



———— CIVIL ————  
**INFRASTRUCTURE**  
———— PLATFORM ————

# Getting (More) Debian Into Our Civil Infrastructure

**Jan Kiszka, Siemens AG**

MiniDebConf, Hamburg, May 20, 2018



An aerial photograph of San Francisco, California, showing the city's dense skyline of skyscrapers and the Golden Gate Bridge spanning the water. The image is used as a background for a presentation slide.

# Our Civilization is run by Linux



## Transport



**Rail automation**



**Vehicle control**



**Automatic ticket gates**

## Energy



**Power Generation**

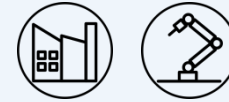


**Power Distribution**



**Turbine Control**

## Industry



**Industry automation**



**CNC control**



**Industrial communication**

## Others



**Healthcare**



**Building automation**



**Broadcasting**



An aerial photograph of San Francisco, showing the city's dense skyline with numerous skyscrapers, including the Transamerica Pyramid. The city is situated on a peninsula, with the San Francisco Bay and the Golden Gate Bridge visible in the background. A blue semi-transparent box is overlaid on the top left of the image.

There are issues to be solved...





# A Railway System:

## 25-50 years products life-cycle

with very reluctant nature for product update and upgrade of hardware and base software platform

