MAI4CAREU | Master programmes in Artificial Intelligence 4 Careers in Europe

University of Cyprus

MAI650 Internet of Things

Vasos Vassiliou
September - December 2023

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

CS6xx Internet of Things (8 ECTS)

Course purpose and objectives: The purpose of the course is to provide an overview on IoT tools and applications and to introduce to students hands-on IoT communication concepts through lab exercises.

Learning outcomes: Upon completion of this course, students will be able to explain the definition and usage of the term “Internet of Things” in different contexts. More specifically, the students will know how to apply the knowledge and skills acquired during the course to build and test a complete, working IoT system involving prototyping, programming and data analysis

Teaching methodology: interactive face-to-face lectures, group activities and discussions, in class/lab activities, student presentations and guest lectures or significant recorded public lectures

Assessment: Final exam (50%), midterm exam (20%) and assignments/project (30%).

Main text:

Rajkumar Buyya, Amir Vahid Dastjerdi, Internet of Things Principles and Paradigms, Morgan Kaufmann; 1st edition, 2016

J. Biron and J. Follett, "Foundational Elements of an IoT Solution", O'Reilly Media, 2016.

Other reading:

Jamil Y. Khan and Mehmet R. Yuce, Internet of Things (IoT) Systems and Applications, 2019, ISBN 9789814800297

David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, and Jerome Henry, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, 2016, Cisco Press.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

INTRODUCTION

Architectural Design and Applications in IoT- Core

CONTENTS

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

INTENDED LEARNING OUTCOMES

Upon completion of this introductory unit, students will be:

1. familiar with the design requirements for IoT Applications
2. familiar with the different software architectural styles in IoT
3. familiar with the major IoT applications Domains
4. familiar with the selected IoT applications analysis
5. familiar with the quality attributes analysis in IoT

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

MAI4CAREU

Master programmes in Artificial Intelligence 4 Careers in Europe

Design Requirements for IoT Applications

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

IoT System Design Requirements

- When designing a system, a number of functional and non-functional requirements need to be fulfilled.
- Functional requirements relate to the actual functionality that the system needs to have.
  - Provided by system implementation
- Non-functional requirements (quality attributes) relate to the quality of the system.
  - Introduced mainly by the software architecture

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Software quality attributes by ISO 25010 standard

- Functional Suitability
- Performance Efficiency
- Compatibility
- Usability
- Reliability
- Security
- Maintainability
- Portability

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

- Interoperability
- Evolvability
- Performance
- Scalability
- Availability
- Resiliency
- Security
- Privacy

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Interoperability

- IoT contains different heterogeneous components, networks and systems that need to work together
- Interoperability can be reached by:
  - integrating one IoT solution as a subsystem in another solution by communicating in the same format,
  - building a bridge (mediator) through which the key functionalities of each solution can be used.
- The type of architecture cannot guarantee interoperability, however design choices can have an impact on how easily a solution can interoperate with another solution.
- Interoperability can be defined in 4 levels:
  - technical
  - syntactic
  - semantic
  - organizational

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Interoperability

- Technical interoperability is associated with hardware/software components.
  - The discussion is usually centered on communication protocols: TCP/IP provides seamless connectivity
- Syntactic interoperability is associated with data formats.
  - The data being transferred must have some well-defined syntax and encoding. Examples of high-level transfer syntaxes include JSON, XML, HTML or ASN.
- Semantic interoperability concerns the understanding of meaning of the exchanged information.
  - Enables systems to combine received information with other information resources and to process it in a meaningful manner
  - Allows providers and requesters of information to communicate meaningfully despite the heterogeneity

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Interoperability

- Organizational interoperability concerns cross-domain service integration through common semantics and programming interfaces.
  - common interpretation of semantic information in a globally shared ontology is not always the case
  - common/generic API established by the EU-funded project BIG IoT between different middleware platforms in co-operation with the Web of Things Interest Group at the W3C

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Evolvability

- IoT system designs need to withstand and adapt to new requirements and changes.

- Evolvable architectures need to **allow changes without damaging system integrity** and evolve in a controllable way.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Performance

- Performance measures the responsiveness and stability of a system
- Measured in:
  - response time,
  - round-trip time (RTT),
  - throughput,
  - ability to gracefully recover from stress testing
- Influenced by:
  - System architecture: edge/fog/cloud computing
  - Software architecture and implementation

- Response Time: amount of time system (edge gateway or server-side infrastructure) takes to process a request after it has received one
- Round Trip Time: amount of time it takes for a request sent from an edge device to a destination (gateway or server-side infrastructure), and for the response to get back to the edge device
- Throughput: transactions / second system can handle
  - Can be measured using load testing
    - Pick up a mix of transactions (frequent, critical, and intensive) and measure how many pass successfully in an acceptable time frame

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Performance

- Stress testing recovery: is the measure of how system recover from over loaded (stressed) conditions.
  - Does it limp back to normalcy or gives up completely?
    - Example 1: if system does not release memory after working at peak loads => run out of memory
    - Example 2: if system fails due to constant heavy I/O load => loss of data
- System designers have to decide how much of the system logic and data storage is pushed to the edge and how much is done on centralized servers in order to achieve acceptable performance
- Edge/Fog computing: improve system performance with respect to response time
- Cloud computing: simplifies edge devices but can be problematic for delay-sensitive applications

Co-financed by the European Union
Connecting Europe Facility

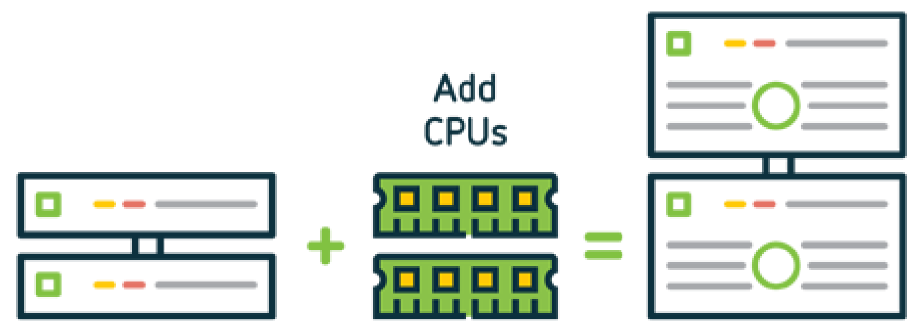
This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Scalability

- Ability of a system to handle increased workload
- In IoT, server-side infrastructure needs to have a scaling strategy
- Cloud computing has made it possible to pay for scalability on demand, meaning that extra resources can be added (or removed) at any time when there is more (or less) work to be done by the system running in the cloud.

