Industrial Internet of Things (IIoT)

Prof. Dr.-Ing. Dipl.-Wirt. Ing.
Diethelm Bienhaus

"Preample"
about me and my university

University of Applied Sciences Mittelhessen

- "University of Applied Sciences"
  - focus on application of science / cooperation with industry
- 3 sites
- > 18.000 students
- > 3.000 foreign students
- > 300 professors
- Where?
  - Giessen
  - Friedberg

About me

- 1993 Degree in Electrical Engineering
  - Thesis: Simulation of Production Lines
- 1999 PhD in Electrical Engineering
  - Focus: Design Patterns for Distributed Systems (Application Area: Virtual Reality, Augmented Reality, Robotics)
- 2000 … 2014 Professor University of Cooperative Education Northhesse
- 2014 … now Professor at University of Applied Sciences Middlehesse

- Name?
  - "Diethelm" ~ "Protector of the People"
  - "Bienhaus" ~ "House of Bees"
  -
  - since 12th century in my home town "Battenberg"
  - → "Mountbatten" English royals
  - State Hessen – County of Hesse
  - → Александра Фёдоровна (жена Николая II)

(Industrial) Internet of Things Overview

Some definitions ...

- Internet of Things
  - ":the networking capability that allows information to be sent to and received from objects and devices (such as fixtures and kitchen appliances) using the Internet" (Webster)
- Industrial Internet of Things
  - "The industrial internet of things (IIoT) refers to interconnected sensors, instruments, and other devices networked together with computers' industrial applications, including manufacturing and energy management. This connectivity allows for data collection, exchange, and analysis, potentially facilitating improvements in productivity and efficiency as well as other economic benefits." (Boyes et al.)
- CPS
  - "Cyber-Physical Systems (CPS) are integrations of computation and physical processes. Embedded computers and networks monitor and control the physical processes, usually with feedback loops where physical processes affect computations and vice versa." (Lee) Good resource: leeseshia.org
- CPPS
  - "CPPS consist of autonomous and cooperative elements and sub-systems that are getting into connection with each other in situation dependent ways, on and across all levels of production, from processes through machines up to production and logistics networks." (Monostori)

"Industry 4.0"

The term “Industrie 4.0” was introduced in 2011 by the Communication Promoters Group of the Industry-Science Research Alliance to describe the widespread integration of information and communication technology in industrial production. The “4.0” alludes to how this trend’s potentially revolutionary impact follows directly in the footsteps of the three previous industrial revolutions. Since 2011, a number of initiatives addressing the theme of digitally connected industrial production have sprung up around the world, for example the Industrial Internet Consortium in the US and the Industrial Value Chain Initiative in Japan.

https://i40mc.de/wp-content/uploads/sites/22/2016/11/acatech_STUDIE_Maturity_Index_eng_WEB.pdf

Индустрии 4.0

Термин «Индустрия 4.0» используется с 2011 года для описания обширного применения информационно-коммуникационных технологий в промышленном производстве. Однако недостаточно рассматривать разработки, связанные с Четвертой промышленной революцией, исключительно с технологической точки зрения — компаниям также нужно преобразовывать свою организацию и культуру. Хотя передовые технологии действительно делают возможным доступ к гораздо более широкому диапазону данных, способность реализовать потенциал, лежащий в основе этих данных, в большой степени зависит от организационной структуры и культуры компании.

Конечная цель — стать постоянно развивающейся, гибкой компанией, готовой быстро и непрерывно адаптироваться к меняющейся среде. Индекс зрелости Индустрии 4.0 предоставляет компаниям руководство по осуществлению этой трансформации. В нем представлена модель развития из шести этапов. Достижение очередного этапа дает дополнительные преимущества.

https://i40mc.de/wp-content/uploads/sites/22/2016/11/acatech_STUDIE_Maturity_Index_rus_WEB-1.pdf

Connectivity as a main enabler

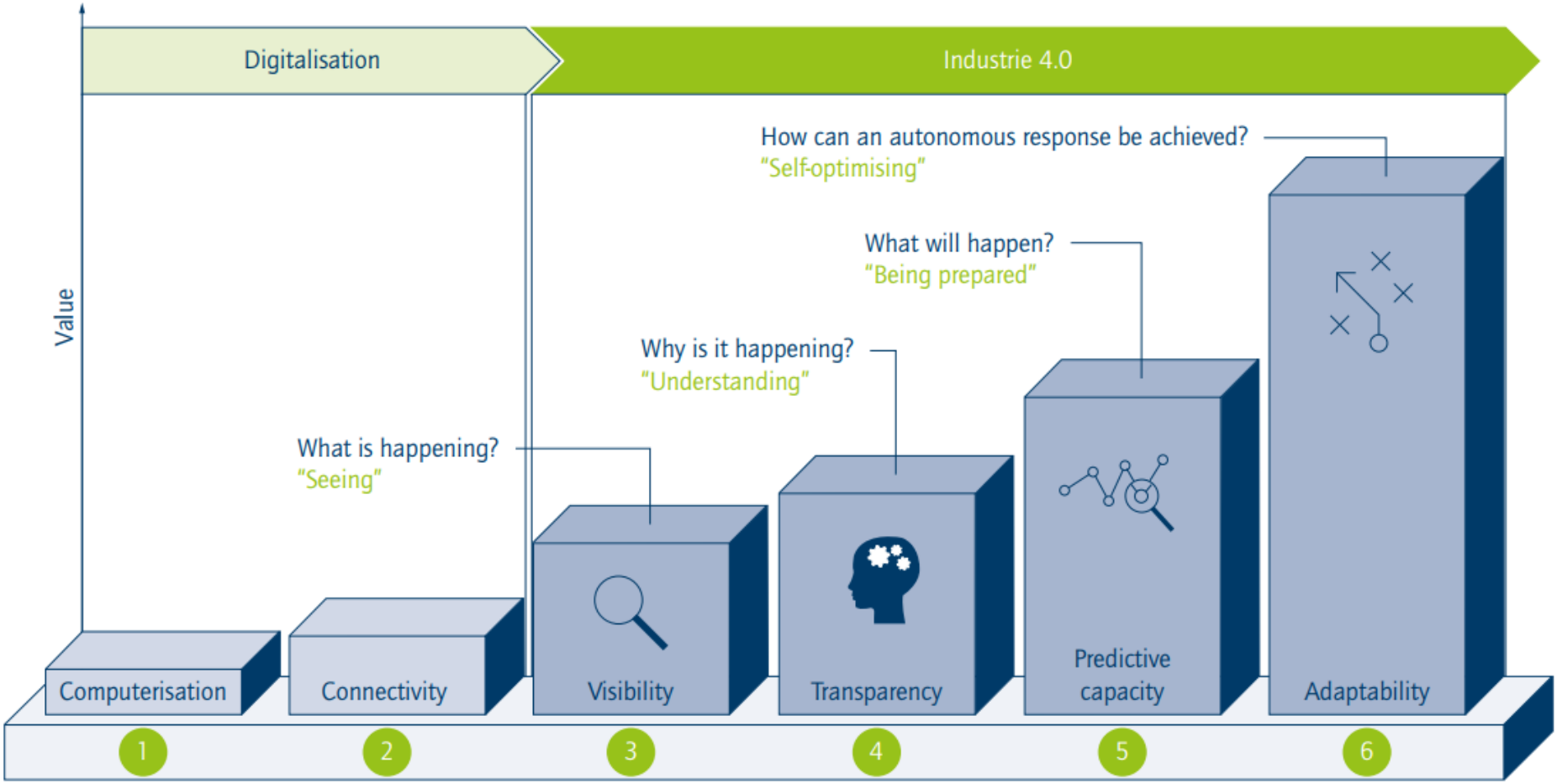


Figure 5: Stages in the Industrie 4.0 development path (source: FIR e. V. at RWTH Aachen University)

Connectivity as a main enabler

Рис. 5. Этапы на пути развития для Индустрии 4.0 (источник: Институт управления промышленной деятельностью (FIR) при Рейнско-Вестфальском техническом университете Ахена)

Standardise data interfaces

To always ensure an agile information flow and data exchange transition when switching from one information system to another, it is necessary to implement neutral or standard interfaces and data exchange formats across all the relevant systems.