# Railway Example



**3 – 5 years development time**

**2 – 4 years customer specific extensions**

**1 year initial safety certifications / authorization**

**3 – 6 months safety certifications / authorization for follow-up releases  
(depending on amount of changes)**

**25 – 50 years lifetime**

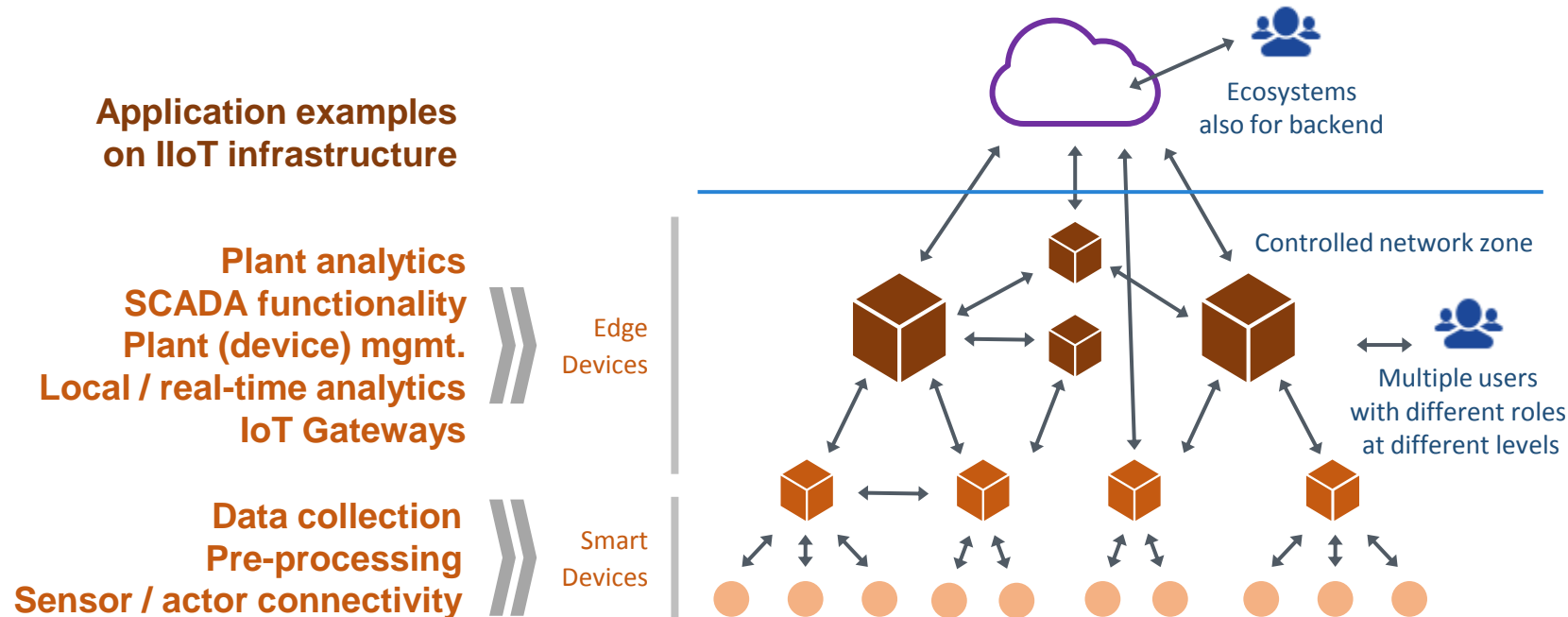


# Industrial IoT: Edge and Fog Computing



Functionality is moving from the cloud to the “Edge”

- Increasing number of networked industrial-grade devices
- Security management requires harmonized software landscape



# The Problems we face ...



- The systems that support our modern civilization need to **survive for a VERY LONG TIME**. Until now the corresponding industrial grade super long term maintenance has been **done individually by each company**.
- These systems not only have to survive for a long time, they must be **“INDUSTRIAL GRADE”** (robust, secure and reliable). And at the same time the industry will also need to **catch up with the latest technology trends**



# **The genesis of a collaborative project**

# CIP is our solution...

Establishing an Open Source Base Layer of industrial-grade software to enable the use and implementation of software building blocks for Civil Infrastructure Systems

