OSGi Overview

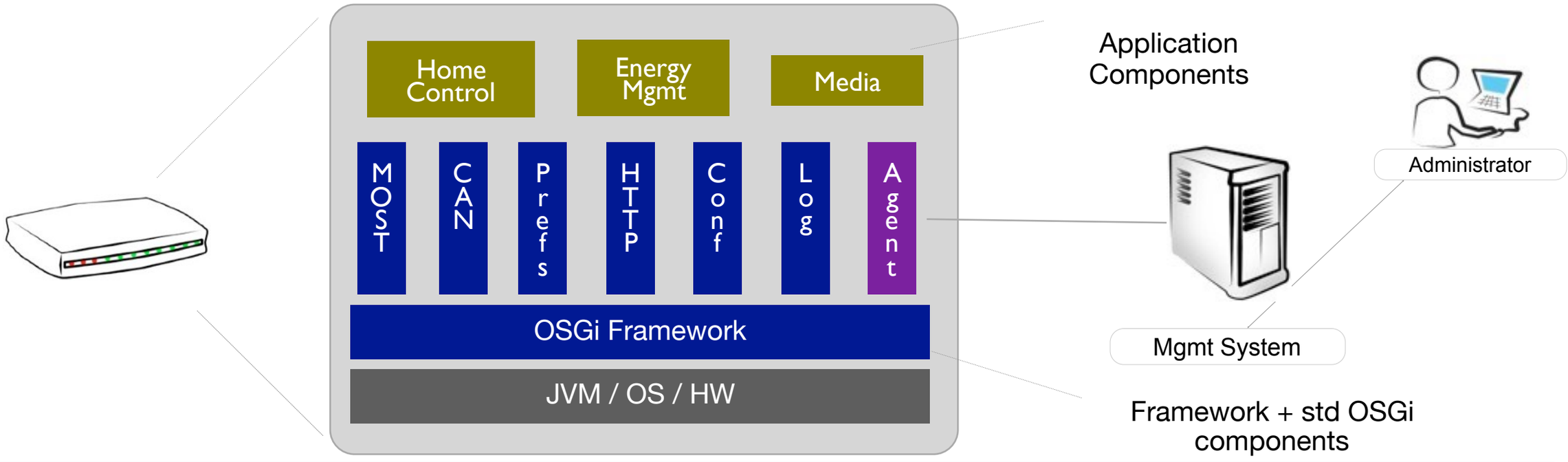
OSGi Alliance and IIC Joint Liaison Workshop

Christer Larsson
VP EMEA OSGi Alliance
CEO Makewave

2018-05-24, Helsinki

What is the OSGi Technology?

- OSGi is a Standardized Software Execution Environment
  - Component based module system defined in Java
  - Open Standard defined by the OSGi Alliance
  - Service oriented & remotely managed (OMA & TR-69)
  - Works like an operating system for small applications called Bundles
  - Ideal for a home gateway, IoT gateway, or similar equipment



The OSGi Alliance

- The OSGi Alliance is worldwide, non-profit consortium
- Creates and controls the OSGi Specifications
- The OSGi Technology is uses in a wide range of open source projects and commercial products for IoT, cloud and enterprise markets.

- OSGi Alliance members include:

The OSGi Alliance?

Founded in 1999

Proven, Mature Software Architecture

Transparent Development Process

Strategic Partnerships/ Collaboration

Global Ecosystem

Best Practices

Industry & End User Adoption

OSGi Alliance Deliverables

- To foster a valuable cross-industry ecosystem, the OSGi Alliance delivers:
  - Specifications
  - Reference Implementations
  - Test Suites
  - Certifications

We are proud to be a **democratic**, **collaborative**, and **non-profit** organization that is operating in a **fully transparent** environment and **open to everyone** to join and contribute.

