Open Distributed Edge Computing

Dr. Dirk Kutscher

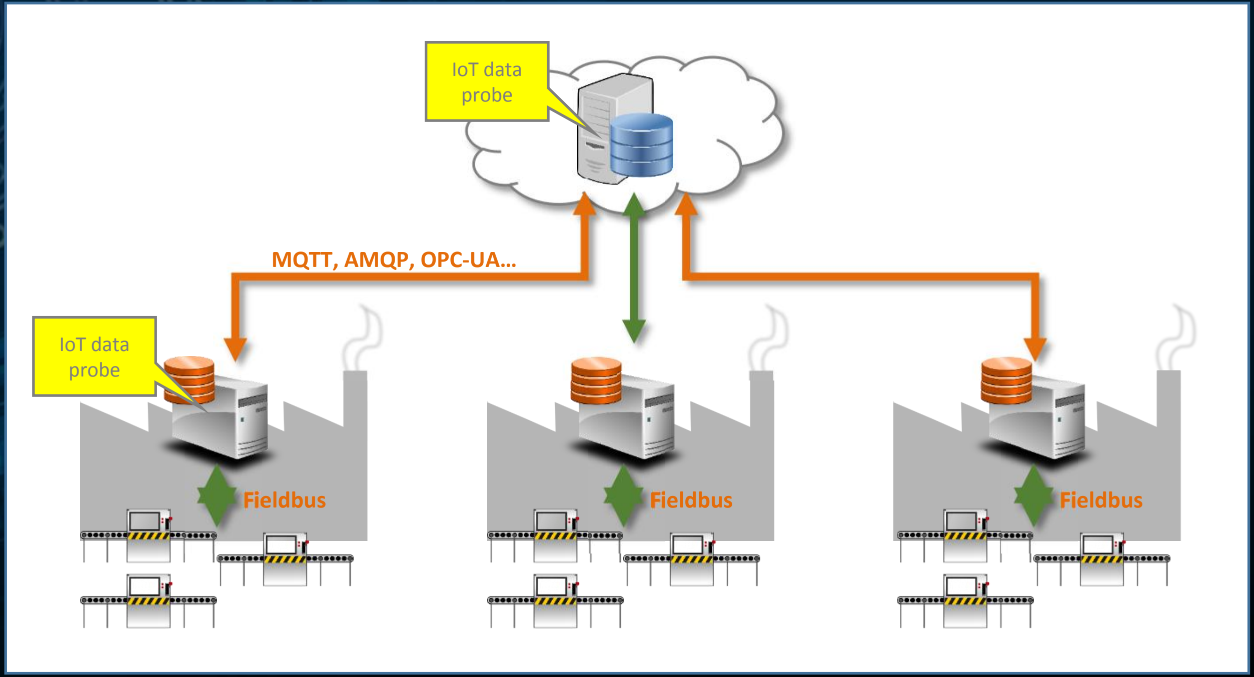
Huawei German Research Center

Computing at the Far Edge

- Mainstream edge computing: extending cloud computing to the edge
  - Virtualized platforms as execution environments
  - Extending orchestration to edge platforms
  - Communicating in virtual circuits (HTTP/TLS/TCP)
- Some edge computing use case have different requirements
- Industrial IoT
  - Ubiquituous data sharing and processing at the factory edge
- Car communication
  - Dynamic interaction with the smart city environment

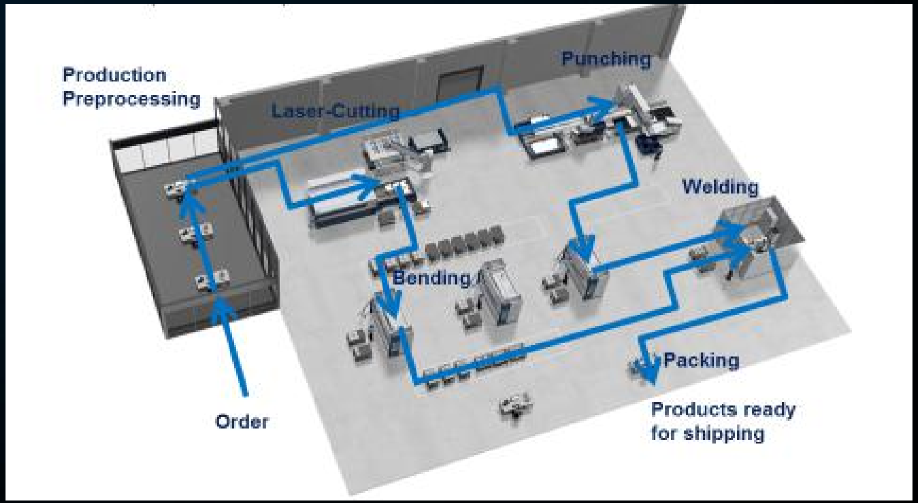
Industrial IoT

1. Data generation in fieldbus networks
   - TSN/Profinet/CAN etc.
   - Edge gateways with application layer data sharing (OPC UA, DDS etc.)