- Option 1: Scale system vertically/up means increasing the resources (e.g. CPU, RAM) of a particular node:
- Costly solution
- No-fault tolerance achieved: in a single-node system if node goes down, system goes down
- Requires node restart => system has to go down

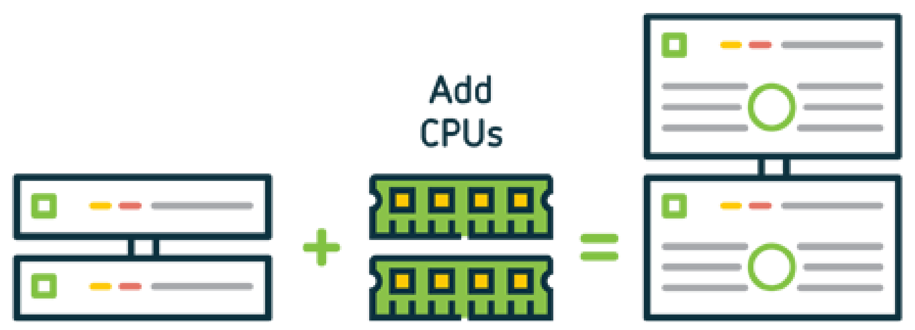


Co-financed by the European Union
Connecting Europe Facility

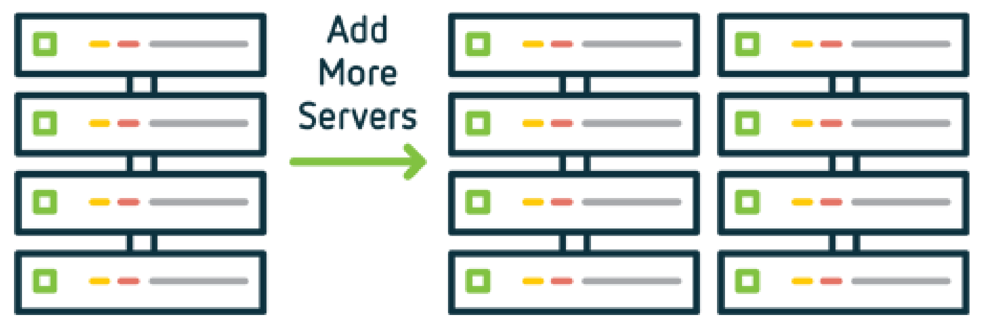
This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Scalability – Scaling strategies

- Option 1: Scale system vertically/up means increasing the resources (e.g. CPU, RAM) of a particular node:
  - Costly solution
  - No-fault tolerance achieved: in a single-node system if node goes down, system goes down
  - Requires node restart => system has to go down



- Option 2: Scale system horizontally/out means getting more nodes to share the workload.
  - decreases requests/sec rate in each node => lower response time, higher throughput
  - achieves fault-tolerance
  - easy setup (just add new node, system not going down)
  - Load balancer needed to balance load across nodes



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Availability

- Ability of a system to be fully or partly operational as and when required
- Eliminate single points of failure by achieving redundancy on all levels both at the edges and at the server-side infrastructure.
- High availability is closely related with disaster recovery or automatic failover which is a mechanism designed to minimize or eliminate downtime.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Resiliency

- Ability to recover from temporary failures or through some explicit error handling and error correction
- Ability to recover from an attack
- Complexity of systems can lead to less resiliency
- Disaster recover mechanisms play important role in resiliency

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Important quality attributes in IoT by Internet of Things Architecture (IoT-A)

Security

- Edge devices may store sensitive information and might be physically reachable => attacker can easily extract information and compromise confidentiality
- More communication channels between edge devices, middleware and servers => attacker can compromise integrity of messages
- Limited computation power and battery disallows running complicated cryptography algorithms => attacker can authenticate himself to an edge device, cause DoS attacks to device and the server-side

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Privacy

- Data gathered by edge devices may contain personal information
- People not aware what personal data is captured
- Data could be gathered without consent
- Privacy is diminished in the IoT is because of the pervasiveness and the scale
- Many approaches to providing privacy in IoT have been proposed, see the specific educational module in “IoT Security and Privacy”

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

MAI4CAREU Master programmes in Artificial Intelligence 4 Careers in Europe

Software architectural styles in IoT

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Software Architectural Styles in IoT

- A general, reusable solution to a commonly occurring problem in software architecture

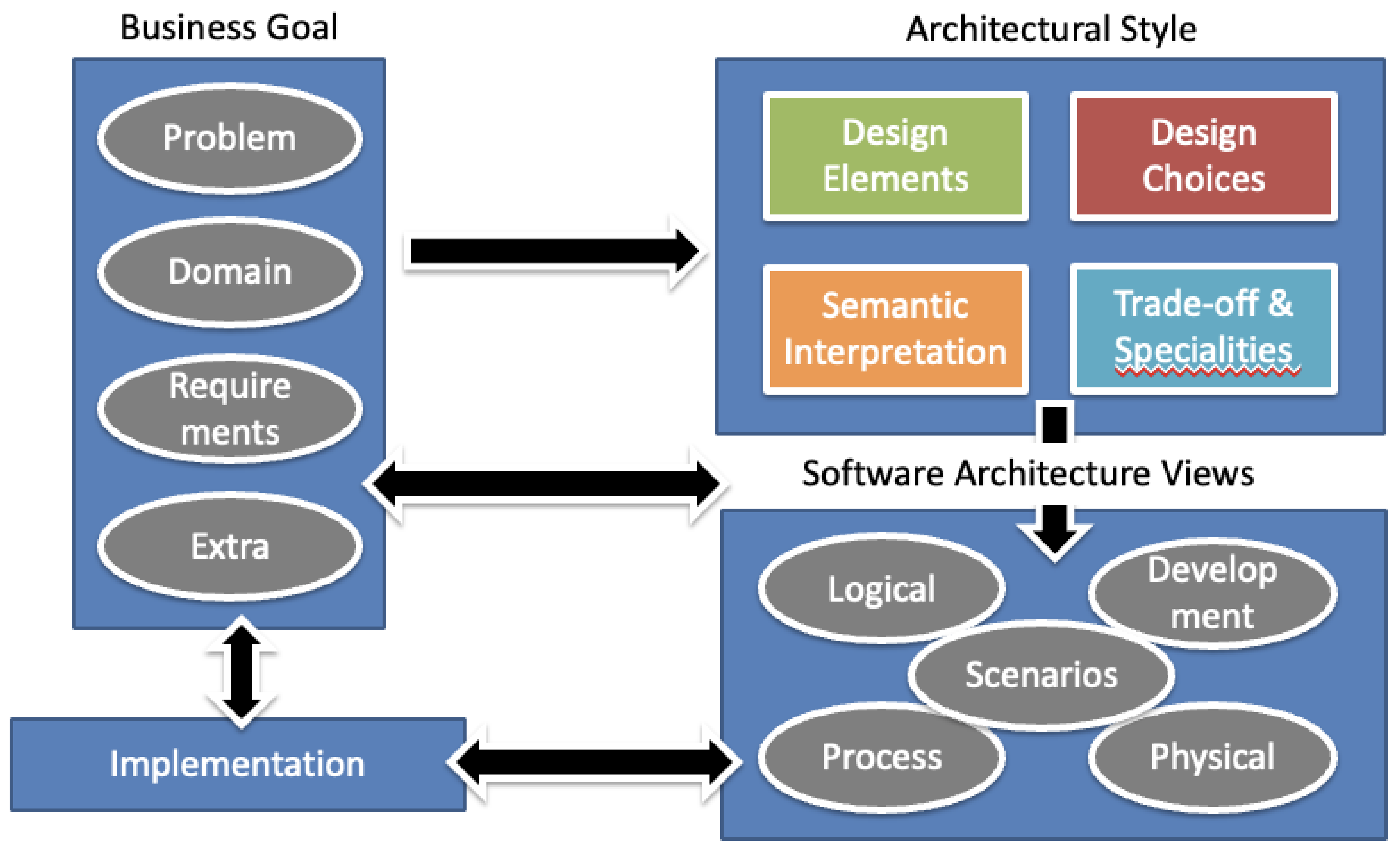
- First step in software design is the selection of the architectural style to be used, based on a given set of quality requirements