<https://www.cip-project.org/>

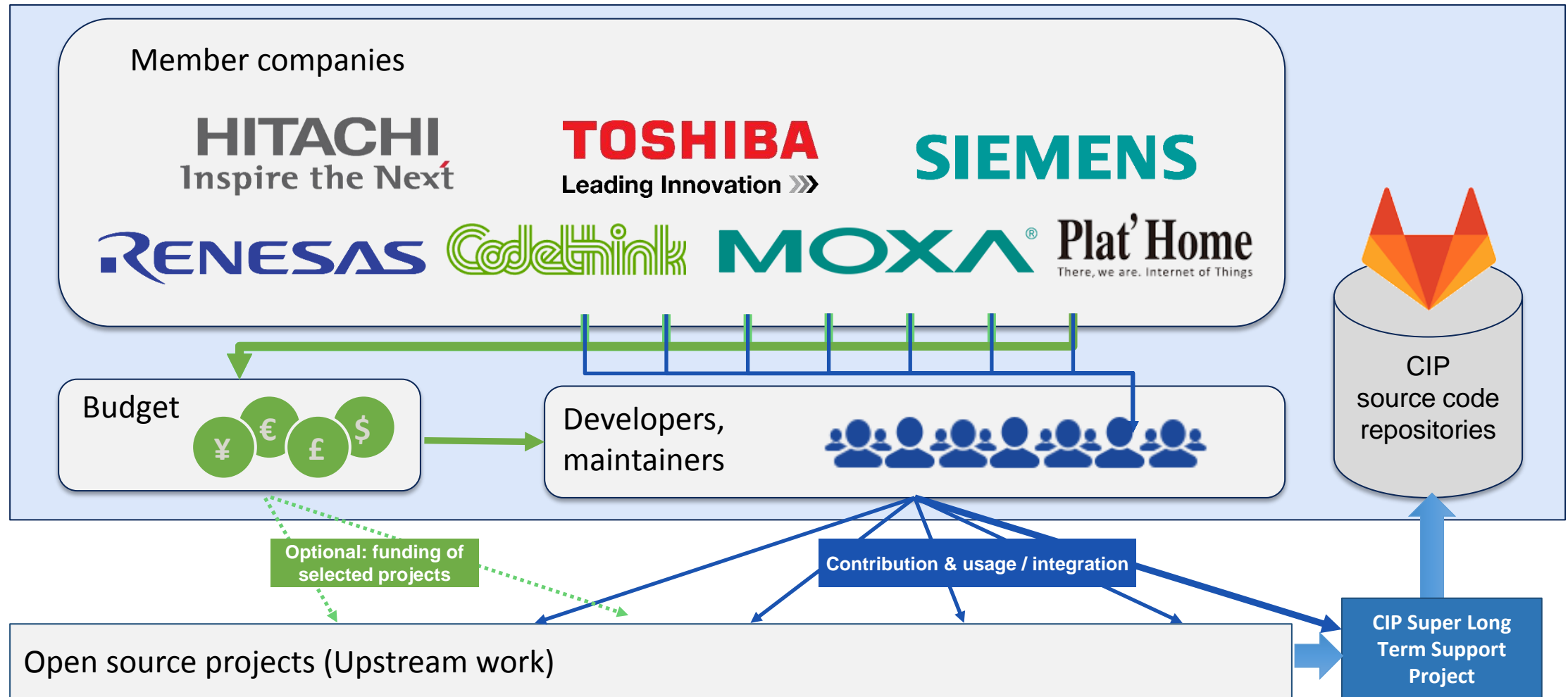


———— CIVIL ————  
**INFRASTRUCTURE**  
———— PLATFORM ————

since April 2016



# The backbone of CIP are the member companies





## 1 Kernel maintenance

- The first action taken by the CIP project is to select and maintain Linux kernels for very long time (+15 years). To achieve goal a group of experts has been assigned.

## 2 PREEMPT\_RT patches are added to the CIP kernel

## 3 Testing

- Civil infrastructure industry has high stability, reliability and security standards in order to ensure safety critical systems. The CIP Testing project has been formed to address this reality. So far the efforts are focused on testing the CIP kernel. In the future they will be extended to the complete CIP platform.

## 4 CIP Core

- This project focus to create reference minimal file system images that allow testing the CIP Core packages: a set of industrial-grade components that require super long-term maintenance.



## 4 Debian as CIP primary reference distribution



- Why Debian?
  - Maturity, focus on stability
  - Professional security properties
  - Licensing hygiene
- What does the primary distribution means?
  - CIP will select CIP Core package from Debian packages
  - CIP started to work with Debian community on LTS topics
    - Project sponsoring: staging repo for security-master
    - LTS Platinum sponsoring
  - CIP supports extended LTS (>5 years)
  - CIP committed to ensure 10 years maintenance for core packages

## 4 CIP Core Packages (1/5)



### Minimal package set for CIP base layer (ongoing discussions)

#### On-device packages set

- |                      |  |
|----------------------|--|
| CIP<br>Kernel        | <ul style="list-style-type: none"><li>• Kernel<ul style="list-style-type: none"><li>• Linux kernel + backported patches</li><li>• PREEMPT_RT patch</li></ul></li></ul>   |
| CIP Core<br>Packages | <ul style="list-style-type: none"><li>• Bootloader<ul style="list-style-type: none"><li>• U-boot</li></ul></li><li>• Shells / Utilities<ul style="list-style-type: none"><li>• Bash</li><li>• Busybox?</li></ul></li><li>• Base libraries<ul style="list-style-type: none"><li>• libgcc</li><li>• glibc</li></ul></li><li>• Security<ul style="list-style-type: none"><li>• OpenSSL</li><li>• OpenSSH, dropbear?</li></ul></li><li>• ...</li></ul> |

#### Packages required for maintenance & reproducible build

- |                 |            |              |              |
|-----------------|------------|--------------|--------------|
| Dev<br>packages | • Flex     | • gcc        | • Make       |
|                 | • Bison    | • Gdb        | • M4         |
|                 | • autoconf | • Git        | • pax-utils  |
|                 | • automake | • Glib       | • Pciutils   |
|                 | • bc       | • Gmp        | • Perl       |
|                 | • binutils | • Gzip       | • pkg-config |
|                 | • bison    | • gettext    | • Popt       |
|                 | • Bzip2    | • Kbd        | • Procps     |
|                 | • Curl     | • Libibverbs | • Quilt      |
|                 | • Db       | • Libtool    | • Readline   |
|                 | • Dbus     | • Libxml2    | • sysfsutils |
|                 | • Expat    | • Mpclib     | • Tar        |
|                 | • Flex     | • Mpfr4      | • Unifdef    |
|                 | • gawk     | • Ncurses    | • Zlib       |