2. Massive real-time data processing
   - Video feeds
   - Real-time analytics
   - Real-time control loops

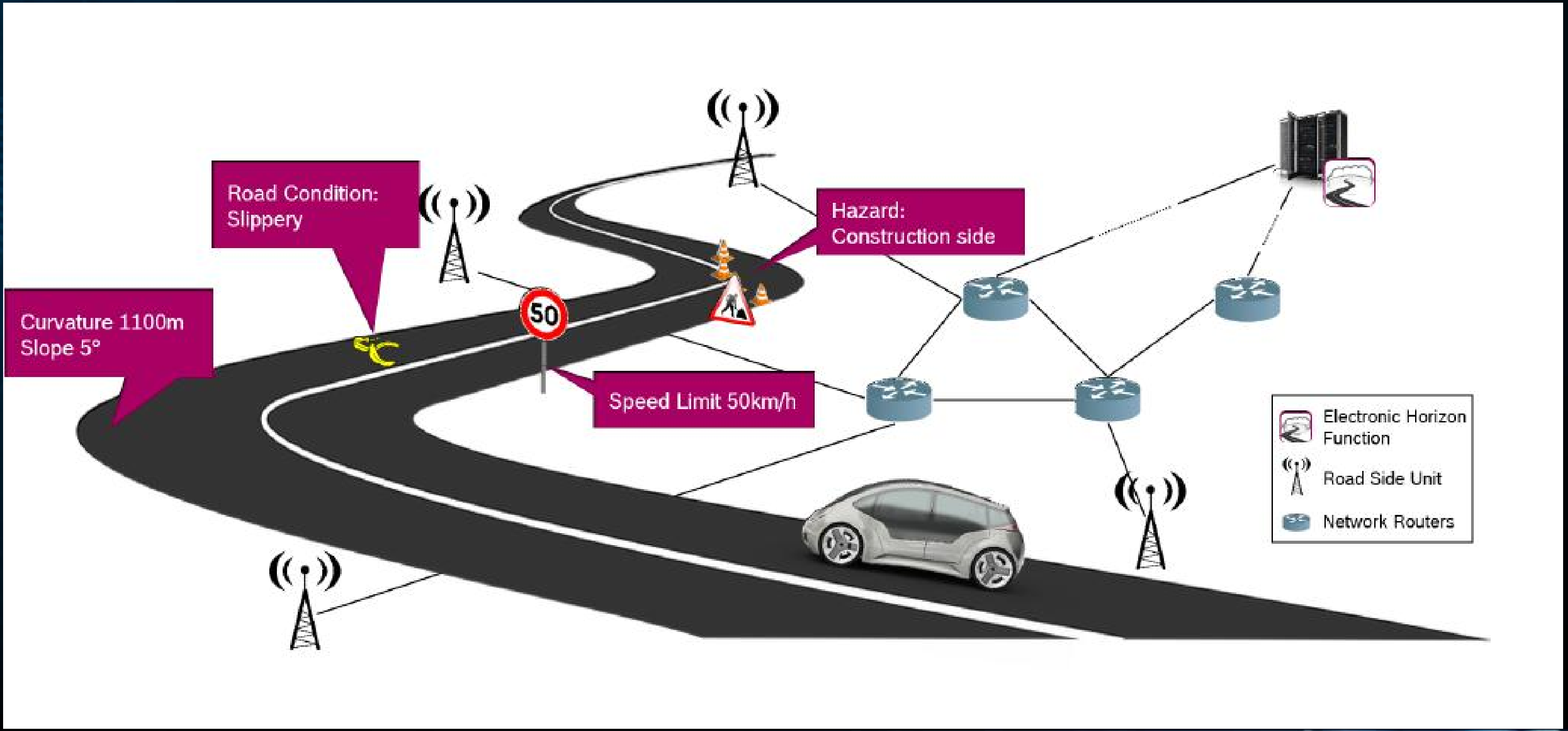
Chained machines in smart factories



www.trumpf.com

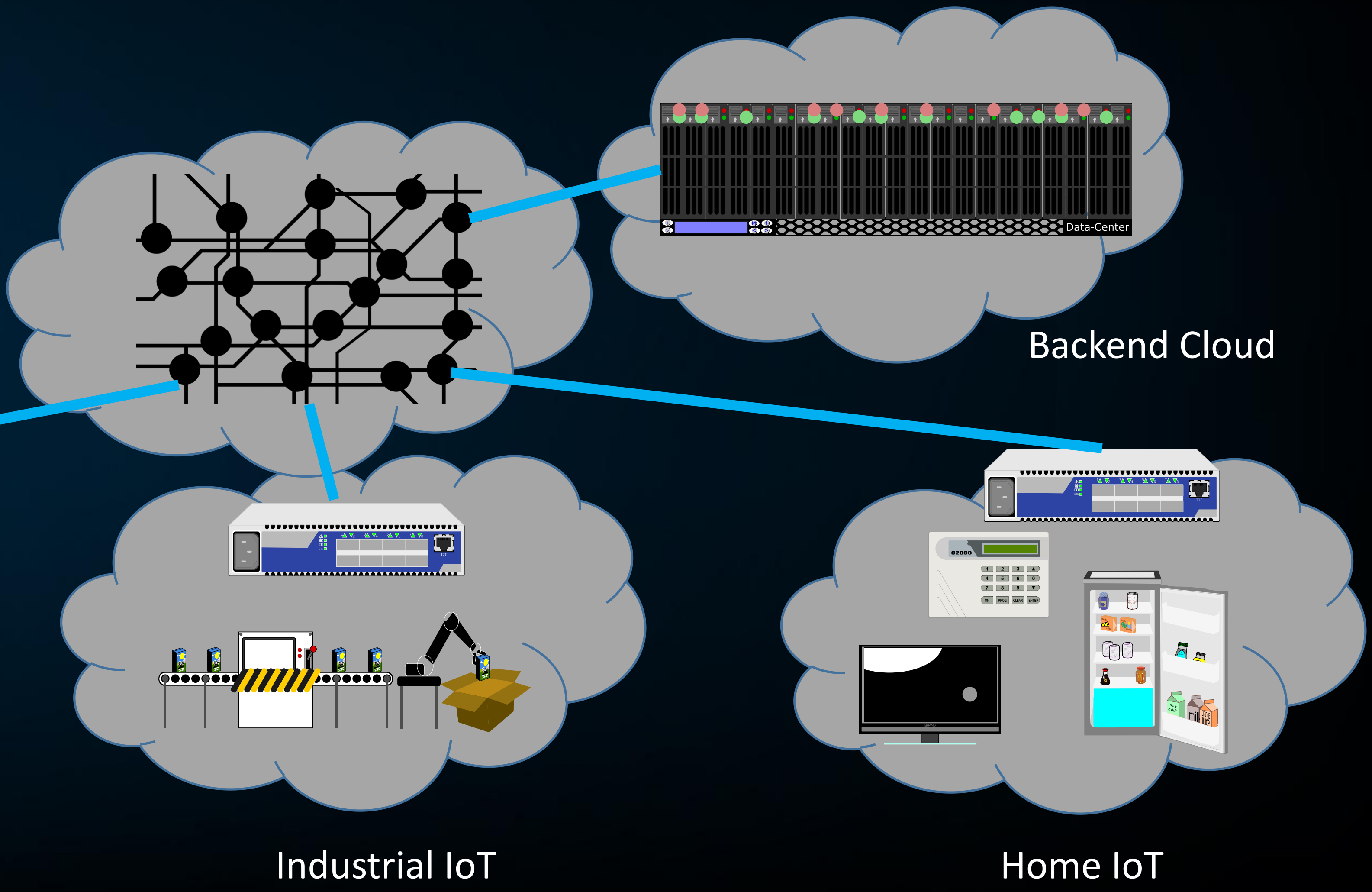
Smart City Vehicular Communications

- Electronic horizon use case: dynamic interactions in the smart city environment
  - Edge computing in the smart city infrastructure and on vehicles
  - Mobility, low-latency, security