- A software architectural style is often described by:
  - a labelled set of components and connectors
  - a set of constraints on how components interact

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Software Architectural Styles in IoT



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Software architectural styles in IoT

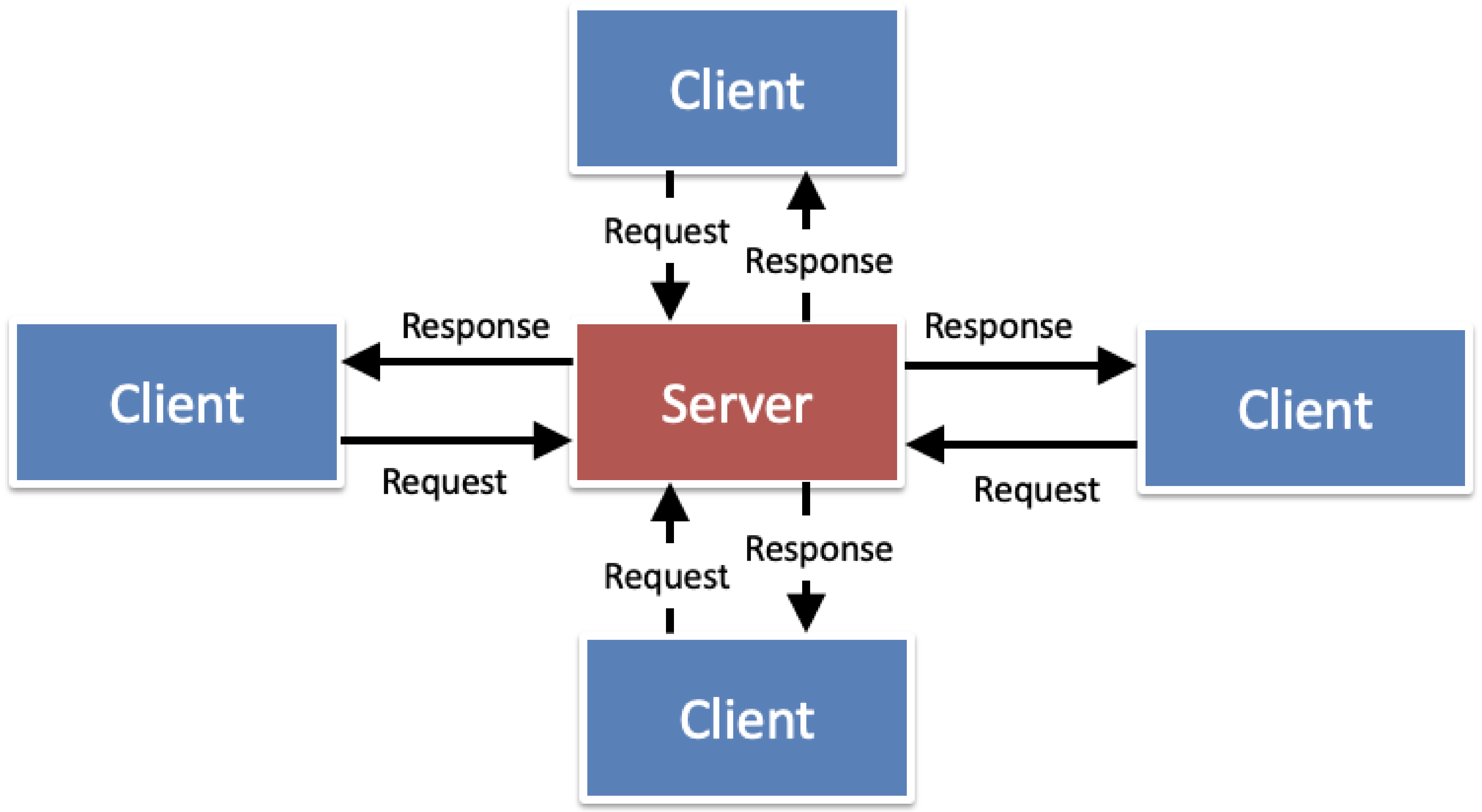
- Client-Server
- Peer-to-Peer
- Pipes and Filters
- Event-Based
- Publish-Subscribe
- Service-Oriented
- REST
- Layered
- Microkernel

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Client – Server Architecture

- Constraints
  - Only clients can initiate communication, servers only respond to requests.
  - Clients cannot communicate to each other.
- Advantages
  - Evolvability: Common services only need to be modified in a single location.
  - Availability/Scalability: Centralizing control of resources allows for distribution among physical servers.
- Disadvantages
  - Performance: The server can be a bottleneck, since all communications go through it.
  - Availability: The server is a single point of failure.
  - Complexity: For some applications it can be hard to decide where to put functionality.