These should use an open standard that is universal for the application category in question. One example of a standard interface that is particularly popular in the field of Industrie 4.0 is the OPC-UA architecture for machine-to-machine interfaces.

Ключевым условием для создания единого источника

достоверных данных является подключение всех ИТ-систем друг к другу посредством стандартных

интерфейсов данных вместо централизованного хранения всех данных.

В них должен использоваться открытый стандарт, универсальный для соответствующей категории применения. Одним из примеров стандартного интерфейса, особенно популярного в сфере Индустрии 4.0, является архитектура OPC-UA для интерфейса типа «машина-машина».

Machine-to-Machine (M2M) Interfaces
интерфейса типа «машина-машина»

- Why is HTTP not enough?

- the HTTP standard revolutionized how people consume data
  - a single simple model: Send a request, read the response
  - multiple end user devices: tablet, laptop, smart phone, PC, …
  - view and well known data sources (server) for many data consumers (clients)

- machine-to-machine communication has different challenges
  - "event-oriented paradigm"
  - emitting information one to many
  - listening for events whenever they happen
  - distributing minimal packets of data, but in huge volumes
  - pushing information over unreliable networks

Machine-to-Machine Communication

- Request-Responce / one to one
  - IoT: REST interface
  - IioT: **OPC UA**
  - ...

- Publisher-Subscriber / one to many
  - IoT/IIoT:
    - Advanced Message Queuing Protocol (AMQP)
    - Message Queuing Telemetry Transport (**MQTT**)
    - Constrained Application Protocol (CoAP)
    - …

Message Queuing Telemetry Transport (MQTT)

“A light weight event and message oriented protocol allowing devices to asynchronously communicate efficiently across constrained networks to remote systems” (OASIS)

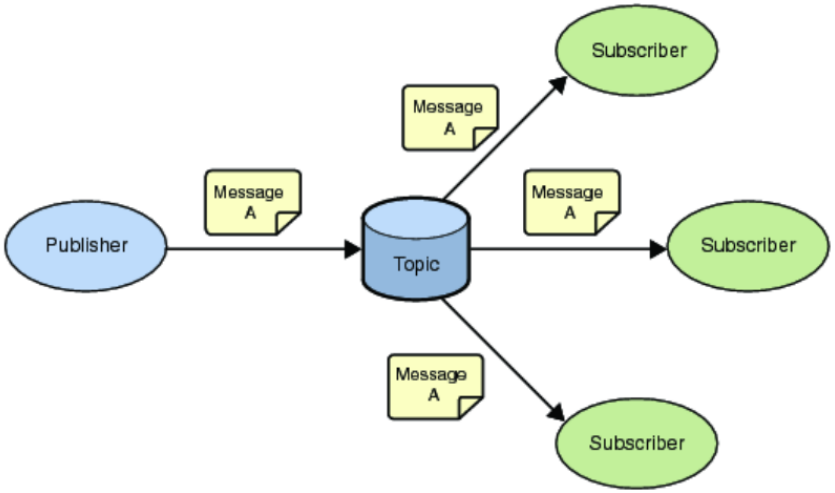
- event and message oriented
- asynchronous communication

Background / Original Design Goals

- "MQTT was originally invented and developed by IBM in the late 1990's.
- Its original application was to link sensors on oil pipelines with satellites. As its name suggests, it is a messaging protocol that supports asynchronous communication between parties.
- An asynchronous messaging protocol de-couples the message sender and receiver in both space and time, and hence is scalable in unreliable network environments.
- Despite its name, it has nothing to do with messaging queues, and uses a publish and subscribe model instead.
- In late 2014, it officially became an OASIS open standard, and it is supported in popular programming languages by using multiple open source implementations." (IBM)

MQTT - Publish Subscribe Messaging / One to Many Communication

- A Publish Subscribe messaging protocol allowing a message to be published once and multiple consumers
- (applications / devices) to receive the message providing decoupling between the producer and consumer(s)
- A producer sends (publishes) a message (publication) on a topic (subject)
- A consumer subscribes (makes a subscription) for messages on a topic (subject)
- A message server / broker matches publications to subscriptions
  - If no matches the message is discarded
  - If one or more matches the message is delivered to each matching subscriber/consumer



MQTT – Topic Structure

- topics form a namespace
- → hierarchical structure with each “sub topic” separated by a /

- example: a house publishes information about its rooms on:
- topic structure:
  <country>/<town>/<postcode>/<street>/<number>/<roomnr>/energyConsumption
  <country>/<town>/<postcode>/<street>/<number>/<roomnr>/temperature
  <country>/<town>/<postcode>/<street>/<number>/<roomnr>/humidity

- Example:
  russia/sankt-peterburg/191002/Lomonosova-street/9/1234/energyConsumption
  russia/sankt-peterburg/191002/Lomonosova-street/9/1234/temperature
  russia/sankt-peterburg/191002/Lomonosova-street/9/1234/humidity

MQTT – getting data

- a subscriber can subscribe to an absolute topic or can use wildcards:
- single-level wildcards “+” can appear anywhere in the topic string
  - example
    - russia/sankt-peterburg/191002/Lomonosova-street/+/+/energyConsumption
    - → energy consumption from every building in Lomonosova-street, any room

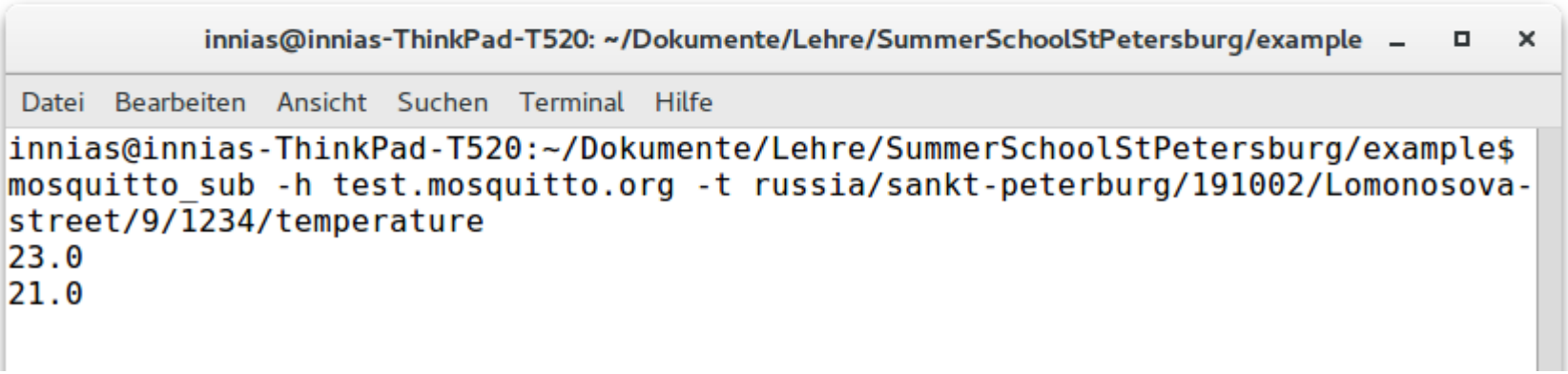
- multi-level wildcards “#” must appear at the end of the string
  - example
    - russia/sankt-peterburg/191002/Lomonosova-street/9/#
    - → any information from building number 9