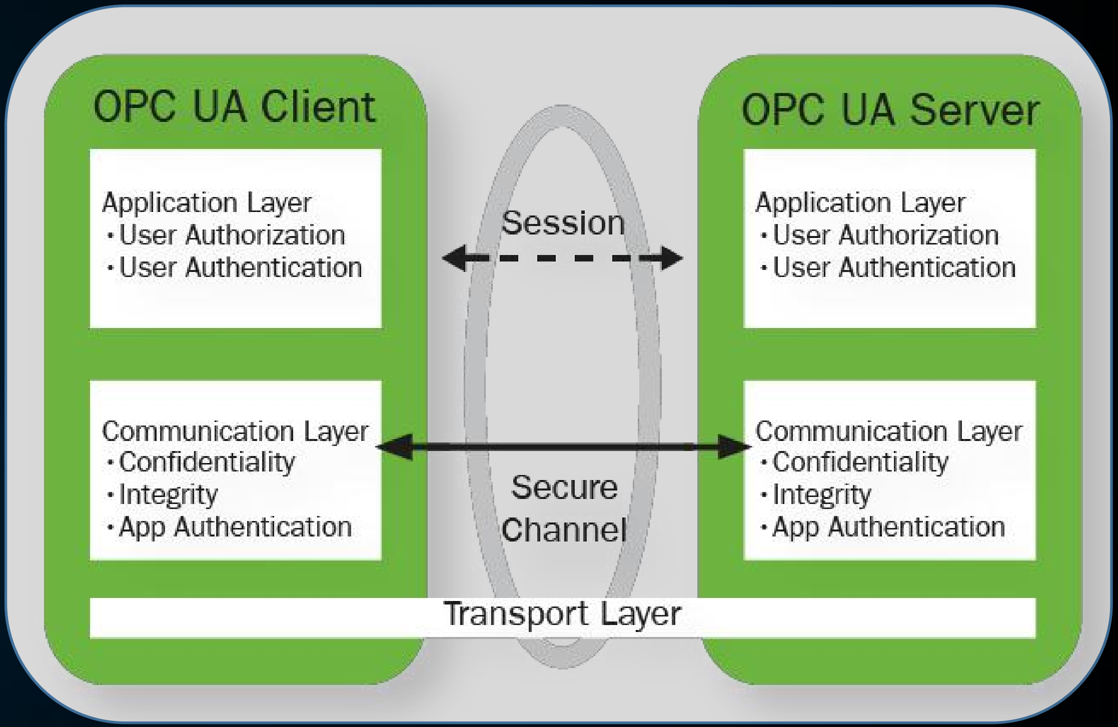
Data Inversion & Silo Problem

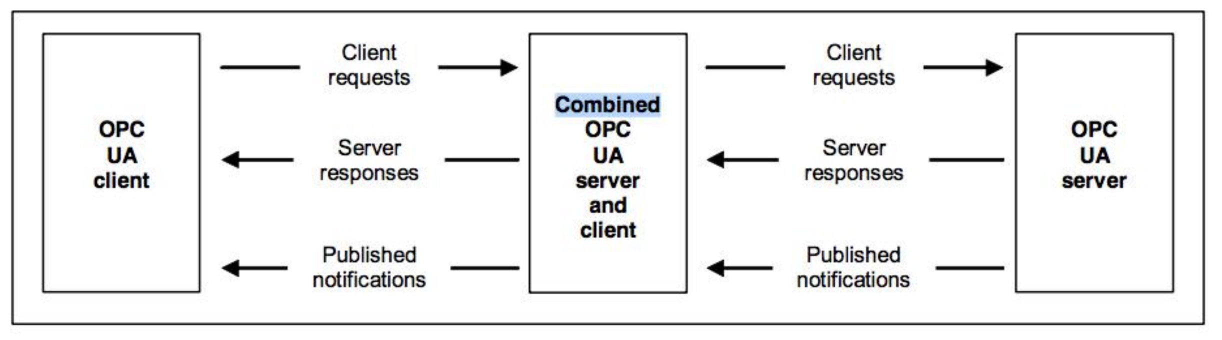
- IoT data flows upstreams
  - Network infrastructure optimized for downstream consumption
- Cloud-based model not always optimal
  - Trust, latency, scale