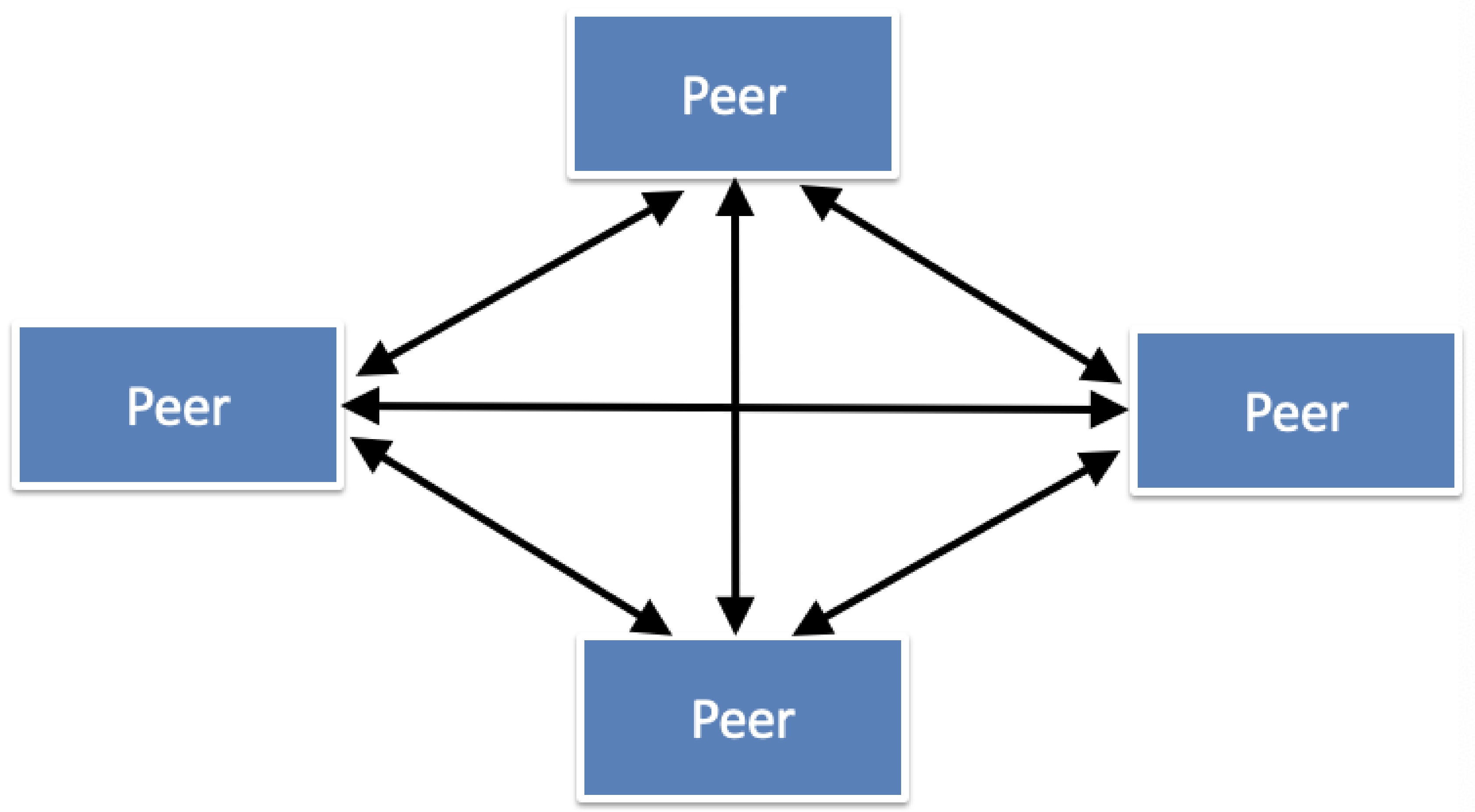


Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Peer to Peer (P2P) Architecture

- Constraints:
  - Peers connect to each other directly.
  - Primary content provided by peers, no central components providing content.
  - Peers are autonomous and control their own activities.
  - Peers can be added and removed from the system at any time.
- Advantages:
  - Performance: Response-time only depends on connection to peer
  - Availability: If a peer should fall, the system would still continue to function
  - Availability: If a peer is not available, another peer can pick up the workload
  - Scalability: The system can be scaled simply by adding more peers
- Disadvantage:
  - The performance and availability gains depend if the network is large enough
  - Administration and security are much more complex
  - Data recovery or backup is difficult