- wildcards must be next to a separator
- cannot use wildcards when publishing

Example

- there are several test servers on the Internet
  - here used: test.mosquitto.org
  - and mosquitto_pub, mosquitto_sub command line tools

```
 mosquitto_pub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosov
a-street/9/1234/temperature -m 23.0
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
 mosquitto_pub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosov
a-street/9/1234/temperature -m 21.0
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
 mosquitto_pub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosov
a-street/9/1235/temperature -m 24.0
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
```



intro e.g.: https://developer.ibm.com/articles/iot-mqtt-why-good-for-iot/

example with wildcard "+"

```
 mosquitto_pub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosov
a-street/9/1235/temperature -m 24.0
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
 mosquitto_pub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosov
a-street/9/1234/temperature -m 21.0
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
 mosquitto_pub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosov
a-street/8/5678/temperature -m 21.0
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
```

innias@innias-ThinkPad-T520: ~/Dokumente/Lehre/SummerSchoolStPetersburg/example

Datei Bearbeiten Ansicht Suchen Terminal Hilfe

```
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
mosquitto_sub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosova-
street/9/1234/temperature
23.0
21.0
^C
innias@innias-ThinkPad-T520:~/Dokumente/Lehre/SummerSchoolStPetersburg/example$
mosquitto_sub -h test.mosquitto.org -t russia/sankt-peterburg/191002/Lomonosova-
street/+/+/temperature
24.0
21.0
21.0
```

example with "#" on root level on the test server

TECHNISCHE HOCHSCHULE MITTELHESSEN

innias@innias-ThinkPad-T520: ~/Dokumente/Lehre/SummerSchoolStPetersburg/example

Datei Bearbeiten Ansicht Suchen Terminal Hilfe


```
41-a337-342062d73ea9","FolderType":0,"Kind":"5135ba21-f1dc-4321-806a-6ce2017343c
0"}},"messageId":"0245bafd-91f3-4845-af29-8fc3461fa037"}}
{"eventHeader":{"id":"8e34398e-b543-46a0-82cc-d58b9915cb10","timestamp":"2019-10
-03T19:27:10.779Z","type":"System Event","version":"1.0","priority":1,"priorityN
ame":"High","name":"Motion Detected","message":"Motion Detected","source":{"name
":"StableFPS-3 - Camera 6","fqid":{"ServerId":{"Type":"XPCORS","Hostname":"pslab
-razgpu","Port":7563,"Id":"6328f4a5-a9d7-4088-bf50-b94fc4961e96","Scheme":"http"
},"ParentId":"6328f4a5-a9d7-4088-bf50-b94fc4961e96","ObjectId":"f674aa31-5220-44
56-806a-3675f90410fc","FolderType":0,"Kind":"5135ba21-f1dc-4321-806a-6ce2017343c
0"}},"messageId":"0245bafd-91f3-4845-af29-8fc3461fa037"}}
{"eventHeader":{"id":"fd459982-de63-4cdf-8096-695b9e31b074","timestamp":"2019-10
-03T19:27:10.837Z","type":"System Event","version":"1.0","priority":1,"priorityN
ame":"High","name":"Motion Detected","message":"Motion Detected","source":{"name
":"StableFPS-3 - Camera 12","fqid":{"ServerId":{"Type":"XPCORS","Hostname":"psla
b-razgpu","Port":7563,"Id":"6328f4a5-a9d7-4088-bf50-b94fc4961e96","Scheme":"http
"},"ParentId":"6328f4a5-a9d7-4088-bf50-b94fc4961e96","ObjectId":"c4e7af61-df1f-4
8d7-9b98-bf39863d671b","FolderType":0,"Kind":"5135ba21-f1dc-4321-806a-6ce2017343
c0"}},"messageId":"0245bafd-91f3-4845-af29-8fc3461fa037"}}
{"lux": 213.2}
{"ts":1570130830387,"objects":[]}
{"ts":1570130830387, "counts":{"person":0}}
{"ts":1570130830387, "counts":{"person":0}}
U2FsdGVkX1/ROoSXuo7MugpwW5zO5fXi2DfSCmWnEN2RcU6XMnYGhv/MAkJVv5orxAezfZWKtOMUOVqK
```

MQTT - Quality of Service

- QoS 0 – Only Once
  - fastest method, requires only 1 message
  - most unreliable transfer mode
  - the message is not stored on the sender, and is not acknowledged
  - the message will be delivered only once, or not at all
  - e.g. sensor data sent frequently, one is lost? no problem!
-
- QoS 2 - At Least Once
  - this level guarantees that the message will be delivered at least once, but may be delivered more than once
  - the sender sends a message and waits for an acknowledgement
- QOS 2 – Only Once
  - this level guarantees that the message will be delivered only once
  - this is the slowest method as it requires 4 messages

MQTT – further features

- a subscription can be **durable** or **non-durable**
  - durable:
    - once a subscription is in place a broker will forward matching messages to the subscriber:
      - immediately if the subscriber is connected
      - if the subscriber is not connected messages are stored on the server/broker until the next time the subscriber connects
  - non-durable:
    - the subscription lifetime is the same as the time the subscriber is connected to the server / broker
- a publication may be **retained** ("retained message")
  - a publisher can mark a publication as "retained"
  - the broker / server remembers the last known good message of a retained topic
  - the broker / server gives the last known good message to new subscribers i.e. the new subscriber does not have to wait for a publisher to publish a message in order to receive its first message

Benefits of MQTT versus HTTP

- push delivery of messages / data / events
  - MQTT – low latency push delivery of messages from client to server and server to client
  - → event oriented architecture to the web
  - HTTP – push from client to server but poll from server to client
- decoupling and publish subscribe – one to many delivery
- efficient use of constrained networks

MQTT Security Mechanisms … its up to the broker

- Client Authentication
  - Client ids
  - usernames and passwords
  - client certificates
- x509 Client Certificates
  - most secure method of client authentication but also the most difficult to implement because you will need to deploy and manage certificates on many clients
  - suited for a small number of clients that need a high level of security
- Restricting Access to topics
  - based on user name or client Id
- Securing Data
  - TLS / SSL Security: TCP/IP protocol and not MQTT
  - Payload encryption

http://www.steves-internet-guide.com/mqtt-security-mechanisms/

Vulnerability

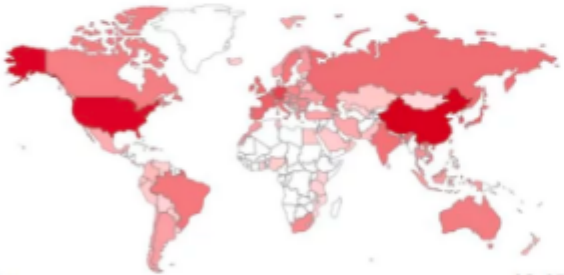
- Example: https://blog.avast.com/mqtt-vulnerabilities-hacking-smart-homes

So, what is the scale of this problem? According to the **Shodan IoT search engine**, there are almost 49,000 MQTT servers exposed on the internet. Of these, there are approximately 32,000 servers with no password protection.

TOTAL RESULTS

49,197

TOP COUNTRIES

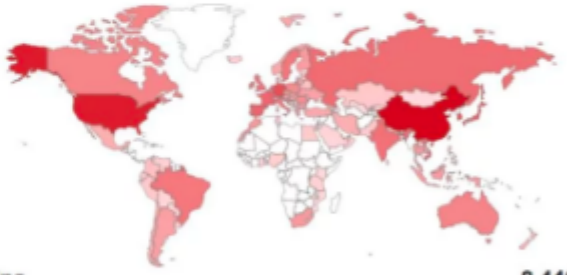


China	12,151
United States	8,257
Germany	3,092
Korea, Republic of	2,003
Hong Kong	2,002