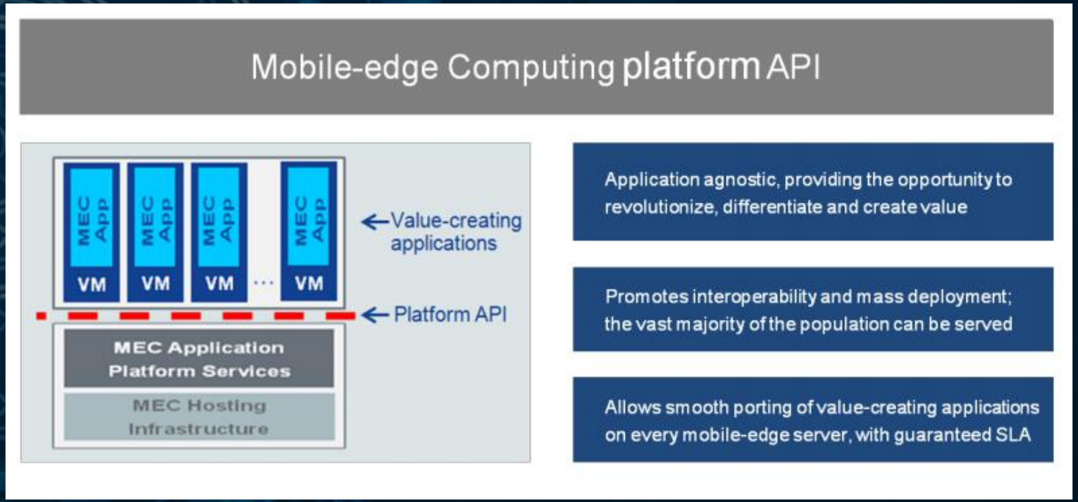
Data-oriented Communication

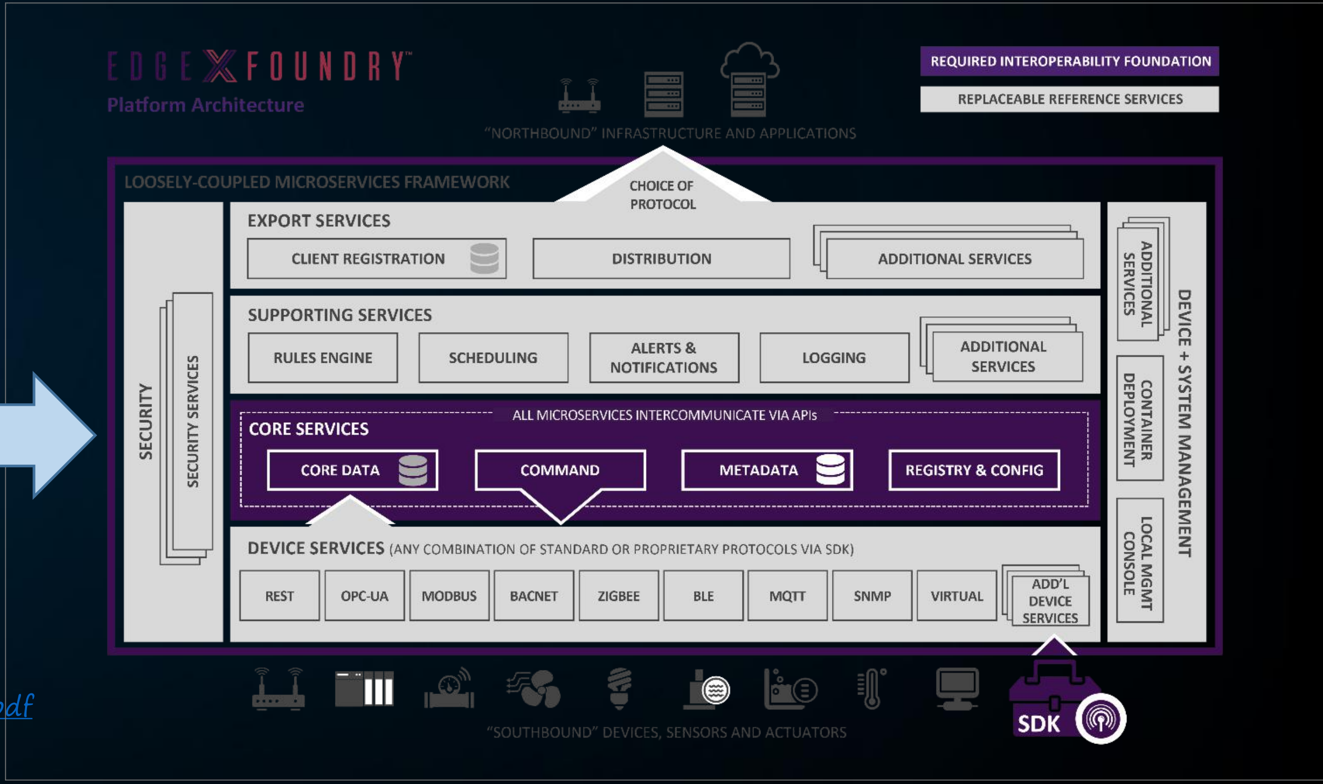
- Several application-layer frameworks
- Data-oriented communication (accessing named data on a server, publish-subscribe etc.)
- Communication inside TLS-secured connections
  - Data sharing difficult
  - Limited scalability
  - Potentially very inefficient
- Not designed for enterprise access control & communication policing
  - NAT & firewall traversal





From Coarse-Grained Telco EC to Microservices





https://www.ietf.org/proceedings/97/slides/slides-97-nfvrg-08-introducing-mec-00.pdf