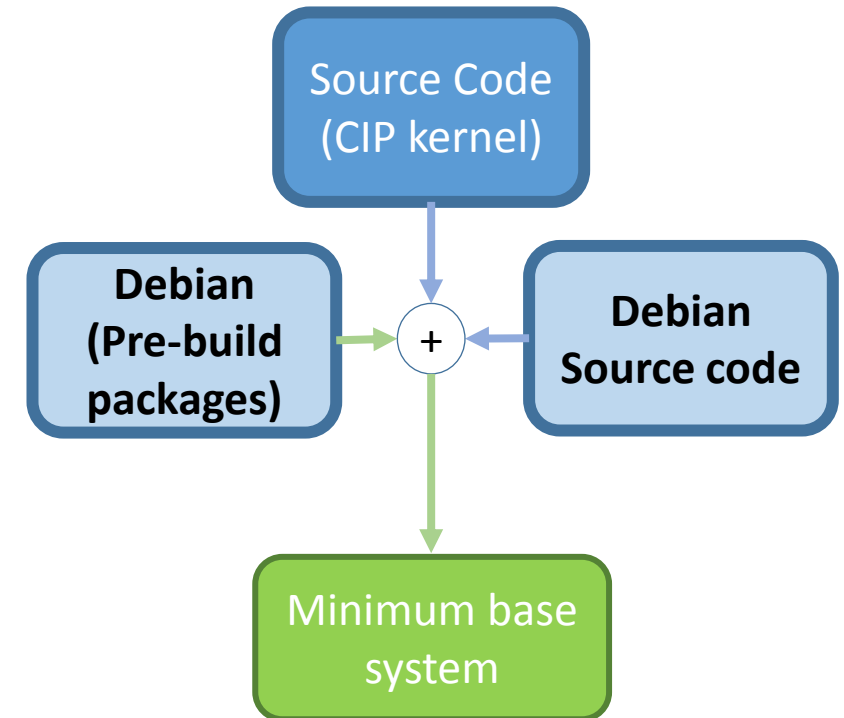
Dimension: 30..300 packages



## 4 CIP Core Packages (3/5)

### CIP Core

- **CIP Core is now become CIP official project**
  - CIP Core aims to provide a way to create and test installable images
- **Goal**
  - **Input:** Debian sources/binaries and cip kernel
  - **Build mechanism:** Bitbake and/or Debian build system
  - **Output:** Minimum deployable base system image for testing
- **Current status**
  - Minimal rootfs can be build for the following hardware
    - Renesas RZ/G1M (iwg20m)
    - BeagleBone Black
    - Cyclone-V
    - QEMUx86