TOTAL RESULTS

32,888

TOP COUNTRIES

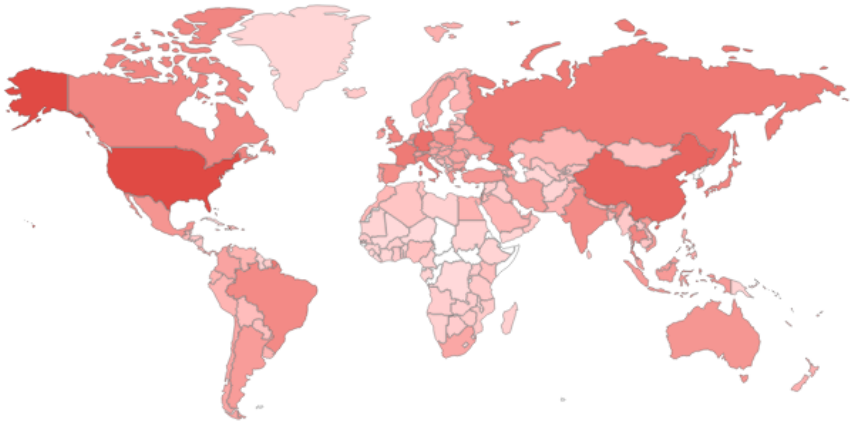


China	8,446
United States	4,733
Germany	1,719
Hong Kong	1,614
Taiwan	1,565

Heartbleed Report

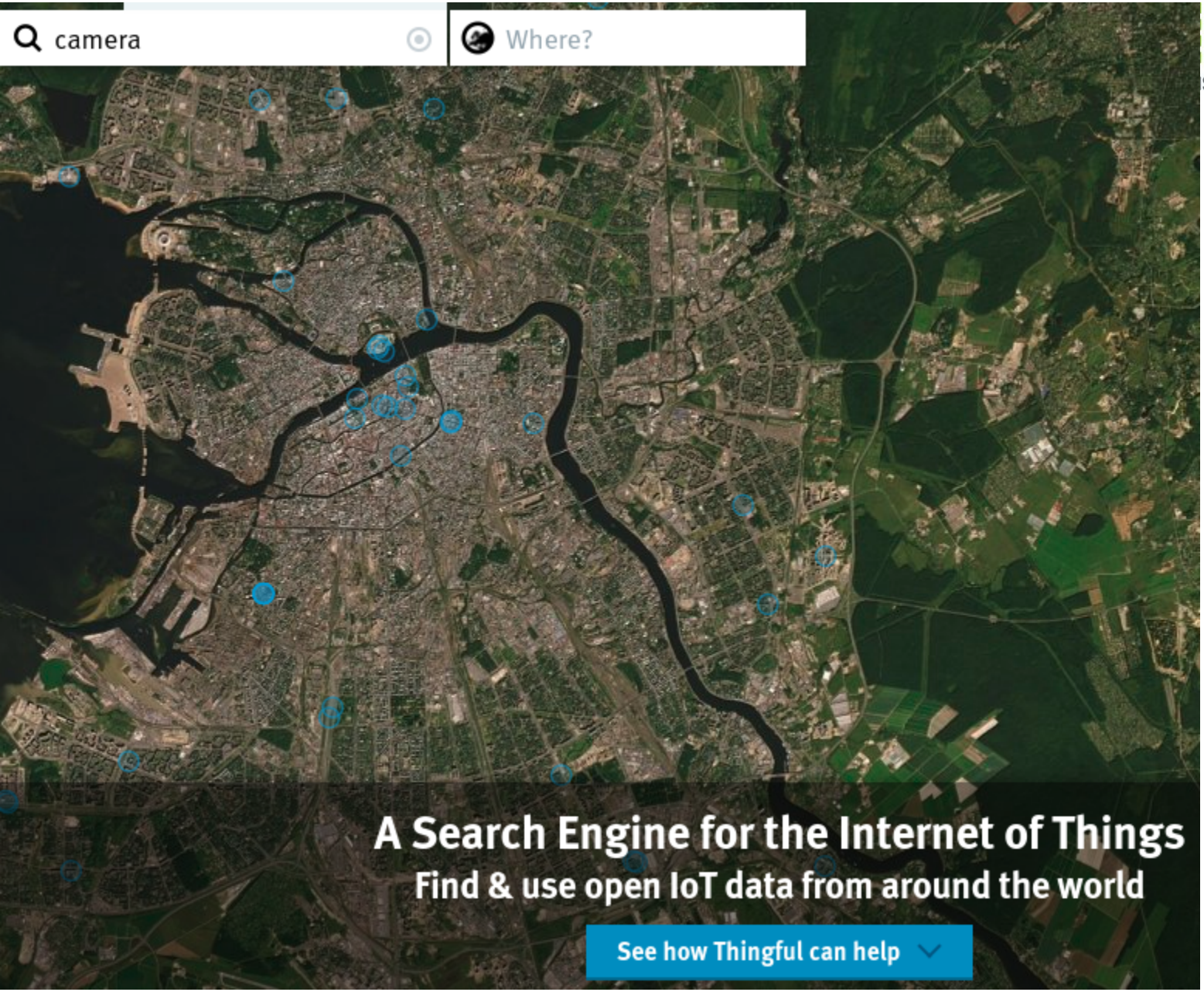
- https://www.shodan.io/report/0Wew7Zq7

Search for vuln:cve-2014-0160 returned 91,063 results on 11-07-2019



Top Countries

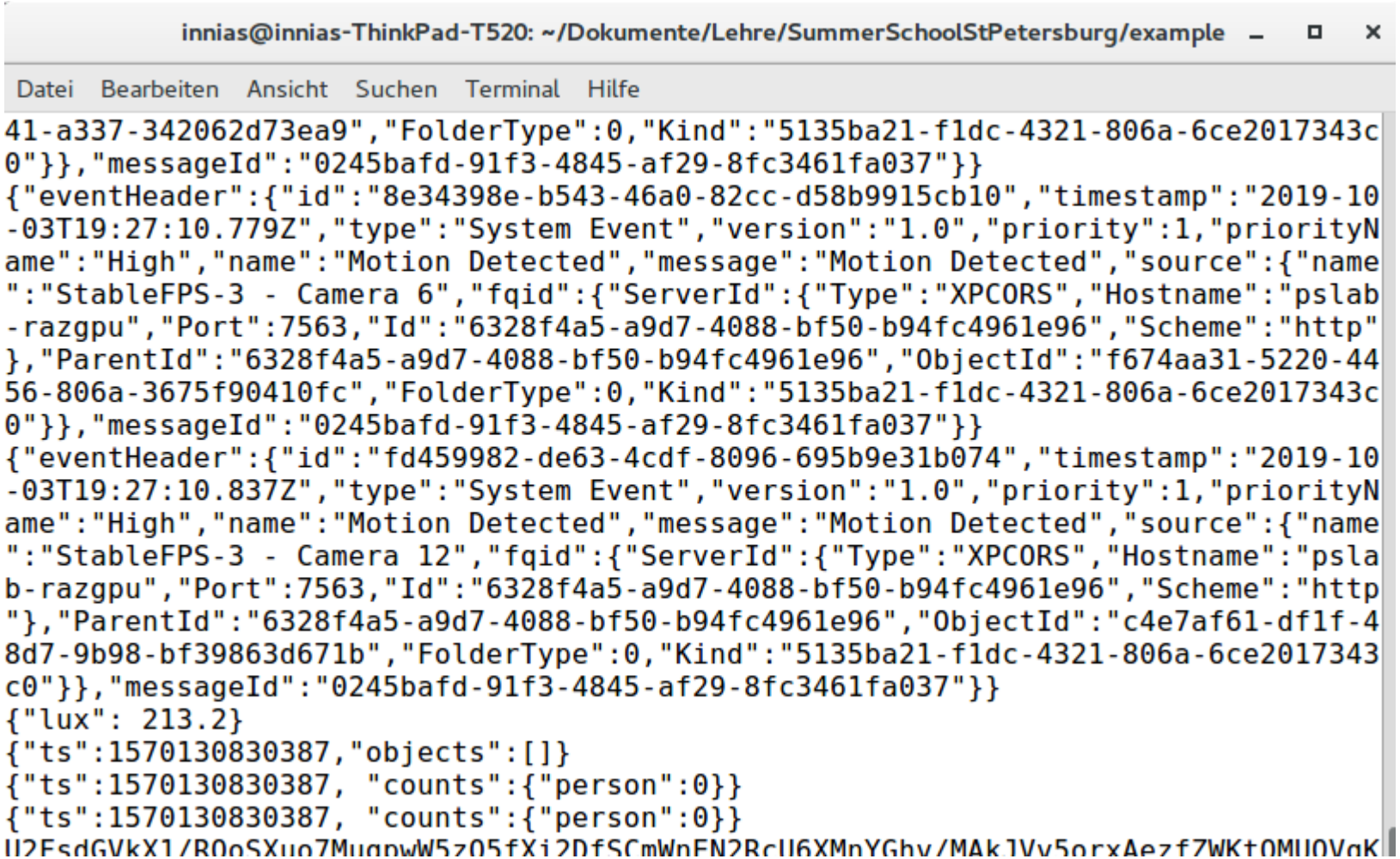
1.	United States	21,258
2.	China	8,655
3.	Germany	5,647
4.	Russian Federation	3,869
5.	France	3,660
6.	Korea, Republic of	3,407
7.	Italy	2,858
8.	Taiwan	2,639
9.	Japan	2,368
10.	United Kingdom	2,176


camera
Where?
A Search Engine for the Internet of Things
Find & use open IoT data from around the world
See how Thingful can help

Why is MQTT not enough?

