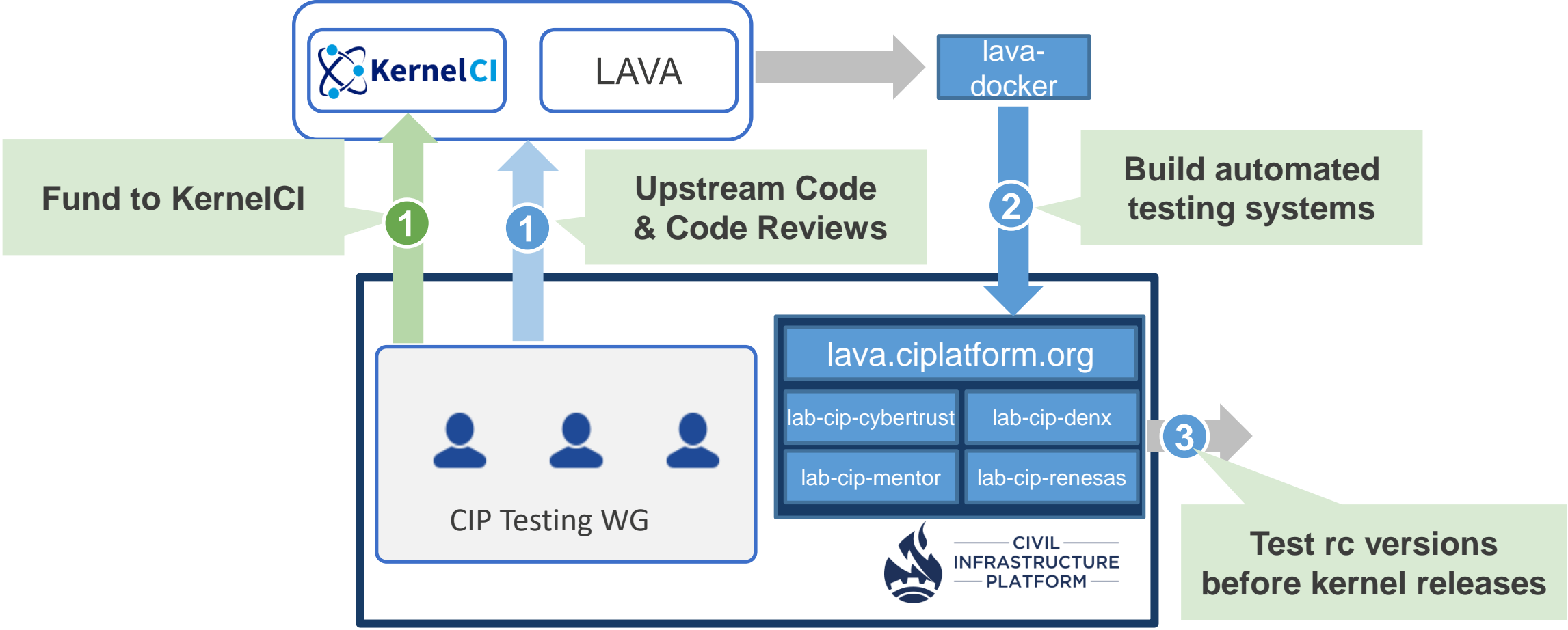


- 中央での一元的コントロール&分散テスト
  - ・ 世界中のCIP開発者がCIPのリファレンスプラットフォームを、ヨーロッパ・インド・日本に構築された4つのテストラボで自由にテストすることができる
- Continuous Integration (CI) での自動テスト
  - ・ コスト最適化された、kernelの定期的長期リリースの維持
- 全CIPリファレンスプラットフォームのサポート
  - ・ リファレンスは現在7つ、追加あれば順次サポート