More Info

OSGi Alliance
Bishop Ranch 6
2400 Camino Ramon,
Suite 375
San Ramon, CA 94583
USA

Phone: +1 (925) 275-6690
Fax: +1 (925) 275 6691

Email: help@osgi.org

Online: www.osgi.org

Twitter: @OSGiAlliance

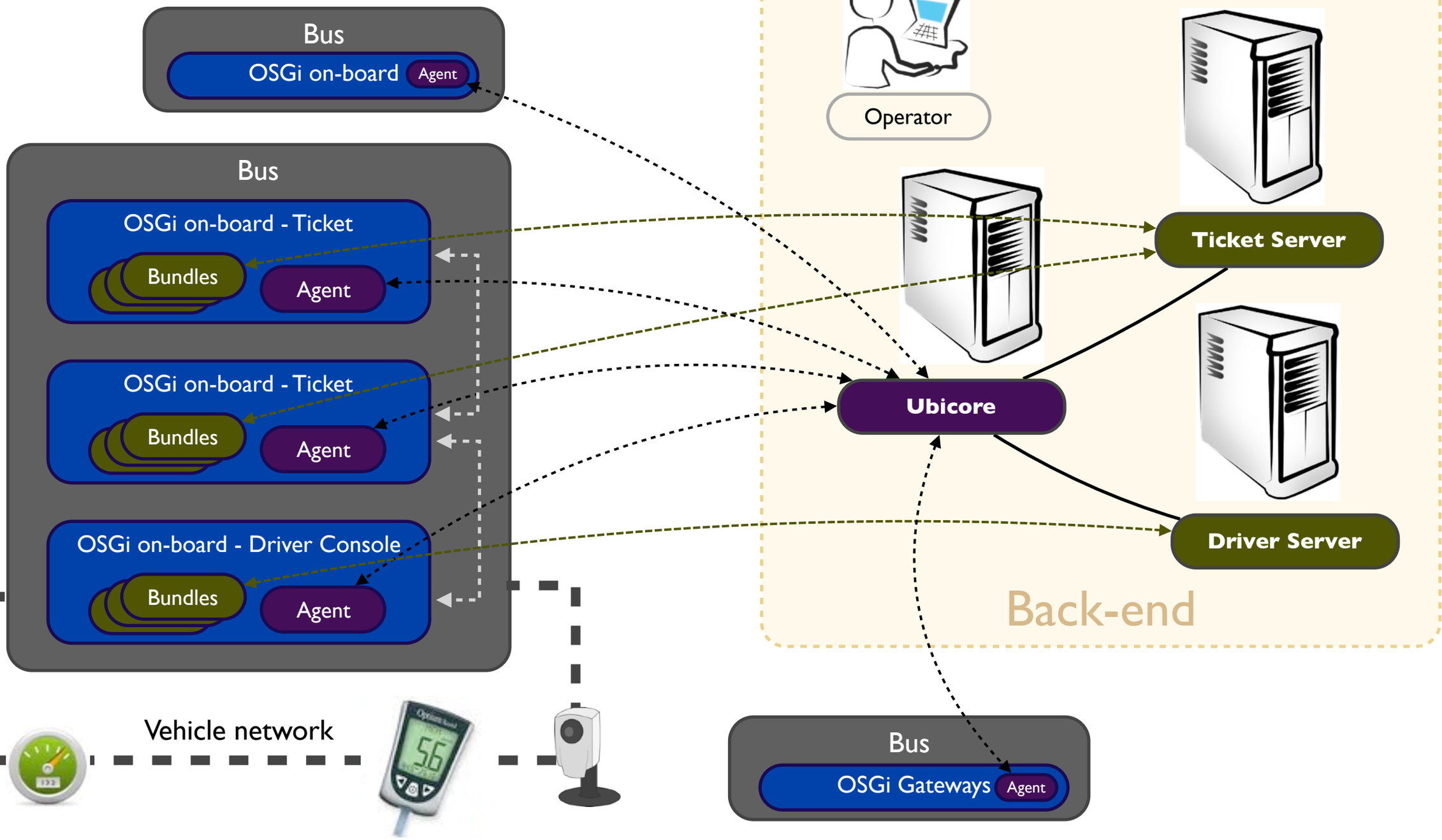
LinkedIn:
https://www.linkedin.com/groups/122461

Real world example of a bus fleet system based on OSGi

- An example of a bus fleet system based on Makewave's OSGi Technology
  - Knopflerfish OSGi edge device stack
  - Ubicore - a remote management system for edge devices
- ~5 000 buses, edge bus having ~5 edge devices running OSGi.