- possible variants
  - <country>/<town>/<postcode>/<street>/<number>/<roomnr>/humidity
  - <country>/<postcode>/<town>/<street>/<number>/<roomnr>/humidity
  - <continent>/<country>/<postcode>/<town>/<street>/<number>/humidity
  - …
- which is the one and only correct version?
  - → all versions are possible
  - → subscriber has to know the structure
- what about the payload?

from scanning the root of the test server with wildcard



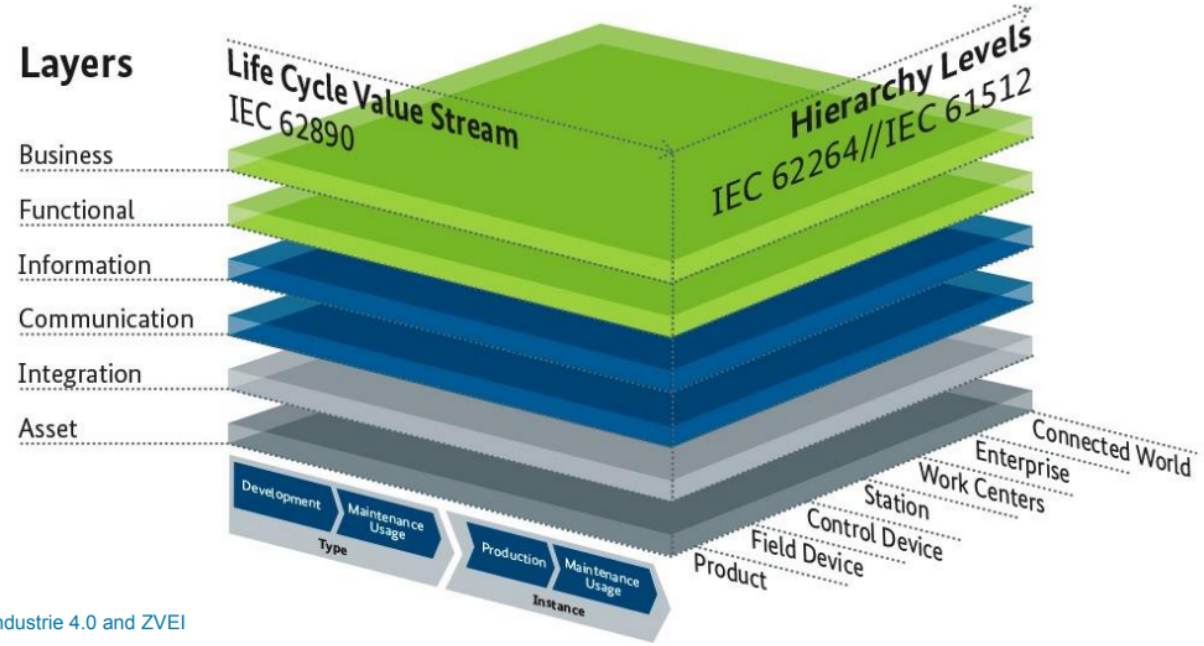
payload is often JSON – but you can use plain, XML, ...

OPC UA

- OPC stands for Open Platform Communications…
  - … and is one of the most important communication protocols for Industry 4.0 and the IoT.
  - → https://opcfoundation.org/wp-content/uploads/2017/11/OPC-UA-Interoperability-For-Industrie4-and-IoT-EN.pdf
  - OPC UA serves as the common data connectivity and collaboration standard for local and remote device access in IoT, M2M, and Industrie4.0 settings.
  - OPC-UA – also published as IEC 62541 – enables exchange of information models of any complexity – both instances and types (metadata). It thus complements the standards referred to above and enables interoperability at the semantic level.

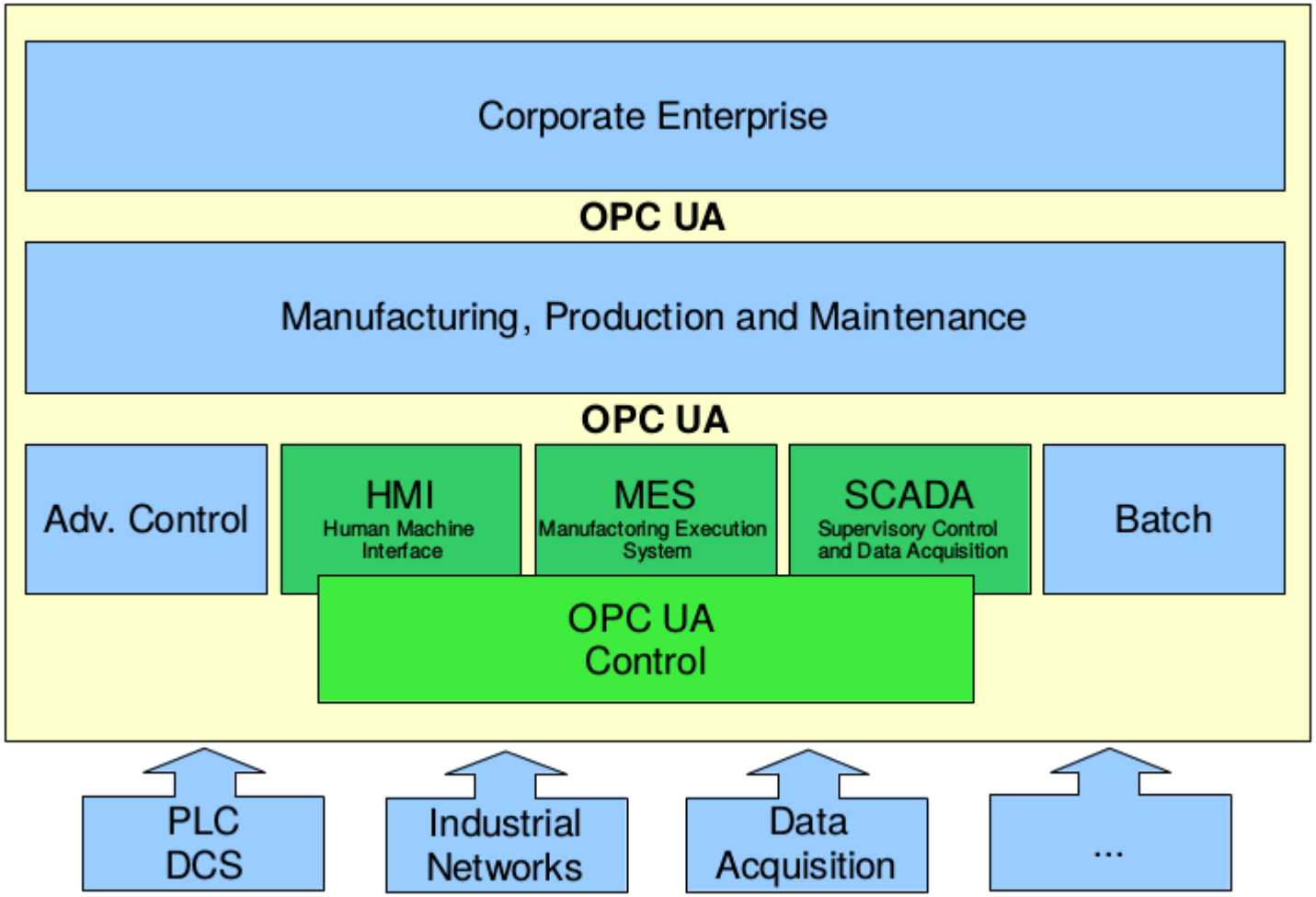
OPC UA as the Standard for both: Communication and Information Layer in RAMI 4.0