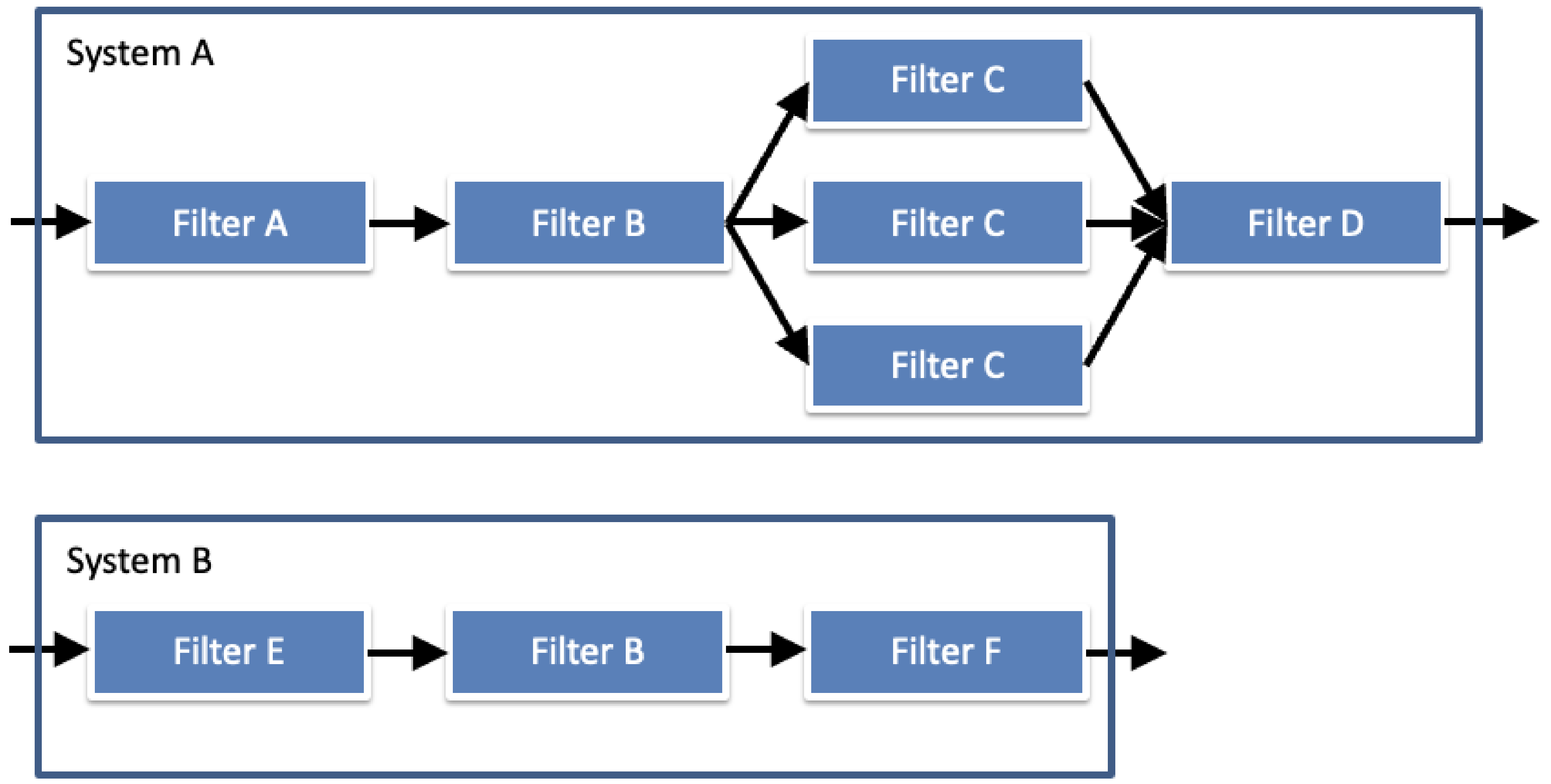


Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Pipes and Filters Architecture

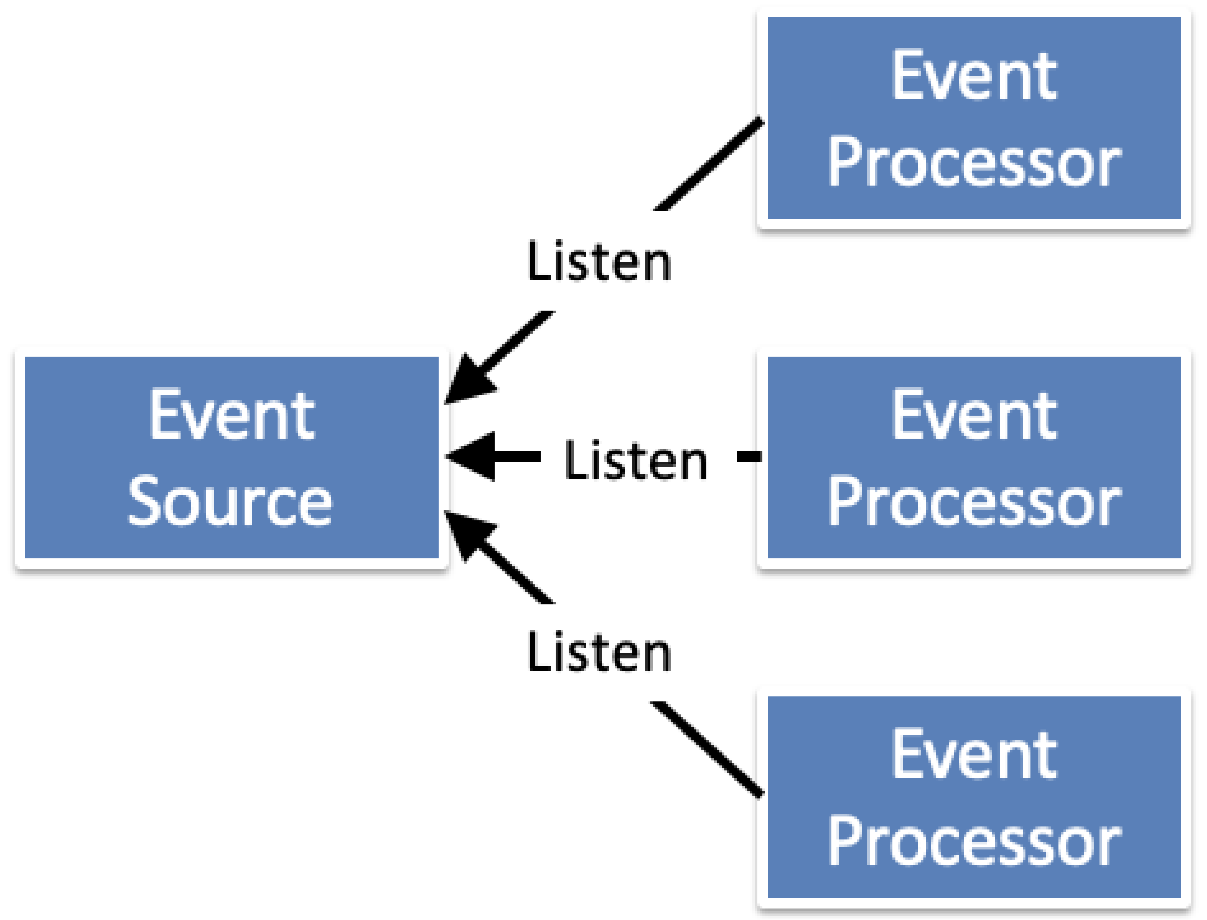
- Constraints:
  - Filters connected by a pipe must agree on the type of data being passed along that pipe.
  - Filters are independent of each other and do not share a state.
  - Filters can execute in parallel.
  - Filters start to producing output as soon as they start processing input.
- Advantages:
  - Understandability: Entire system can be considered a composition of behaviors of individual filters
  - Reusability: A filter can be reused in another system completely.
  - Evolvability: New filters can be added and old filters can be replaced since they are all independent.
  - Scalability: It is possible to individually scale certain filters by running more instances in parallel.
- Disadvantages:
  - Interactivity: Due to the batch organization of processing, interactivity cannot be handled well.
  - Availability: If a filter in the chain of filters fail, it usually leads to failure of the system (unless there are multiple instances of the filter running).
  - Performance: System may have to wait for parallel instances of a filter to finish processing the data.
  - Performance: Certain implementations may force a lowest common denominator on the transmission of the data, which may result in more work for some filters.