System Architecture - bus fleet

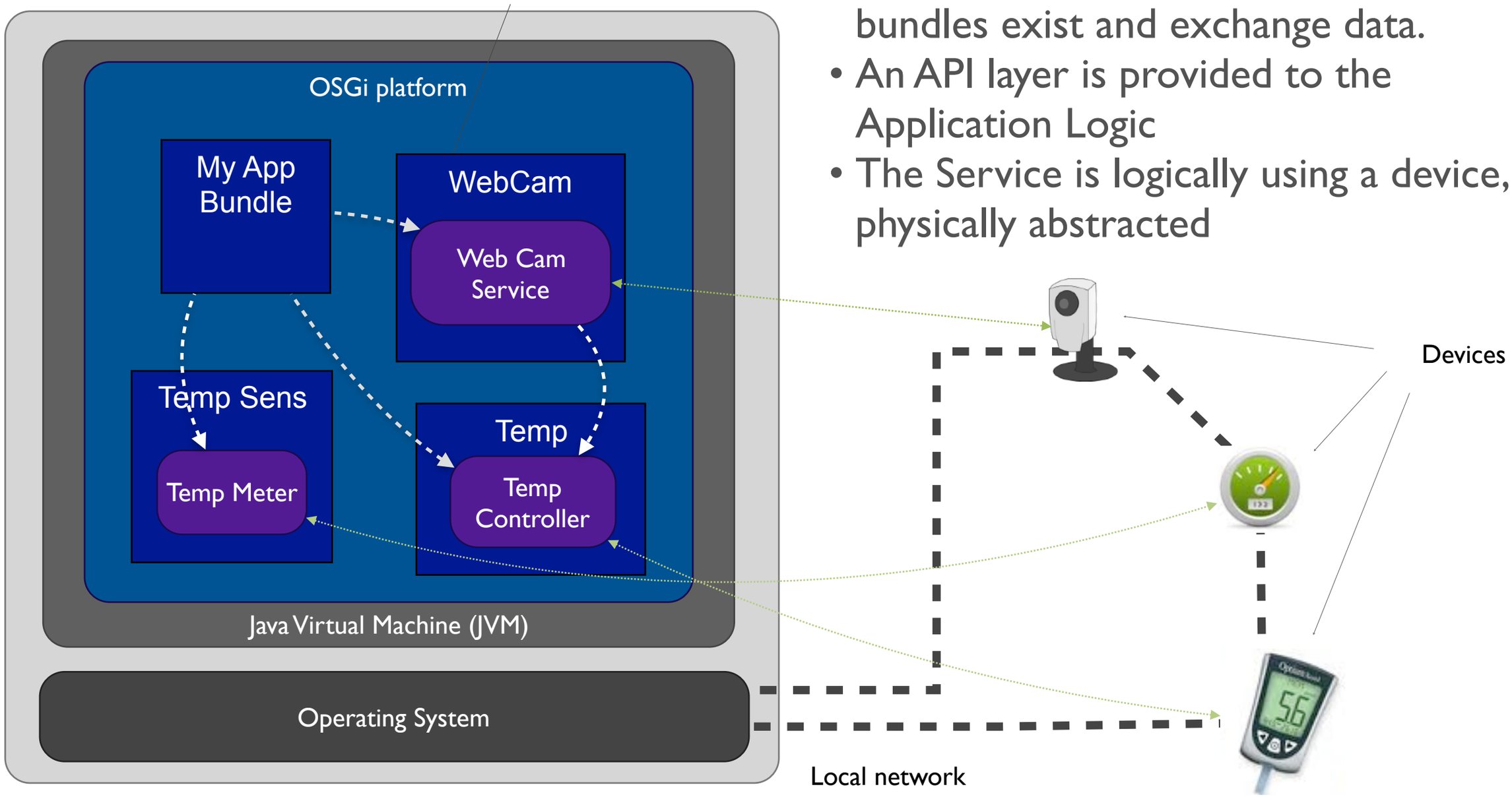
All buses are equipped with one or more OSGi based on-board computers / devices. Each OSGi platform has a management agent which connects it to Ubicore