- RAMI 4.0: Reference Architectural Model for Industrie 4.0
  - "RAMI 4.0 is a three-dimensional map showing how to approach the issue of Industrie 4.0 in a structured manner.
  - RAMI 4.0 ensures that all participants involved in Industrie 4.0 discussions understand each other."
  - (https://ec.europa.eu/futurium/en/system/files/ged/a2-schweichhart-reference_architectural_model_industrie_4.0_rami_4.0.pdf)



Graphics © Plattform Industrie 4.0 and ZVEI

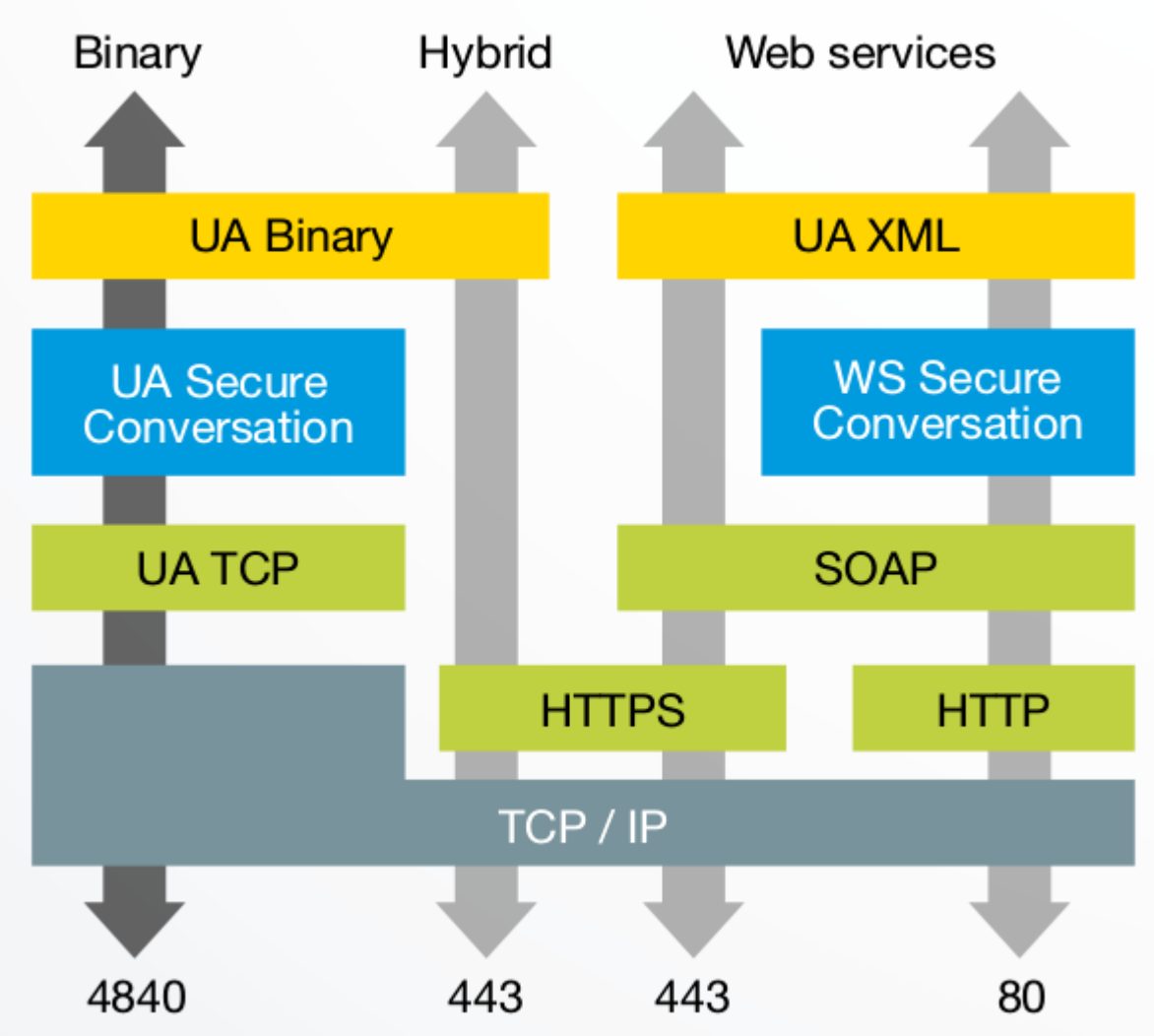
OPC UA Application Domain



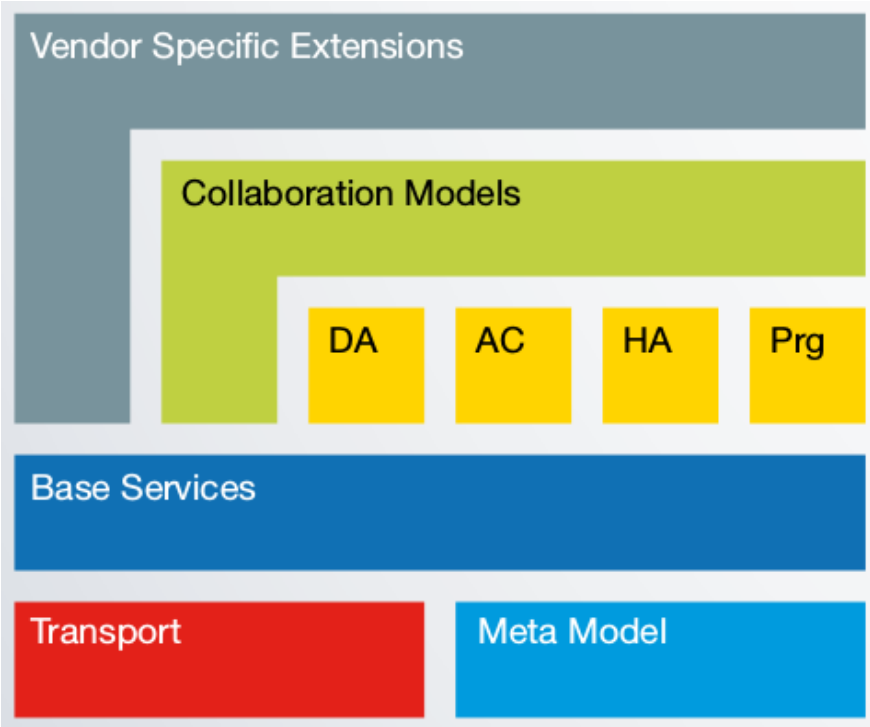
OPC-UA Design Objectives

- support of a wide range of systems:
  - ranging from PLC's in production to enterprise servers
  - diversity in terms of size, performance, platforms and functional capabilities
- → specification of common functionalities
  - **Transport** – for the data exchange mechanisms between OPC-UA applications. Different transport protocols exist for different requirements (optimized for speed and throughput = UA TCP with UA Binary; firewall-friendly = HTTP + Soap).
  - **Meta model** – specifies the rules and basic components for publishing an information model via OPC-UA. It also includes various basic nodes and basic types.
  - **Services** – they constitute the interface between a server as information provider and clients as users of this information.
  - Further reading:
    https://opcfoundation.org/wp-content/uploads/2016/05/OPC-UA-Interoperability-For-Industrie4-and-IoT-EN-v5.pdf