# CIP Testing Team



# CIP Testing Team



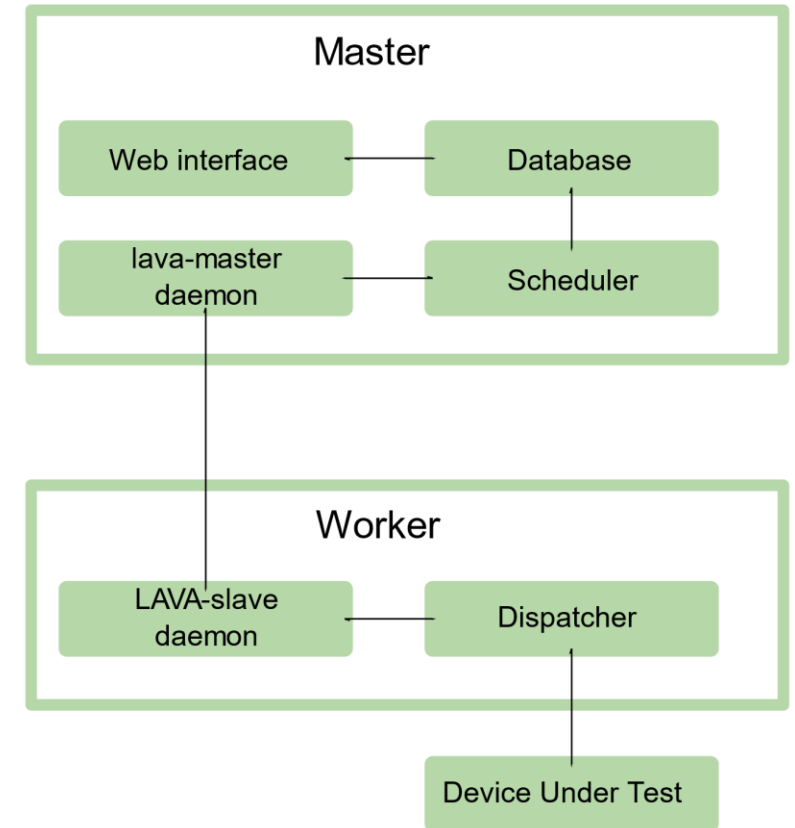


# LAVAとは



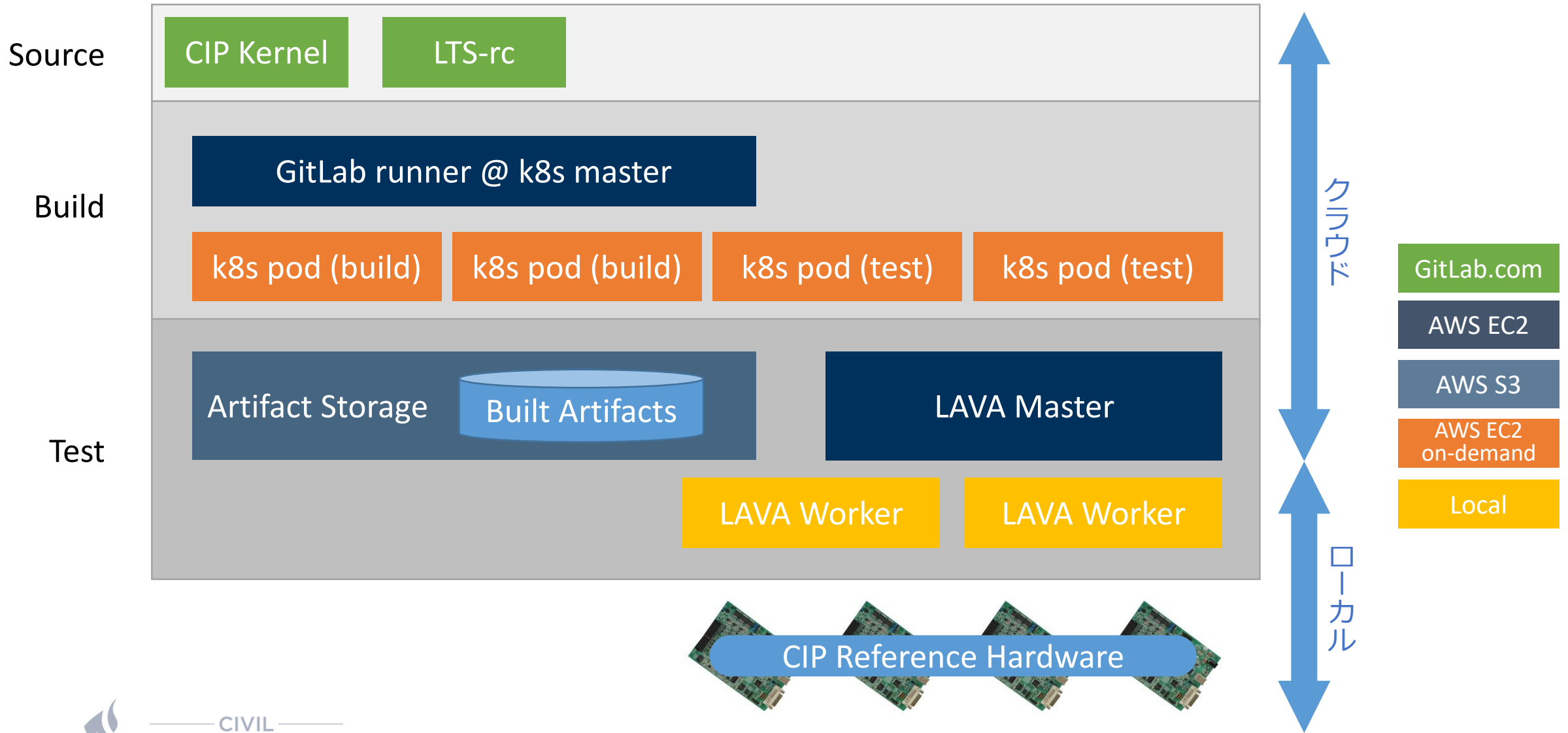
## ■ Linaro Automated Validation Architecture

- <https://git.lavasoftware.org/lava>
- Linaroにより最初の版が2010年にリリースされ、現在も活発に開発が続いている
- OSを物理/仮想マシンにデプロイしテストを実行するCI (continuous integration) システム (の一部)
- Linaroが作成したものを始め、オープンソースの様々なテストケースを利用可能
- CIPでは、旧テスト環境のBoard at Deskの時代からLAVAをテスト環境として利用