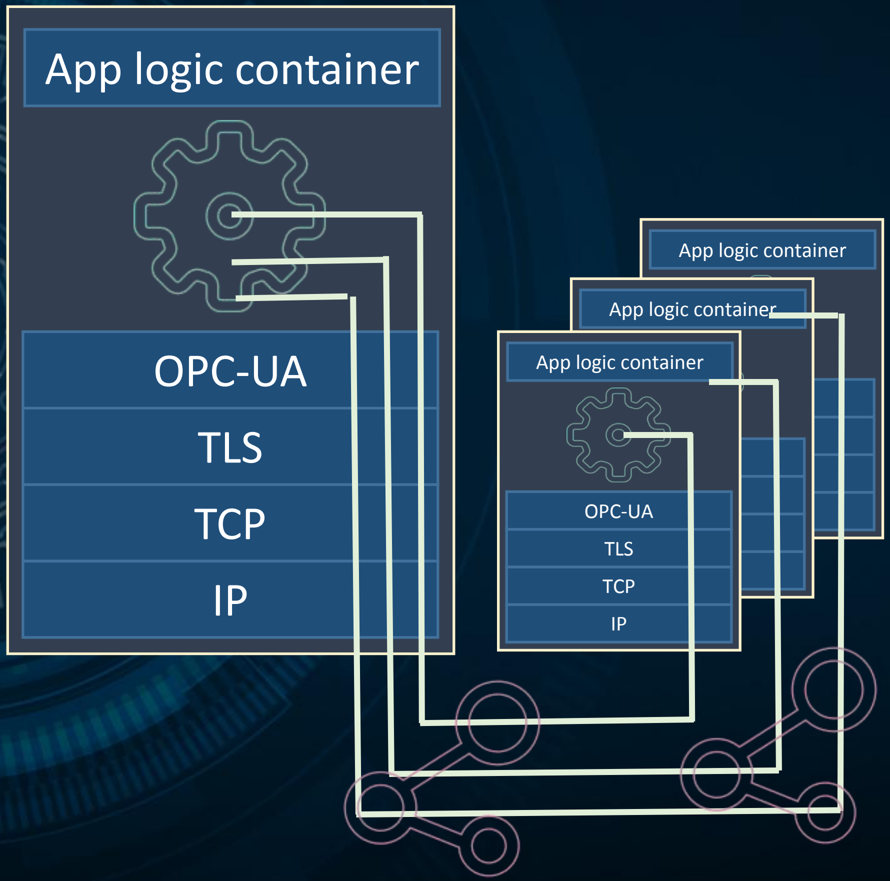
Open Distributed Edge Computing

- Efficient and secure data-oriented networking
  - Accesing named data directly – not inside application layer silos
  - Securing data – not connections
  - Optimizing availability through flexible forwarding and caching in the network
- Light-weight edge computing
  - Moving functions to data and vice versa
  - Microservice framework for secure distributed computation
  - Avoiding VM/container ossification
- Openness
  - Supporting multiple applications and tenants
- Decentralized operation
  - Avoid cloud dependency
  - Trust management, AAA, name resolution and similar functions in the local edge

Technical Challenges for Distributed Edge Computing

- Communication
  - Secure data sharing at the edge
  - What is the right layering?
  - What is the right security approach?
- Computation
  - What is the right computation abstraction?
  - Virtual machines vs. smaller units of computation
- Semantic Interoperability
  - How to learn about computation services?
  - How to combine edge computing function into a larger application context
- Security & Trust
  - How to manage access control, authorization?
  - How to trust compute functions, mobile code – and their computation results?
- Fine-granular data access, authorization and computation
  - Database-like transaction in edge networks
  - Secure projection, filtering, joining of data sets from multiple sources

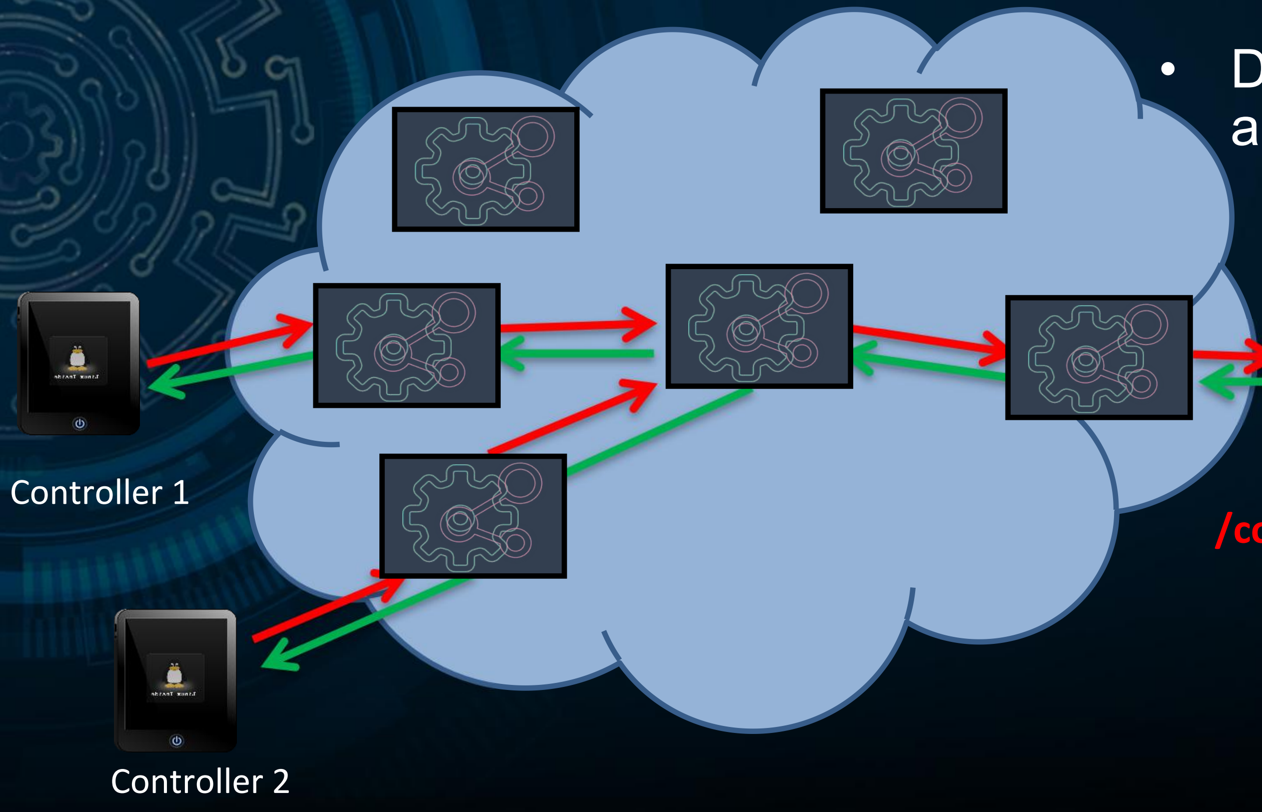
Data-oriented Communication
Overlays and Application-Layer Silos