System Architecture - bus fleet

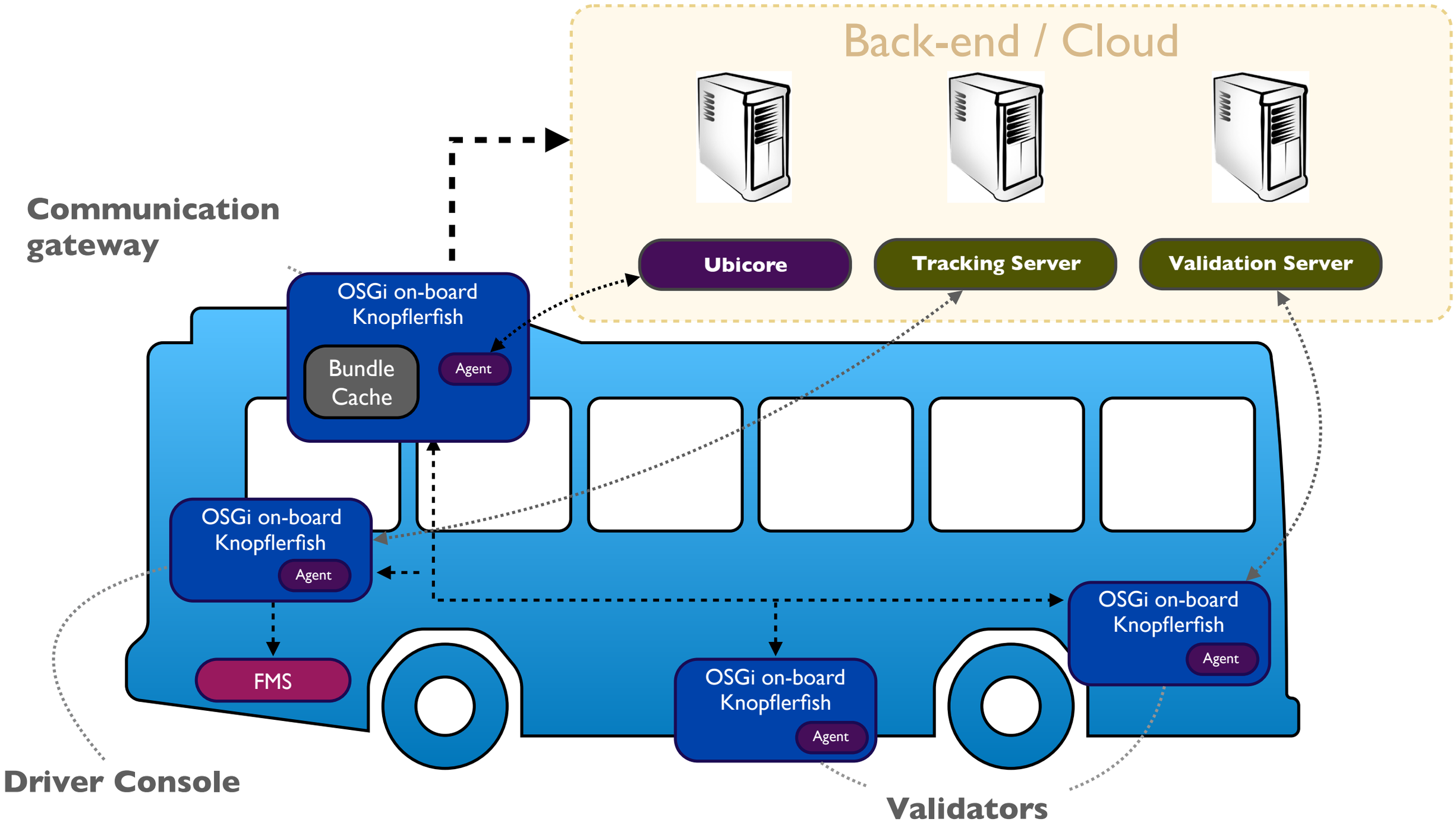
- Every bus has one or more on-board computers that has an embedded OSGi platform (Knopflerfish)
- The on-board computers are connected to the in-vehicle network (FMS)
- The on-board computers are connected to each other (in-vehicle IP)
- Every on-board computer has an OSGi management agent. The agent is responsible for the connection to Ubicore and performs mgmt tasks decided by Ubicore.
- All business logic is implemented as OSGi bundles. The business logic bundles are all managed via Ubicore.
- The business logic bundles sends / receives data directly to/from its corresponding back-end server. I.e. the IP traffic is not routed via Ubicore
- The Ubicore server is part of the complete back-end server infrastructure. It is integrated with other parts via a REST API.