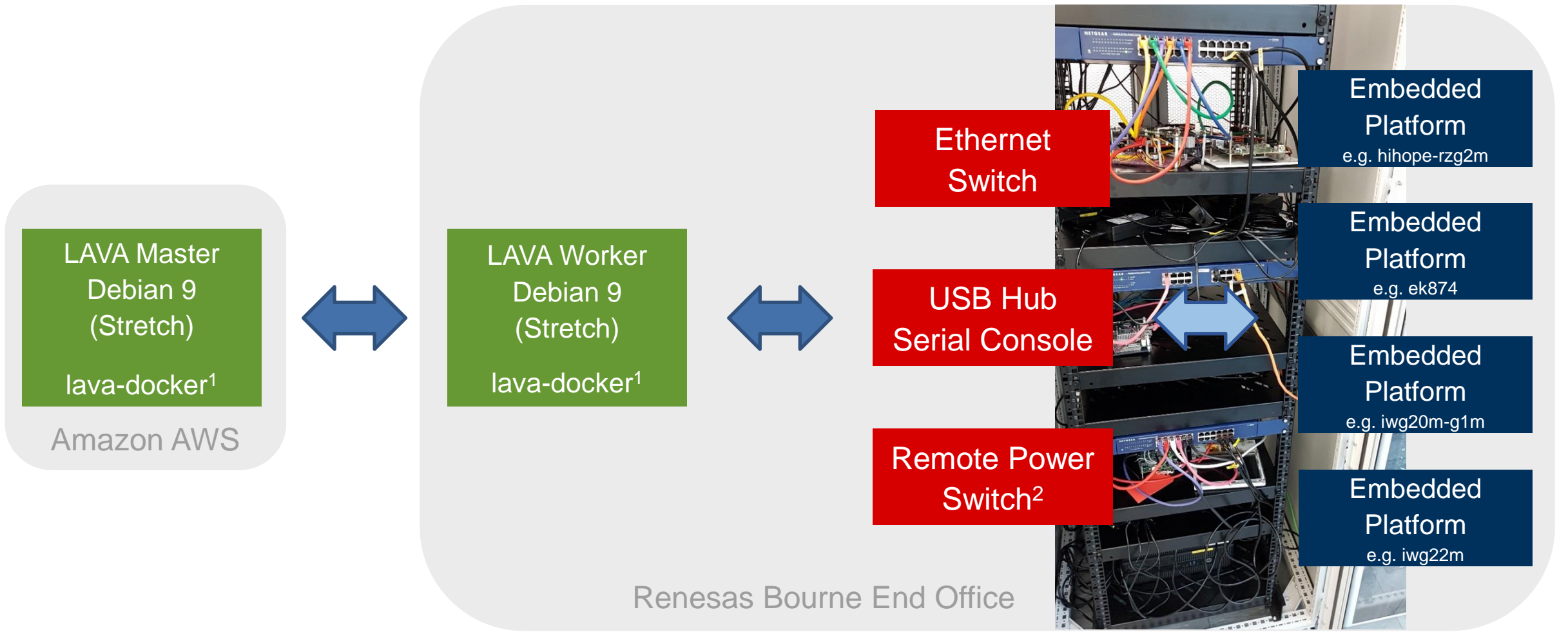


<https://docs.lavasoftware.org/lava/index.html>

# Testing Architecture Overview

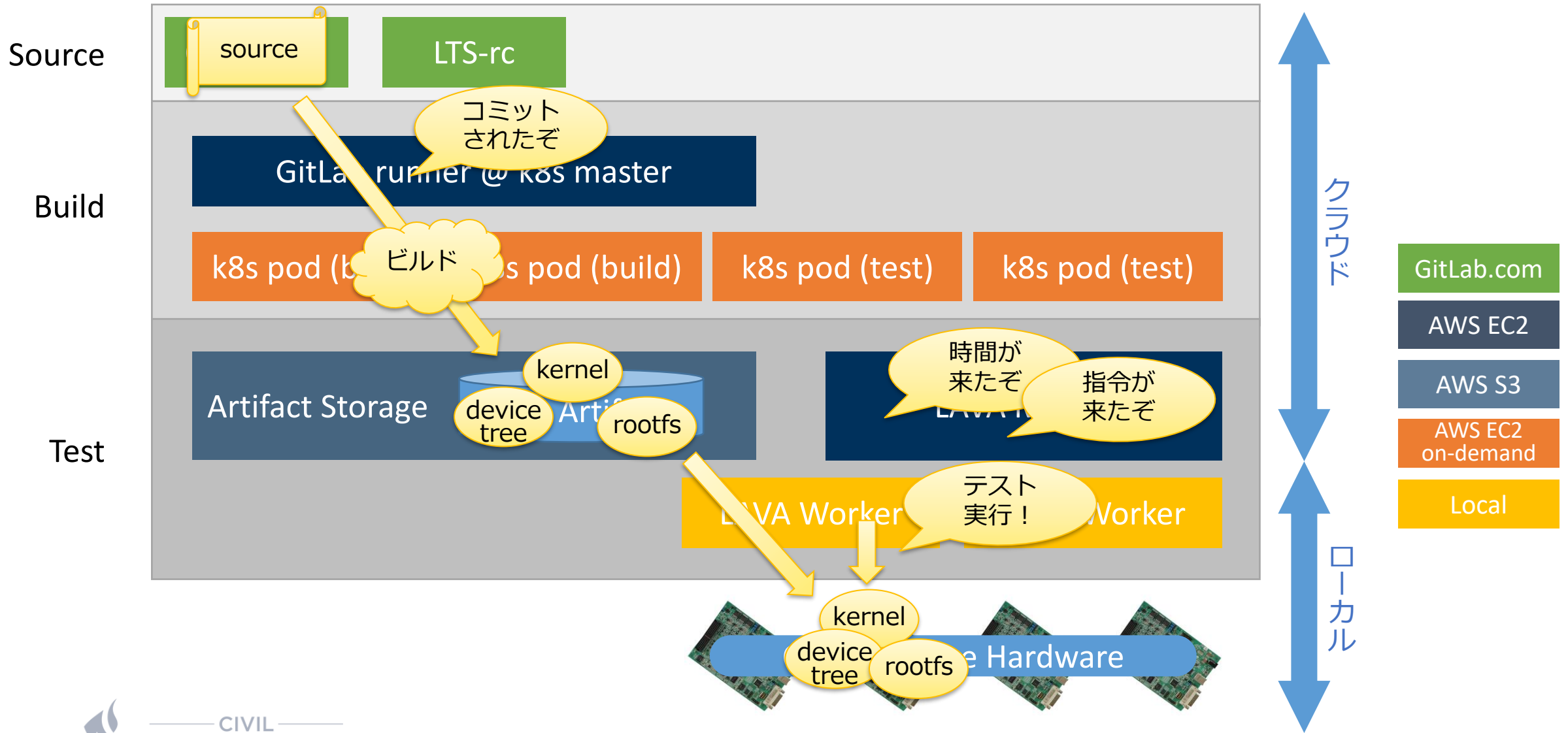


# LAVA Implementation



<sup>1</sup><https://gitlab.com/cip-project/cip-testing/lava-docker>  
<sup>2</sup><https://www.lindy.co.uk/networking-c5/ipower-switch-classic-8-power-management-over-ip-p4456/s4598>

# Testing Architecture Overview



# CIP Reference Boards



CIP Reference Boards		Supported Kernels			
Platform	Architecture	SLTS v4.4	SLTS v4.4-rt	SLTS v4.19	SLTS v4.19-rt
AM335x Beaglebone Black	Armv7	Y	Y <sup>1</sup>	Y	Y <sup>1</sup>
Cyclone V DE0-Nano-SoC Development Kit	Armv7	N	N	Y	Y <sup>1</sup>
QEMU	x86_64	Y	Y <sup>1</sup>	Y	Y <sup>1</sup>
RZ/G1M iWave Qseven Development Kit	Armv7	Y	Y <sup>1,2</sup>	Y	Y <sup>1,2</sup>
RZ/G2M HopeRun HiHope	Armv8	N	N	Y	Y <sup>1,2</sup>
SIMATIC IPC227E	x86-64	N	N	Y	Y <sup>1</sup>
OpenBlocks IoT VX2	x86-64	N	N	Y	Y <sup>1</sup>