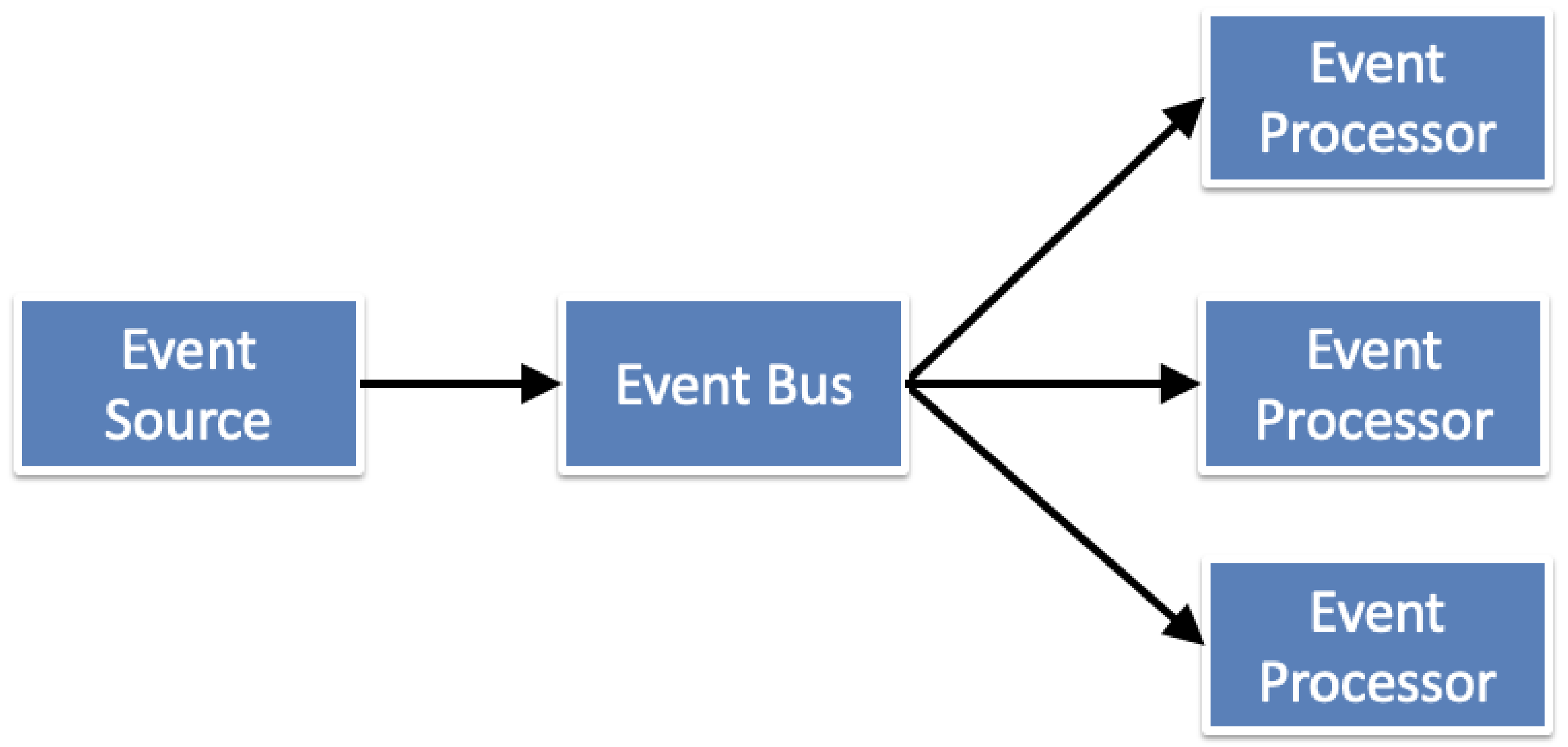


Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Event-Based, Event-Driven, Event-Oriented Architecture





Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Event-Based, Event-Driven, Event-Oriented Architecture

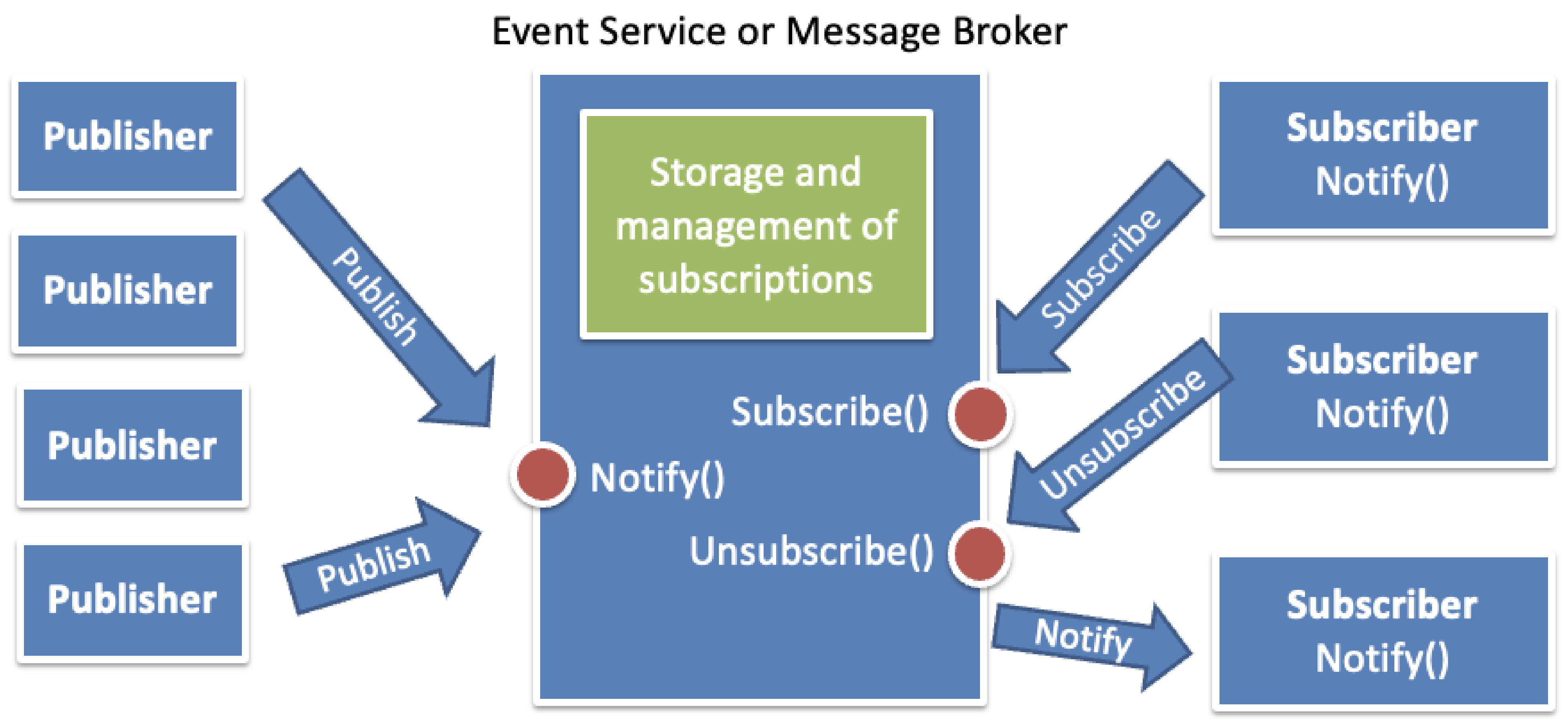
- Constraints:
  - The event source does not know the identity of the event processors.
- Advantages:
  - Reusability: A processor can be used in any other system by registering to its event source.
  - Evolvability: Processors can evolve or be replaced without affecting the event-source or other event-processors.
  - Performance: Ability to perform tasks in parallel.
  - Scalability: Event processors can be dynamically added by simply registering to the Event Bus. However, the max number of connections depends on the Event-Bus capabilities.
  - Efficiency: For data monitoring applications, efficiency can be increased (no need for polling)
- Disadvantages:
  - Testability: It is generally harder to recreate some events and the asynchronous characteristics also make testing harder.
  - Correctness: Due to the asynchronous nature, it is hard to reason about termination and correctness of the end state.
  - Performance: The Event Bus is an intermediary and adds to overall latency.
  - Availability: The Event Bus is a single point of failure.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Publish-Subscribe Architecture