OSGi is an embedded integration platform



- OSGi provides a sandbox in which bundles exist and exchange data.
- An API layer is provided to the Application Logic
- The Service is logically using a device, but physically abstracted

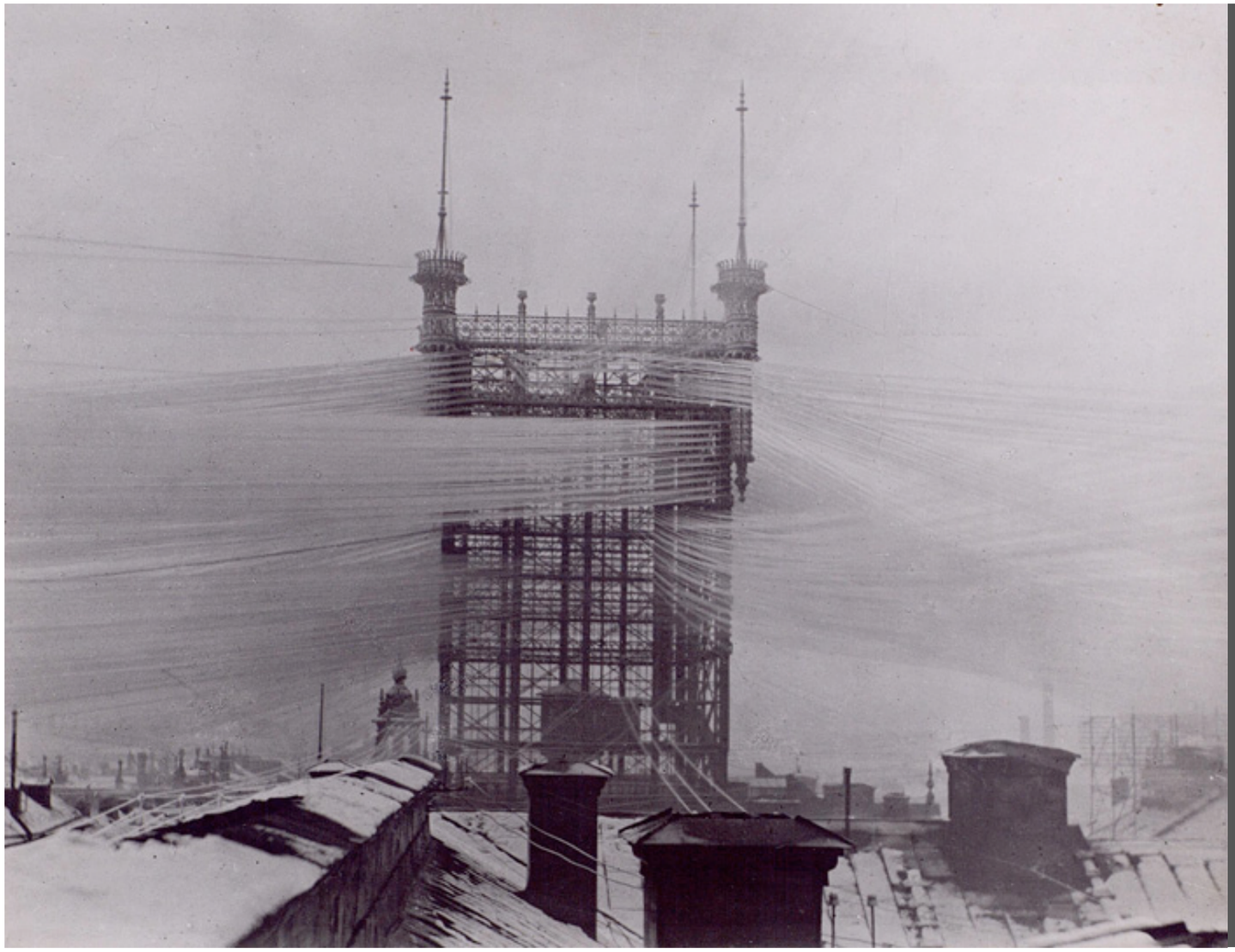
Inside one vehicle



Benefits with an OSGi solution

- Clear separation between business logic (implemented as OSGi bundles) and lower lever parts device code (C/C++).
- Uniform architecture - exactly the same business logic bundles can be used regardless of on-board computer architecture (ARM, X86). No need to recompile, or support different versions.
- Remotely Managed and Flexible - it is very easy to add, remove or update functionality in the business logic layer over time.

Thank you!



Christer Larsson
CEO Makewave
www.makewave.com

VP EMEA OSGi
www.osgi.org

Knopflerfish OSGi
www.knopflerfish.org