CIP Reference Board Candidate		Supported Kernels			
Platform	Architecture	SLTS v4.4	SLTS v4.4-rt	SLTS v4.19	SLTS v4.19-rt
Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit	Armv8	N	N	Y	Y <sup>1</sup>

<sup>1</sup> Tested with standard Kernel configuration (non-RT)

<sup>2</sup> Tested with Real-Time enabled Kernel configuration



# Automated Testing



- Currently CIP are running the following tests:
  - Boot test
    - `uname -a`
  - Spectre/Meltdown checker
  - LTP
    - `ltp-cve-tests`, `ltp-dio-tests`, `ltp-fs-tests`, `ltp-ipc-tests`, `ltp-math-tests`, `ltp-open-posix-tests`, `ltp-sched-tests`, `ltp-syscalls-tests` and `ltp-timers-tests`
  - Cyclictest+Hackbench
    - This test measures event latency in the Linux Kernel, with `hackbench` running in the background to stress the system.
- In Development:
  - Kselftest

# LAVA Results



- 誰でもテスト定義や結果を見ることができます  
<https://lava.ciplatform.org/scheduler/alljobs>

LAVA Scheduler / Jobs

All Jobs

Show 25 entries

ID	Actions	State	Device ID	Device type	Description	Submitter	Submit Time	End Time
17602		Complete	zynqmp-zcu102-01	zynqmp-zcu102	zynqmp-zcu102 healthcheck	lava-health	June09, 2:26a.m.	June09, 2:27a.m.
17601		Complete	qemu-03	qemu	qemu x86_64 healthcheck	lava-health	June09, 2:26a.m.	June09, 2:27a.m.
17583		Complete	r8a7743-iwg20d-q7-01	r8a7743-iwg20d-q7	r8a7743-iwg20d-q7 healthcheck	lava-health	June08, 9:56a.m.	June08, 9:58a.m.
17582		Complete	zynqmp-zcu102-01	zynqmp-zcu102	zynqmp-zcu102 healthcheck	lava-health	June08, 1:27a.m.	June08, 1:27a.m.
17581		Complete	qemu-03	qemu	qemu x86_64 healthcheck	lava-health	June08, 1:27a.m.	June08, 1:27a.m.
17580		Complete	qemu-01	qemu	ci-iwamatsu-linux-4.4.y-cip-rc_bzImage_cip_qemu_defconfig_4.4.222-cip45_a5f3949c_x86_cip_qemu_defconfig_ltp-timers-tests	CIP-Testing	June08, 12:01a.m.	June08, 12:27a.m.
17579		Complete	qemu-04	qemu	ci-iwamatsu-linux-4.4.y-cip-rc_bzImage_cip_qemu_defconfig_4.4.222-cip45_a5f3949c_x86_cip_qemu_defconfig_ltp-syscalls-tests	CIP-Testing	June08, 12:01a.m.	June08, 1:02a.m.

LAVA Results / Test Job 17580 / Suite 2\_ltp-timers-tests

### Results for test suite 2\_ltp-timers-tests - Test Job 17580

Exports ?

Test suite export :  CSV or  YAML

Show 25 entries

Name	Test Set	Result	Measurement	Units	Logged	Bug Links
timer_create02	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
timer_create03	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
timer_create04	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
timer_delete02	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
timer_delete03	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
timer_settime02	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
timer_settime03	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]
leapsec_timer	—	✓ pass	—	—	06/08/2020 12:27 a.m.	[0]

# An Example of Test Definitions



## ■ Boot test (uname -a)

<https://lava.ciplatform.org/scheduler/job/20696/definition>

```
# ACTION_BLOCK
actions:
- deploy:
  timeout:
    minutes: 2
  to: tftp
  kernel:
    url: https://s3-us-west-2.amazonaws.com/.../Image
    type: image
  ramdisk:
    url: https://s3-us-west-2.amazonaws.com/.../initramfs_64.cpio.gz
    compression: gz
  dtb:
    url: https://s3-us-west-2.amazonaws.com/.../r8a774a1-...-ex.dtb
  os: oe

#BOOT_BLOCK
- boot:
  timeout:
    minutes: 5
  method: u-boot
  commands: ramdisk
  prompts:
    - 'linaro-test'
    - 'root@debian:~#'
    - '/ #'
```

```
# TEST_BLOCK
- test:
  timeout:
    minutes: 5
  definitions:
  - repository:
    metadata:
      format: Lava-Test Test Definition 1.0
      name: kernel-version-basic
      description: "check kernel version"
      os:
        - oe
      scope:
        - functional
    run:
      steps:
        - lava-test-case uname --shell uname -a
    from: inline
    name: kernel-version-inline
    path: inline/kernel-version-:basic.yaml
```

# An Example of Test Definitions



- Boot test (uname -a)

<https://lava.ciplatform.org/scheduler/job/20696>

```
<LAVA_SIGNAL_STARTRUN 0_kernel-version-inline 20696_1.4.2.4.1>
+ lava-test-case uname --shell uname -a
Received signal: <STARTRUN> 0_kernel-version-inline 20696_1.4.2.4.1
Starting test lava.0_kernel-version-inline (20696_1.4.2.4.1)
Skipping test definition patterns.
<LAVA_SIGNAL_STARTTC uname>
Linux 192.168.1.95 4.19.124-cip27-arm64-renesas #1 SMP PREEMPT Sat May 23 23:11:31 UTC 2020 aarch64 GNU/Linux
<LAVA_SIGNAL_ENDTC uname>
<LAVA_SIGNAL_TESTCASE TEST_CASE_ID=uname RESULT=pass>
+ set +x
Received signal: <STARTTC> uname
Received signal: <ENDTC> uname
Received signal: <TESTCASE> TEST_CASE_ID=uname RESULT=pass
case: uname
case_id: 686995
definition: 0_kernel-version-inline
result: pass
<LAVA_SIGNAL_ENDRUN 0_kernel-version-inline 20696_1.4.2.4.1>
<LAVA_TEST_RUNNER EXIT>
Received signal: <ENDRUN> 0_kernel-version-inline 20696_1.4.2.4.1
Ending use of test pattern.
Ending test lava.0_kernel-version-inline (20696_1.4.2.4.1), duration 0.03
case: 0_kernel-version-inline
case_id: 686996
definition: lava
```