- Topic-based:
  - messages grouped into topics that are identified by keywords
- Content-based:
  - messages classified based on their properties and not by some predefined grouping
  - subscription language is provided in order to allow subscribers to express in detail which messages they would like to subscribe to
- Type-based:
- Requires messages published to have a well-defined type



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Publish-Subscribe vs Event-based

- Publish-Subscribe:
  - subscribers are all interested in a type of message happening without knowing the publisher of the message
- Event-based:
  - subscribers are interested in a particular source of a message
- Publish-subscribe decouples message sources and message consumers in systems that are large in scale and heterogeneous

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Publish-Subscribe Architecture

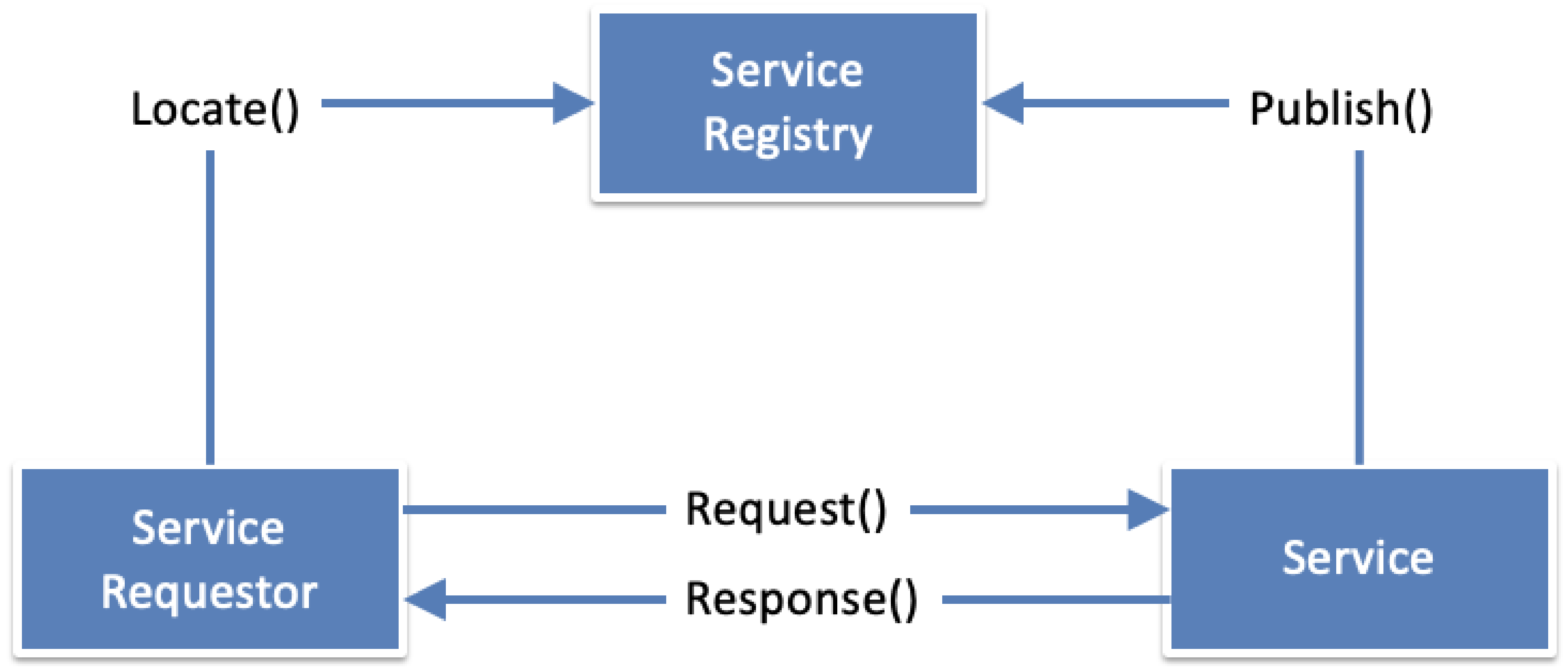
- Constraints:
  - All publishers and subscribers are connected to the event service.
- Advantages:
  - Evolvability: Publishers and Subscribers are loosely coupled and thus allow for independent evolution.
- Disadvantages:
  - Performance: The message broker is an extra layer and adds to overall latency. It increases latency more than the Event Bus for the Event-Based style since it also has to be able to classify data and understand complex queries.
  - Availability: The message broker is a single point of failure
  - Scalability: as the number of publishers, subscribers and messages grows, the likelihood of instabilities increases, limiting the maximum scalability of a pub/sub network

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Service-oriented Architecture (SOA)

- Constraints:
  - All services are independent, meaning they do not share a state and do not have to know the identity of each other.
  - Service providers need to register their services to the service registry.
- Advantages:
  - Interoperability: The registry component allows for lookup of services and also knowledge about their interfaces. Orchestration services can be used to script and translate interactions between multiple services.
  - Evolvability: Services are completely decoupled.
- Disadvantages:
  - Complexity: Service-Oriented Architectures are typically complex to build.
  - Loss of control: The evolution of independent services cannot be controlled.