OPC-UA transport profiles



OPC-UA layer model



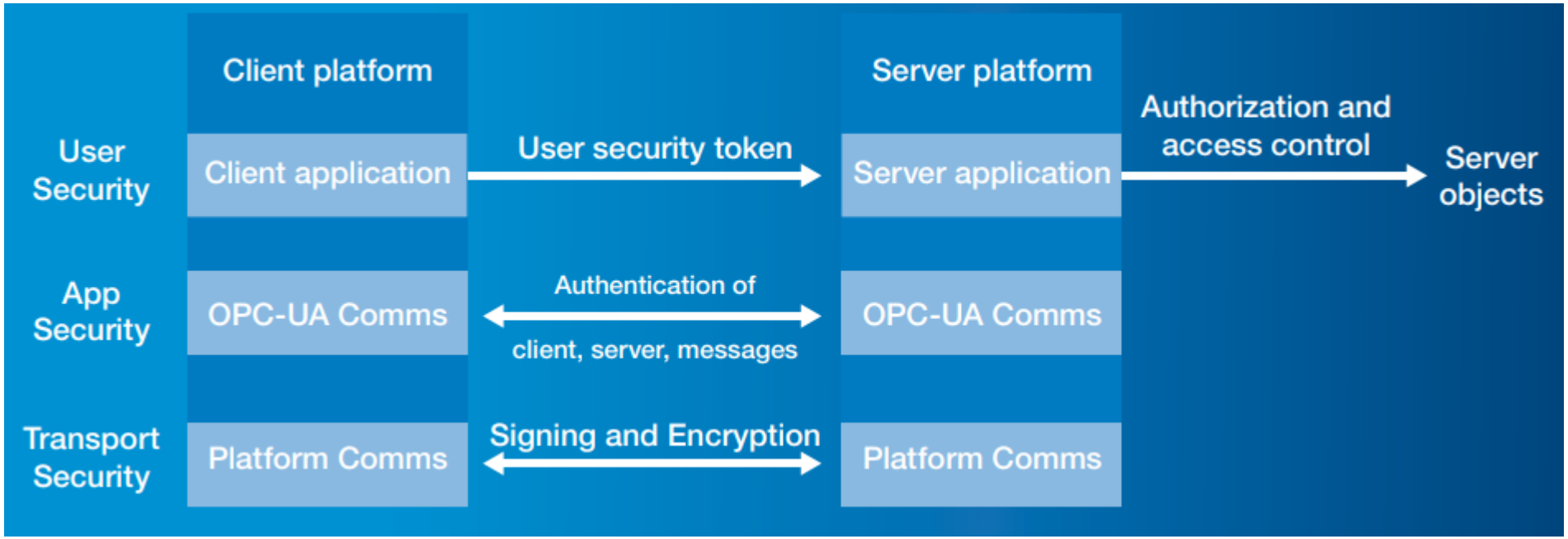
- Generic information models
  - Models for generally valid information e.g. alarms or automation data
- example: Data access (DA)
  - "modelling of real-time data, i.e. data that represent current state and behaviour of the underlying industrial or business process data.
  - It includes the definition of analog
  - and discrete variables, engineering units and quality codes. Data sources are sensors, controllers, position encoders etc."

OPC-UA services overview

Service Set	Description
SecureChannel Service Set	retrieve endpoint and security configuration to establish a secure connection
Session Service Set	create and administrate user-specific connection between application
NodeManagement Service Set	modify the server's address space (if permitted)
View Service Set	navigate and follow (hierarchical) references in the server's address space, search for and filter information
Attribute Service Set	read and write attributes of (an) node(s), especially the value attribute, but historical data or events as well
Method Service Set	invoke methods which a server provides at the nodes in its address space
MonitoredItem Service Set	create a set of attributes of nodes to be monitored by the server and for which changes should be reported
Subscription Service Set	create, modify, or delete monitored items
Query Service Set	perform a filtered search for information in the server's address space

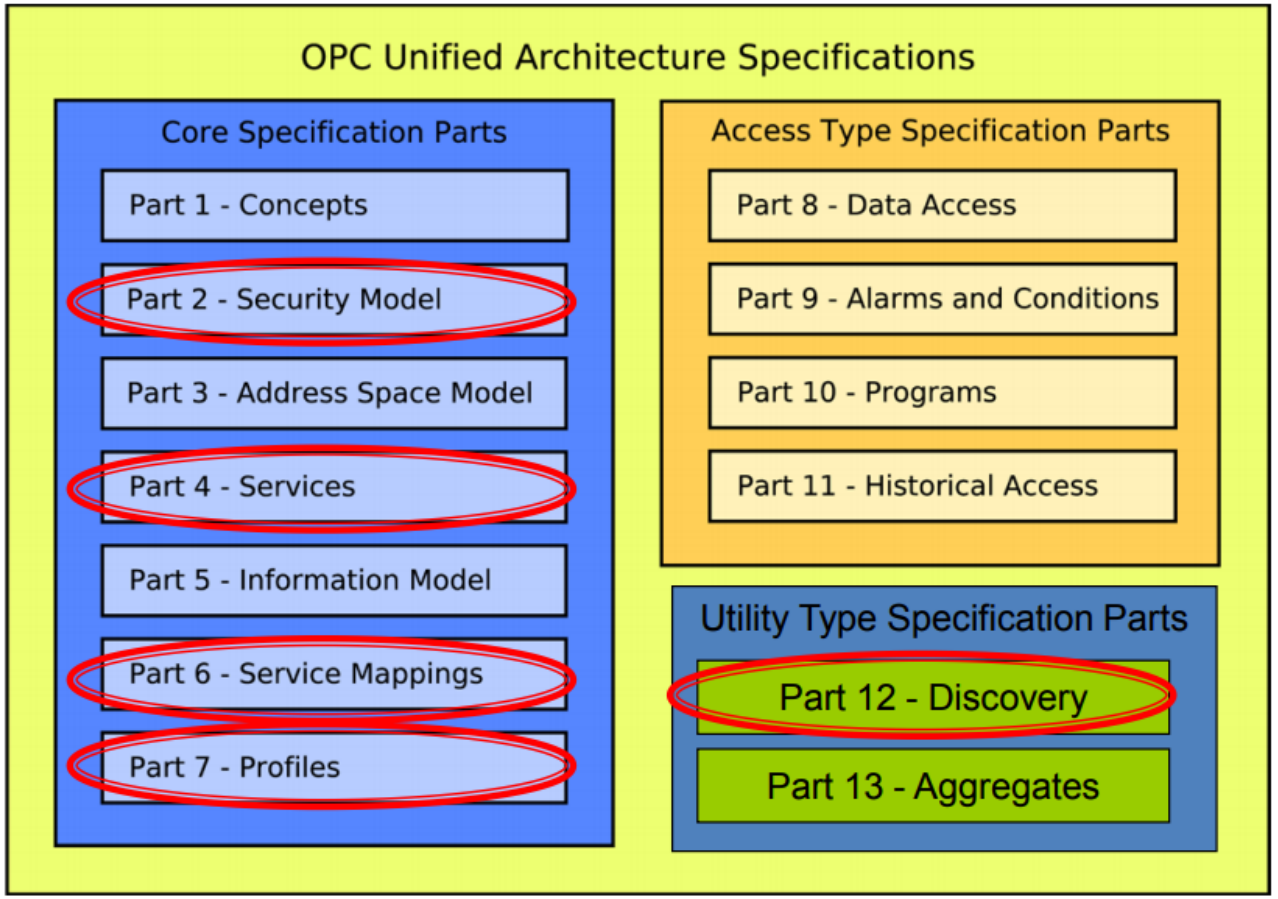
Example: SecureChannel service set

- includes services to determine the security configuration of a server and establish a communication channel in which the confidentiality and completeness (integrity) of the exchanged messages is guaranteed
- are not implemented directly in the OPC-UA application but are provided by the communication stack used
- Scalable security concept:



OPC UA: Security by Design

- Security built into specification from ground up:



- Further reading:
  https://opcfoundation.org/wp-content/uploads/2014/08/11_OPC_UA_Security_How_It_Works.pdf

OPC UA Security Analysis

- OPC UA communication stack was subjected to the following security tests in the second part of the project:
  - Certificate tests
  - Static code analysis
  - Fuzzing
  - Dynamic code analysis

- Main results
  - The specification analysis performed has shown that OPC UA, in contrast to many other industrial protocols, provides a high level of security.