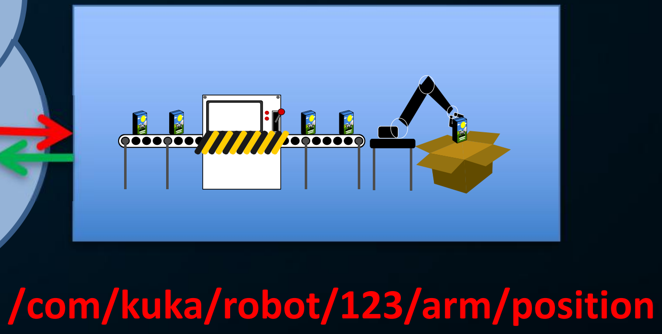


- Overlays
  - Connection-based security
  - Client-server / broker-based
- Limited scalability
  - Pub-sub distribution to many clients through single-server bottleneck
- Limited efficiency
  - Cannot share data directly
- Limited performance and robustness
  - Network cannot assist data dissemination

Data-oriented Communication
Example: Information-Centric Networking

- Accessing Named Data Objects
- Data-centric security approach





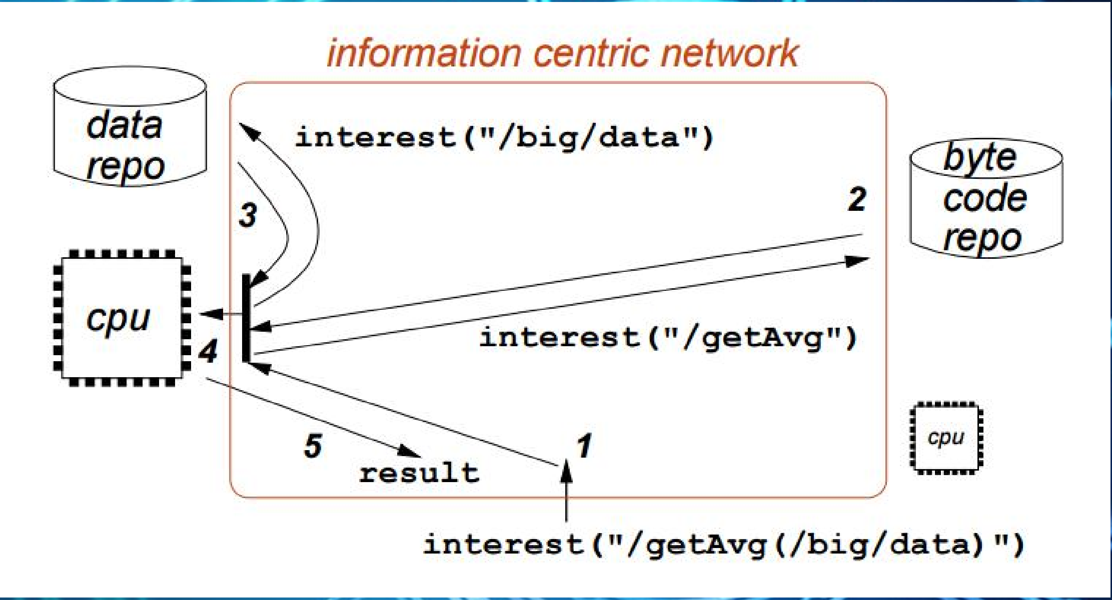
- Ubiquitous Caching
- Per-Hop forwarding strategies
- Automated access control for named data
- Efficient multicast & pub-sub supported by the network

Named Function Networking

ICN: Accessing named data in the network
- Securely
- Both static and dynamic (e.g., live stream)

Challenge: How to achieve dynamic computation?
- With similar security properties?
- ... And automatic function placement?
- Think: edge computing, big data, stream processing, service chaining

Named Function Networking
- /getAverage(/roomA/temp, /roomB/tmp)
- Apps specify desired results
- Networks finds data and functions – and execution locations
- Results can be cached, just like in regular ICN



Christian Tschudin; **Named Function Networking**
Service chaining, big picture, division of labor boundaries
https://www.ietf.org/proceedings/interim-2015-icnrg-01/slides/slides-interim-2015-icnrg-1-13.pdf

Security and Trust Management

- Security in a data-oriented framework has specific opportunities and requirements
  - Authenticating/encrypting data directly
  - Verifying data producer provenance
- Challenges
  - Establishing trust anchors in a distributed system
  - Automating authentication & authorization for fine-granular data communication and edge computing

Decentralized Trust and Identity Management

- Why decentralized?
  - No trust in centralized functions
  - Avoid „rent-seeking“ opportunities
  - Easier to operate!
  - Latency!