# Collaboration with KernelCI



- Linux Foundation projectとなったKernelCIともコラボ開始  
<https://foundation.kernelci.org/>
- KernelCIのスポンサー：Baylibre, **CIP**, Collabra, Foundries.io, Google, Microsoft, Redhat
- LTPを皮切りに、KernelCIがサポートするテストを相互運用することでテスト範囲拡大&品質向上
- 既に10以上のデバイスタイプで、KernelCIから直接CIPのテストラボにテストジョブを投入可能





# Plans



## Use **cip-core-tiny** and **cip-core-generic** for all Kernel testing

Finish test support for all reference platforms

Improve Kernel test results front end

- Setup and configure [kernelci.ciplatform.org](https://kernelci.ciplatform.org) (or perhaps [cip.kernelci.org](https://cip.kernelci.org)?)
- **Integrate CIP's GitLab based Kernel builds with KernelCI's backend**
- Automatic test regression detection

Test case expansion

- **LTP for all reference boards**
- kSelftest

Add 'small instance' support to gitlab-cloud-ci for test jobs

Speed up build times / test times

Add monitoring for the various CI components (LAVA master, LAVA workers etc.)

Add support for the next SLTS Kernel

Add support for the next CIP Core versions

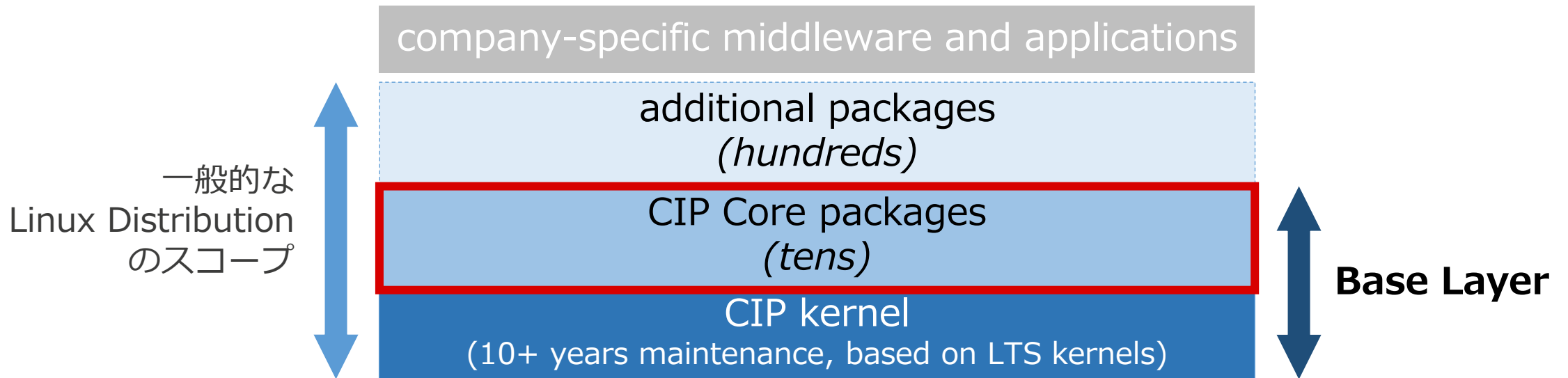


# CIP Core

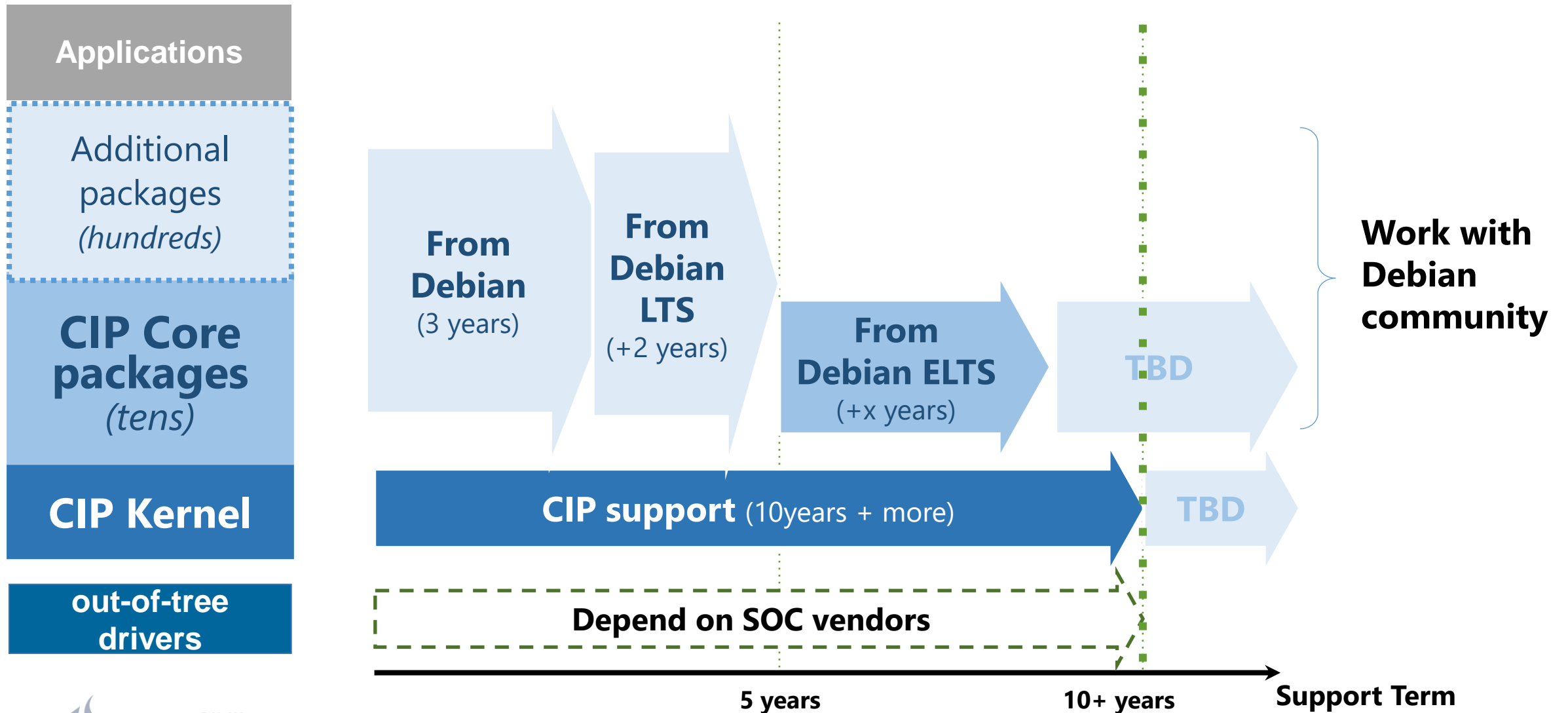
# CIP Coreとは



- ユーザー層のソフトウェアとツールにフォーカスした活動
  - “CIP Core packages” を定義し、長期メンテナンスを実施
  - “CIP Core packages” のリファレンス実装を提供
  - CIPのリファレンスハードウェアでテストを実施



# CIPによる超長期メンテナンスの進め方



# CIP Core: Implementation

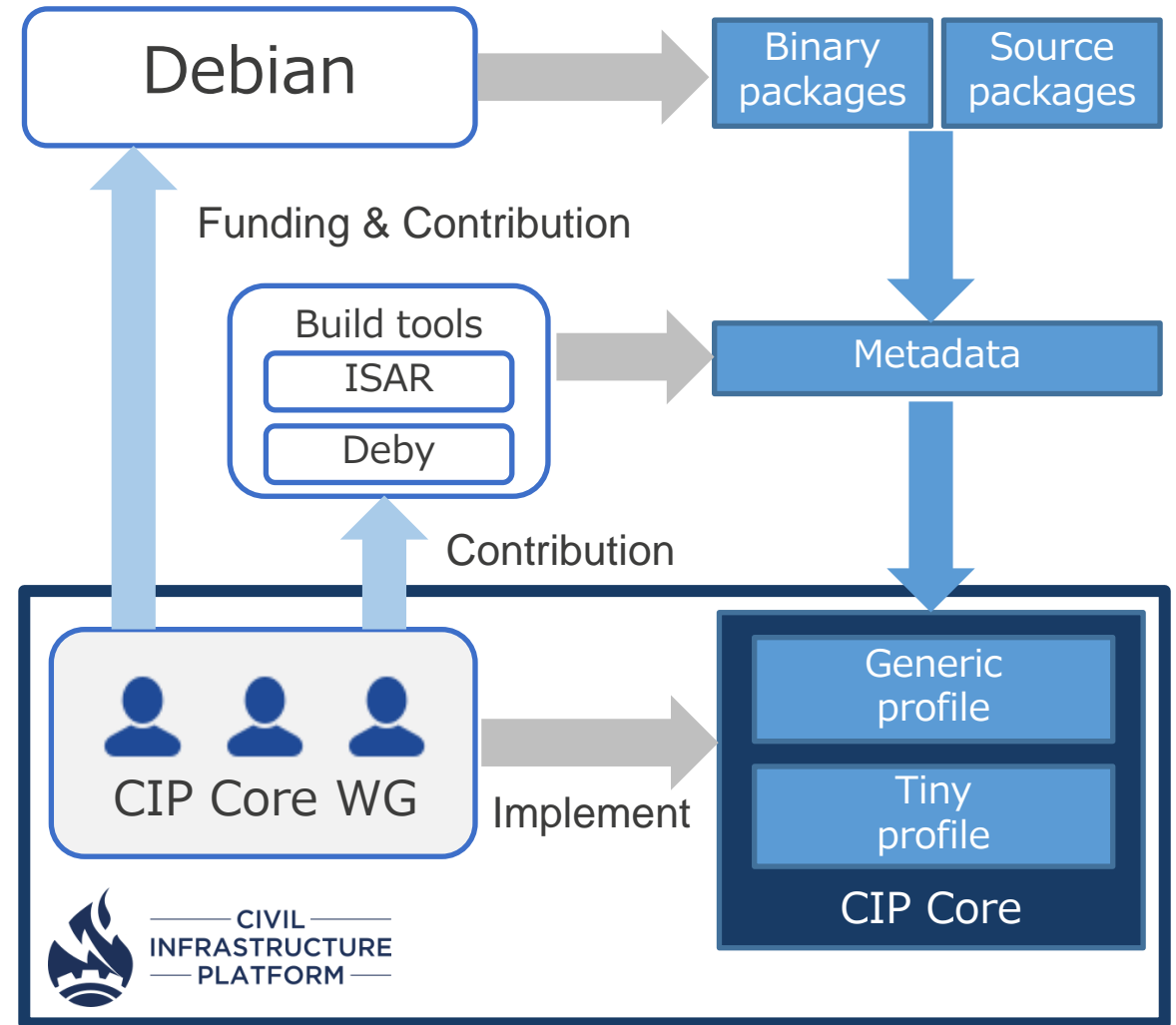


## ■ 2つのProfile

	Generic	Tiny
Approach	Binary packages	Source packages
Build tool	ISAR	Deby (meta-debian)