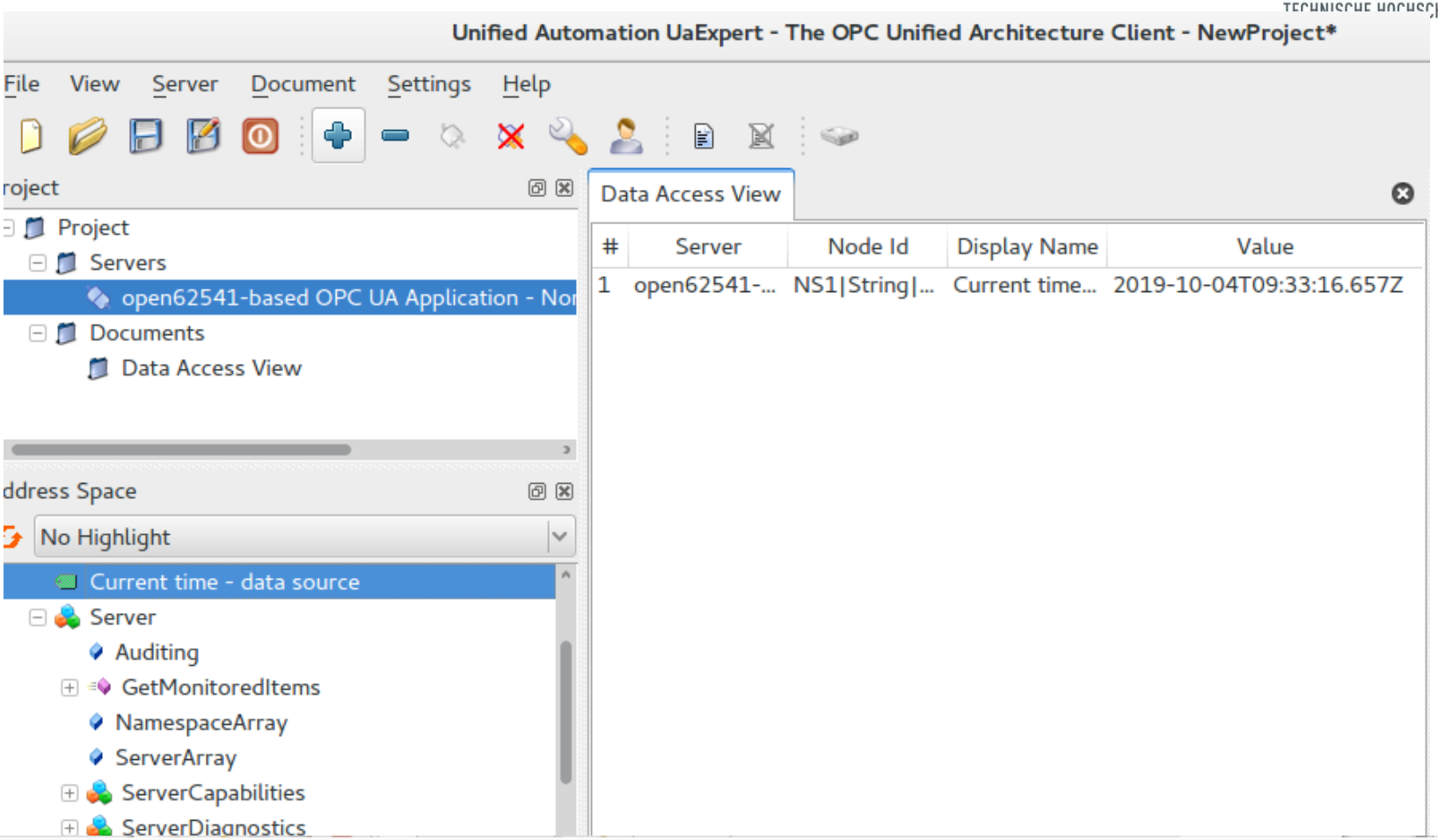
  - **No systematic errors could be detected.**

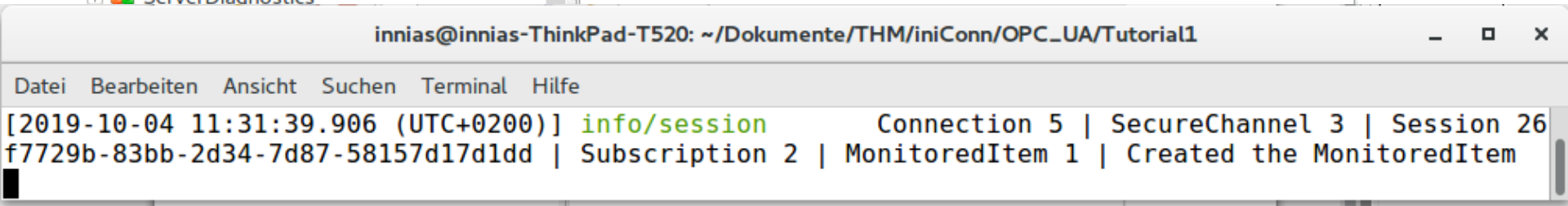
Getting started

- Several implementations of OPC UA stacks: commercial and open source
- Open source: open62541
  - https://open62541.org/doc/current/toc.html
  - good introduction
  - several tutorials
  - maintained by community under leadership of Fraunhofer IOSB
  - acceptance in industry

- Free (not open source) client
  - UA Expert (https://www.unified-automation.com/products/development-tools/uaexpert.html)

simple time server ...

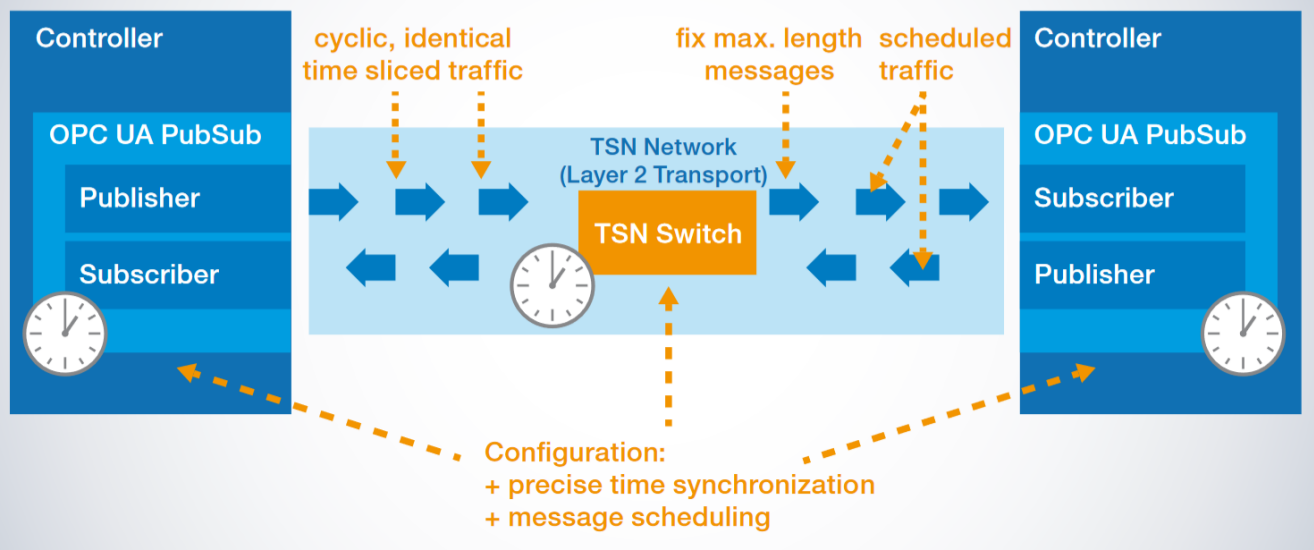




Our Research and Development

- integration of hardware-based trust anchors into OPC UA
- multi-protocol interoperability via model transformation

- further steps in OPC UA open62541
  - publisher / subscriber and time sensitive networks



- https://opcfoundation.org/wp-content/uploads/2017/11/OPC-UA-Interoperability-For-Industrie4-and-IoT-EN.pdf

Conclusion

- IoT / CPS need for suitable communication techniques
  - events, constrained applications, unreliable network infrastructure

- IIoT / CPPS need for
  - standardisation
  - performance
  - security
  - shared information models

- here introduced
  - MQTT and OPC UA

- "hot topics"
  - OPC UA pub/sub via MQTT
  - OPC UA time sensitive networks