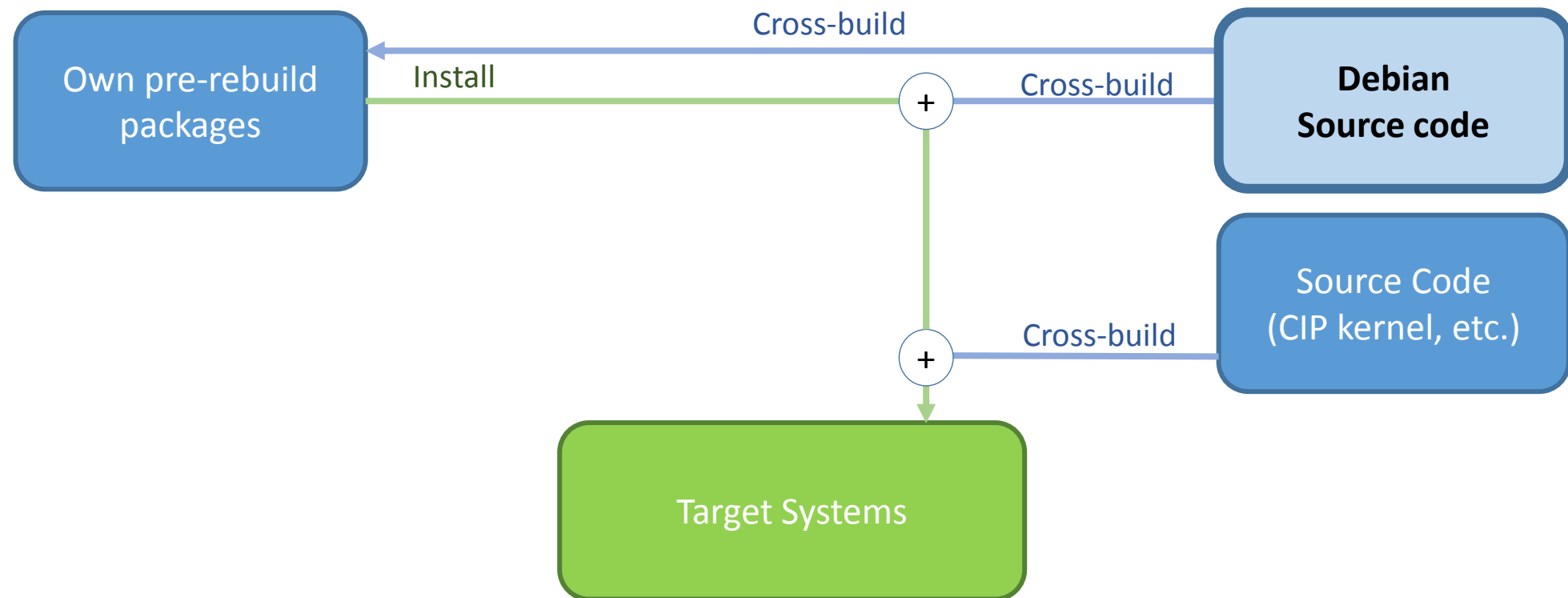
**Source code:** <https://gitlab.com/cip-project/cip-core>

## 4 CIP Core Packages (4/5)



### Creating Debian-based image (Currently supported)

Deby: <https://github.com/meta-debian/meta-debian>





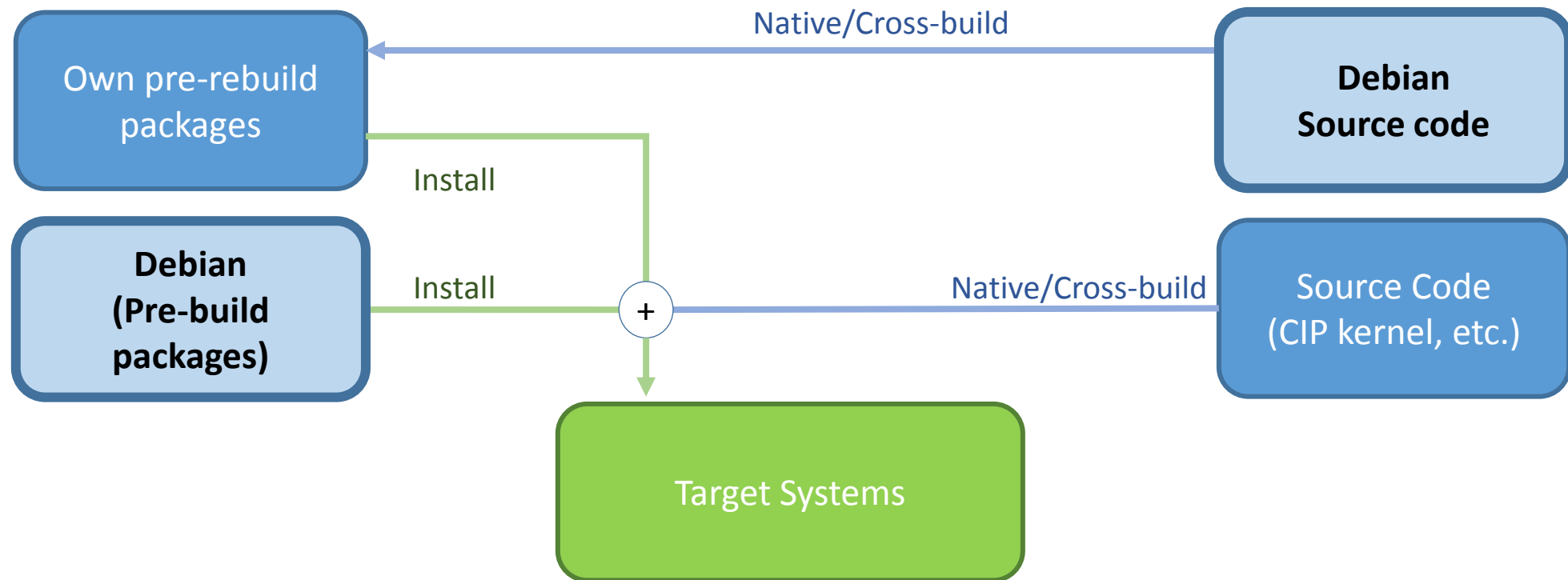
## 4 CIP Core Packages (5/5)