## ■ Debianがベース

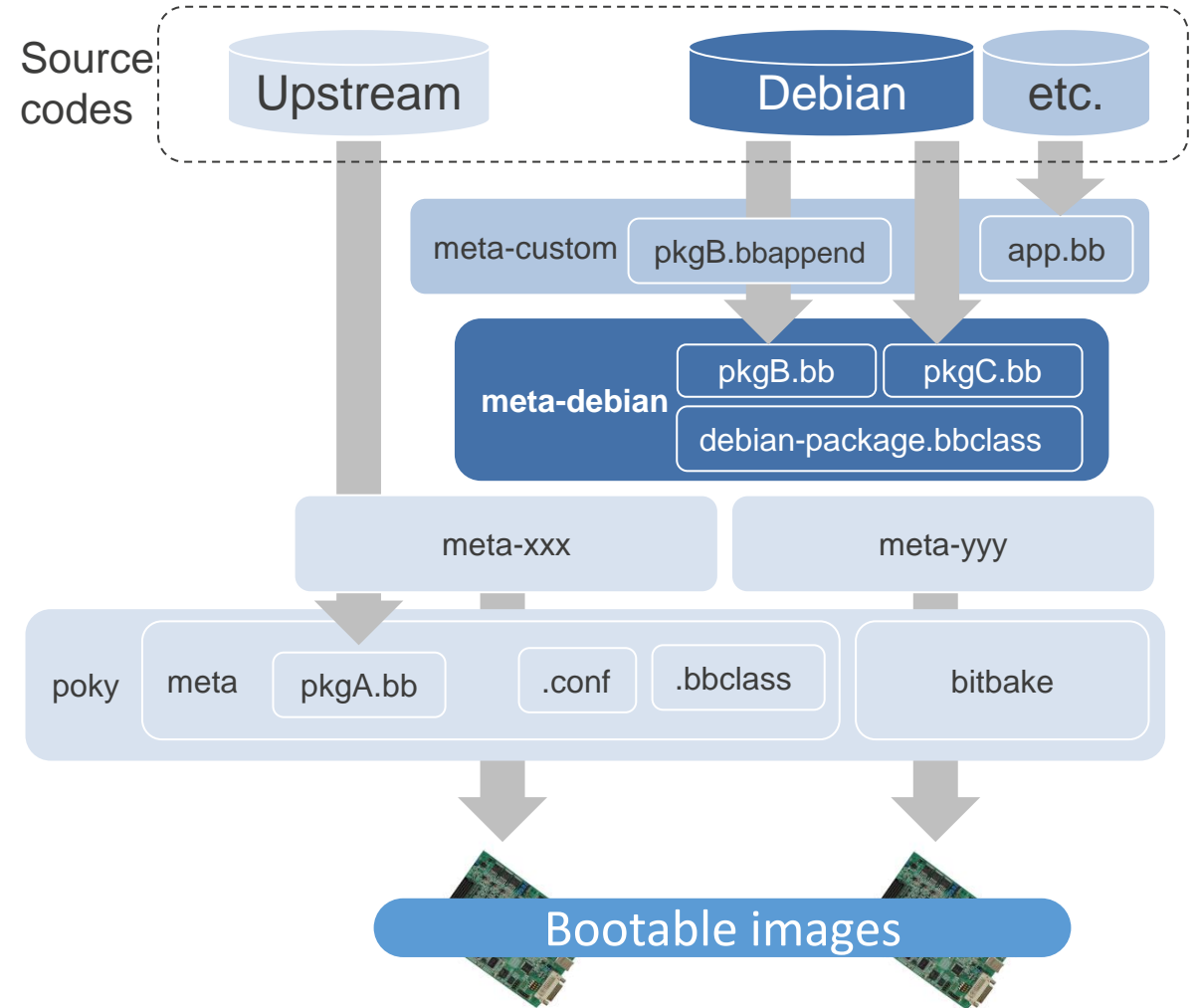
	Debian 8 (jessie)	Debian 9 (stretch)	Debian 10 (buster)
kernel 4.4	Tiny	Generic	N/A
kernel 4.19	N/A	Generic	Generic & Tiny





# Deby (meta-debian) とは

- Yocto Project extension
  - Debianのソースコードを Yocto Project の Build system で扱うためのレイヤー
  - <https://github.com/meta-debian/meta-debian>
- CIP Core Tiny Profileで使用
  - poky + meta-debian の組み合わせでビルド
  - <https://gitlab.com/cip-project/cip-core/deby>
  - Target poky branch: **warrior** (Yocto Project 2.7)



# CIP Core Package List



## ■ Initial package list for Debian 10 (Buster)

- [https://gitlab.com/cip-project/cip-core/cip-pkglist/-/blob/master/pkglist\\_buster.yml](https://gitlab.com/cip-project/cip-core/cip-pkglist/-/blob/master/pkglist_buster.yml)
  - Debianのminimum base system相当 (debootstrap --variant=minbase) の58パッケージが承認済み
  - IEC-62443-4-2 の要求を満たすための下記12パッケージも長期メンテナンス対象として追加される見込み  
aide, chrony, fail2ban, nftables, openssh, openssl, pam-pkcs11, sudo, syslog-ng, tpm2-abrmd, tpm2-tools, tpm2-tss

## ■ Package Decision Process (PDP)

- <https://gitlab.com/cip-project/cip-core/cip-pkglist/-/blob/master/doc/pdp.md>
- CIP member company (のみ) が必要な情報とともに propose
  - 追加理由, パッケージ名 (source, binary), 対応するDebian version, 依存関係リスト, 過去のCVE数, ... 等
  - 依存関係を自動検査する、提案書 (yaml形式) 作成ヘルパースクリプトあり
- レビュー期間後、50%以上のレビュアーの合意を得て TSC により承認

# Plans



Test CIP Core (isar-cip-core and deby) on LAVA

- Security tests
- Other tests

Merge security layer into CIP Core (isar-cip-core and deby)

Future topics:

- SDK
- Reproducible builds



# まとめ



- CIPプロジェクトは
  - Linux kernelだけではなく、CIP Core packageとして**ユーザー層も超長期にメンテナンス（10年超）**
  - カーネル層とユーザー層を組合わせ、**リファレンスハードウェア上で自動テストの実行を継続**



- この社会に必要な産業グレードOSBLを提供
- Open Sourceで連携し、各企業の共通課題を協力して解決

# 興味を持たれたら



## ■ 是非使ってみてください

- CIP web site:
  - CIP project portal: <https://www.cip-project.org>
- CIP source code
  - CIP GitLab: <https://gitlab.com/cip-project>
  - CIP kernel: <git://git.kernel.org/pub/scm/linux/kernel/git/cip/linux-cip.git>
- CIP LAVA Lab
  - 現在のScheduler Status: <https://lava.ciplatform.org/scheduler/>
  - CIPのLAVA環境: <https://gitlab.com/cip-project/cip-testing/lava-docker>
- CIP Core
  - Package list: <https://gitlab.com/cip-project/cip-core/cip-pkglist>