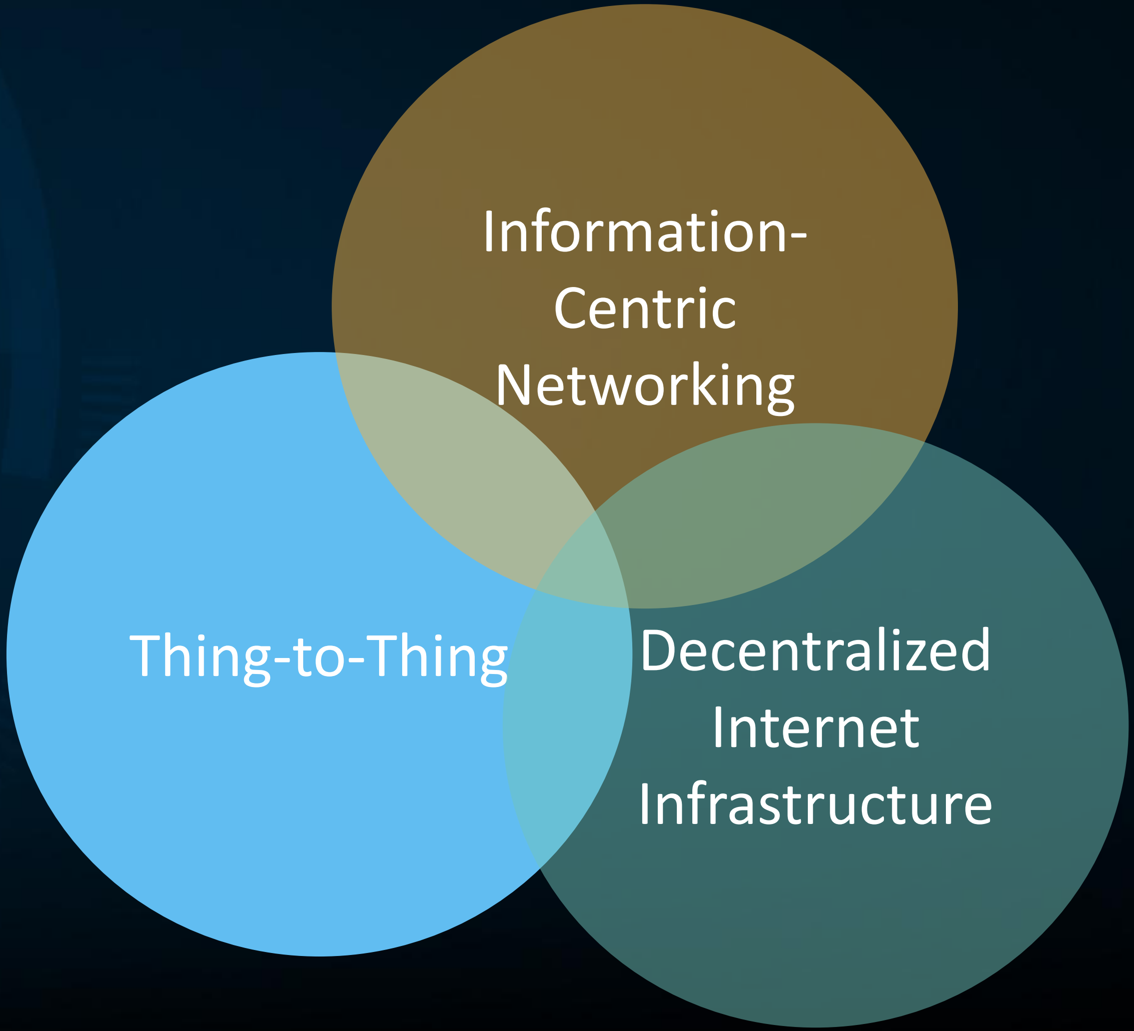
- Reputation systems
  - Users of information can assess its quality
  - Make available signed, anonymous reputation systems
  - Authorization based on reputation

- Data sharing
  - Finding data resources
  - ... And storage
  - Micropayments for data and data services

- Local Processing
  - Find a local „data center“ within 1 ms RTT
  - Obtain attestation of services
  - Authority-based or reputation-based
  - Help creating services with smart contracts

- Mobile code
  - Can‘t upgrade everything at the same time
  - Many functions can be solved in mobile code
  - Running in IoT devices (!)
  - Running in local data bus
  - Controlled resource usage

Internet Research Task Force (IRTF)



Summary: Open Distributed Edge Computing

- Industrial IoT and dynamic smart city environments have specific edge computing requirements
  - Efficient and secure data-oriented communication
  - Dynamic computation at the edge
  - Decentralized, cloudless operation
  - Application independence and multi-tencancy
  - Open platforms and protocols
- On-going work in IRTF and IETF
- Intel/NSF-sponsored projects on Information-Centric Wireless Edge Networking