### Creating Debian-based image (Other options)

ISAR: <https://github.com/ilbers/isar>

ELBE: <https://elbe-rfs.org/>



## ④ Gaps and Common Goals between Debian and CIP



Debian	CIP requires	Chance to collaborate with Debian
<p><b>Support</b></p> <ul style="list-style-type: none"><li>▪ Term: 3+2 years by Debian-LTS</li><li>▪ Num of pkgs: 67776</li></ul> <p><b>Build</b></p> <ul style="list-style-type: none"><li>▪ Should support native build</li><li>▪ Working on cross build packaging (Debian-cross)</li><li>▪ Reproducible build</li></ul> <p><b>OSS license compliance</b></p> <ul style="list-style-type: none"><li>▪ DEP-5 adoption is ongoing</li></ul> <p><b>Testing</b></p> <ul style="list-style-type: none"><li>▪ Packages are tested by community</li><li>▪ autopkgtest</li></ul>	<p><b>Support</b></p> <ul style="list-style-type: none"><li>▪ Term: 10+ years</li><li>▪ Num of pkgs: 30..300</li></ul> <p><b>Build</b></p> <ul style="list-style-type: none"><li>▪ Need to have both native and cross build</li><li>▪ Binary / Source code should be managed and reproducible</li></ul> <p><b>OSS license compliance</b></p> <ul style="list-style-type: none"><li>▪ Generate reports automatically</li><li>▪ Easy to redistribute</li></ul> <p><b>Testing</b></p> <ul style="list-style-type: none"><li>▪ All packages should be tested in timely manner</li></ul>	<ul style="list-style-type: none"><li>▪ <b>Longer</b> term maintenance for limited number of packages</li><li>▪ Contributing to <b>Debian-cross</b></li><li>▪ Supporting <b>reproducible build</b> efforts</li><li>▪ Attracting more <b>embedded users</b></li><li>▪ Exchange and share the <b>license review results</b></li><li>▪ Contributing <b>test cases</b> to upstream</li></ul>

# What's currently under discussion in CIP



- Functional safety
- Security standards for industry
  - E.g. IEC62443-4 (-> update strategies, ensuring integrity / “secure boot”)
- Y2038



# Summary

# Summary



- **Our civil infrastructure is too critical to run it on arbitrary software platforms**
- CIP enables and preserves an open source base layer
- **Debian** was chosen **as solid foundation**
- CIP is looking forward to deepen support of and collaboration with the Debian community

# Contact Information and Resources



To get the latest information, please contact:

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

## Other resources

- CIP Web site: <https://www.cip-project.org>
- CIP Wiki: <https://wiki.linuxfoundation.org/civilinfrastructureplatform/>
- CIP source code
  - CIP GitLab: <http://www.gitlab.com/cip-project>
  - CIP kernel: <git://git.kernel.org/pub/scm/linux/kernel/git/bwh/linux-cip.git>





# Thank you!



# Questions?