## ■ もっと重要 : **CIPに加入しませんか？！**



## ■ CIP Mini Summit @ Open Source Summit North America 2020

- July 2, 2020 Virtual Experience
  - State of Civil Infrastructure Platform
  - CIP Kernel Team Activities towards Super Long Term Support
  - Status update for testing within CIP
  - CIP Security towards achieving industrial grade security

## ■ CIP Mini Summit @ Open Source Summit Europe 2019

- October 31, 2019 Lyon Convention Centre
  - State of Civil Infrastructure Platform
  - CIP SLTS kernel development  
(e.g. Patch management for collaboration with stable kernel team)
  - Security in industrial systems and its future
  - Safe software update for industrial IoT devices
  - Use cases of CIP open source base layer

録音やスライドもあります: <https://wiki.linuxfoundation.org/civilinfrastructureplatform/cipconferences>



# References



CIP Blog: <https://www.cip-project.org/blog>

CIP news: <https://www.cip-project.org/news/in-the-news>

CIP wiki: <https://wiki.linuxfoundation.org/civilinfrastructureplatform/>

CIP Testing overview:

<https://wiki.linuxfoundation.org/civilinfrastructureplatform/ciptesting/centralisedtesting/cioverview>

Twitter: @cip\_project

CIP Mailing list: [cip-dev@lists.cip-project.org](mailto:cip-dev@lists.cip-project.org)

Linaroによるテスト定義: <https://github.com/Linaro/test-definitions/tree/master/automated/linux>

いまさら聞けないCIP入門:

<https://monoist.atmarkit.co.jp/mn/articles/1909/24/news015.html>

過去のEmbedded Linux Cross Forumでの講演資料もご覧ください。

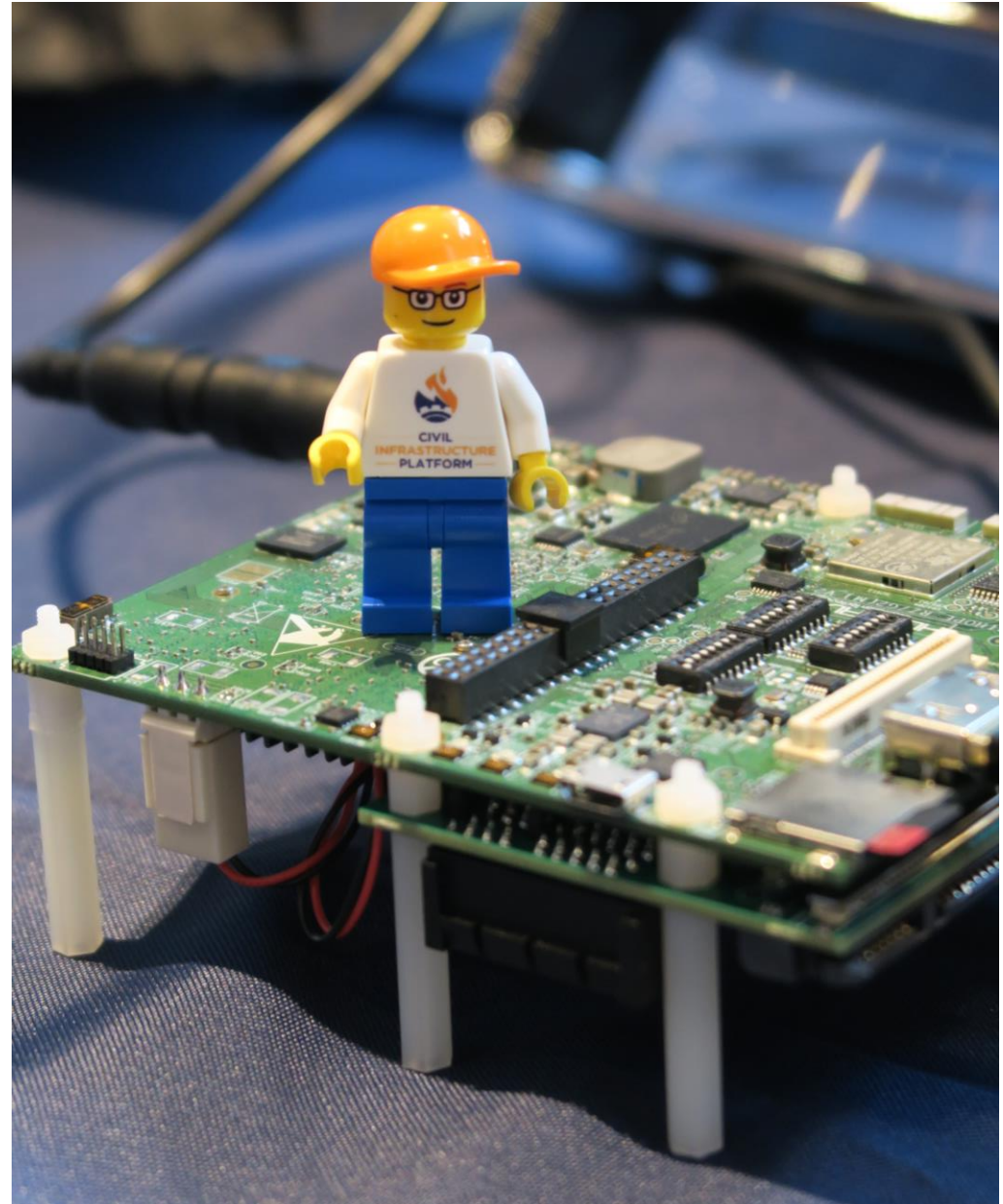
<https://www.nds-osk.co.jp/forum/onlcf7.html>

<https://www.nds-osk.co.jp/forum/onlcf8.html>

<https://www.nds-osk.co.jp/forum/onlcf9.html>



# Questions?





# Thank you!