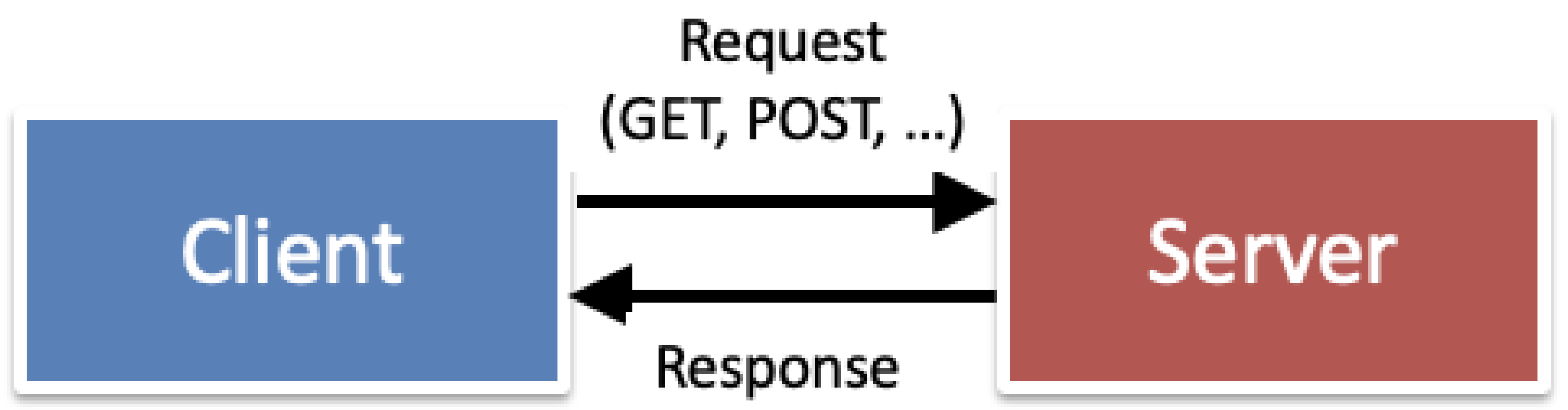


Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Representational State Transfer (REST)

- Defines a set of constraints to be used for creating Web services
- Follows the Client-Server architectural style, but with a uniform and predefined set of stateless operations
- HTTP protocol is commonly used to send/receive messages



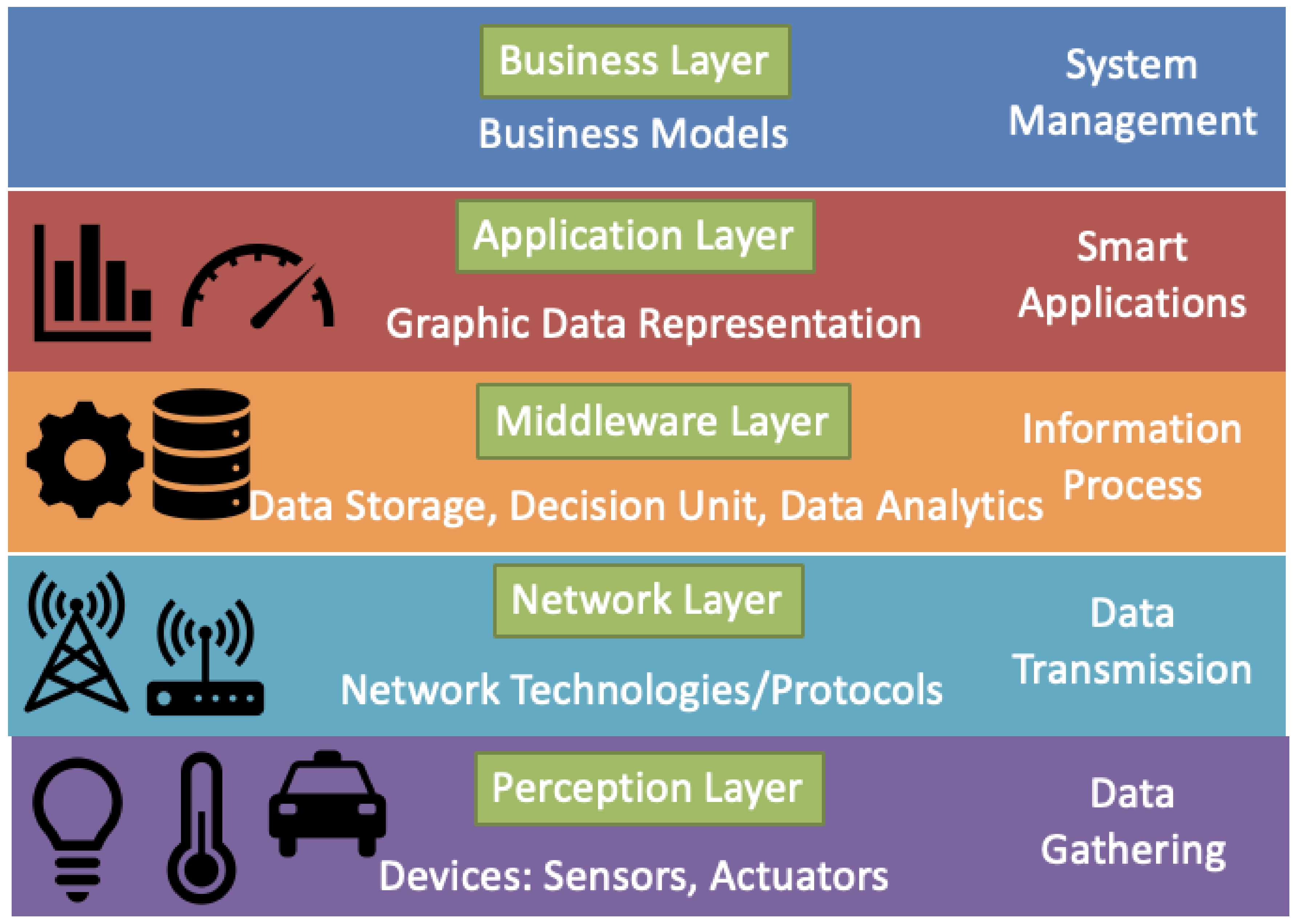
- Constraints:
  - Caching
  - Stateless communication
- Advantages:
  - Evolvability: decouples client and server operation
  - Interoperability: uniform and predefined set of stateless operations known throughout the system
  - Scalability: REST system can grow by reusing components that can be managed and updated without affecting the system as a whole, even while it is running
- Disadvantages:
  - Standardization: Data to be sent must be changed into a standardized form
  - Performance: amount of data transferred in request and response messages is significant due to the presence of HTTP headers

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Layered Architecture

- Constraints:
  - Every piece of software is part of exactly one layer.
  - There are at least two layers.
  - A lower layer cannot "use" an upper layer.
- Advantages:
  - Evolvability: Layers can change as long as their interfaces remain the same.
  - Reusability: Layers can be reused in other systems.
  - Security: Layers can be put in place to authenticate connections before allowing anything to change in sensitive areas of the system.
  - Testability: Other layers can be mocked in order to test specific layers.
- Disadvantages:
  - Performance: Multiple layers need to be traversed to fulfill a request
  - Complexity: Added overhead for simple applications
  - Resiliency: If a single layer fails, the entire system becomes inoperable

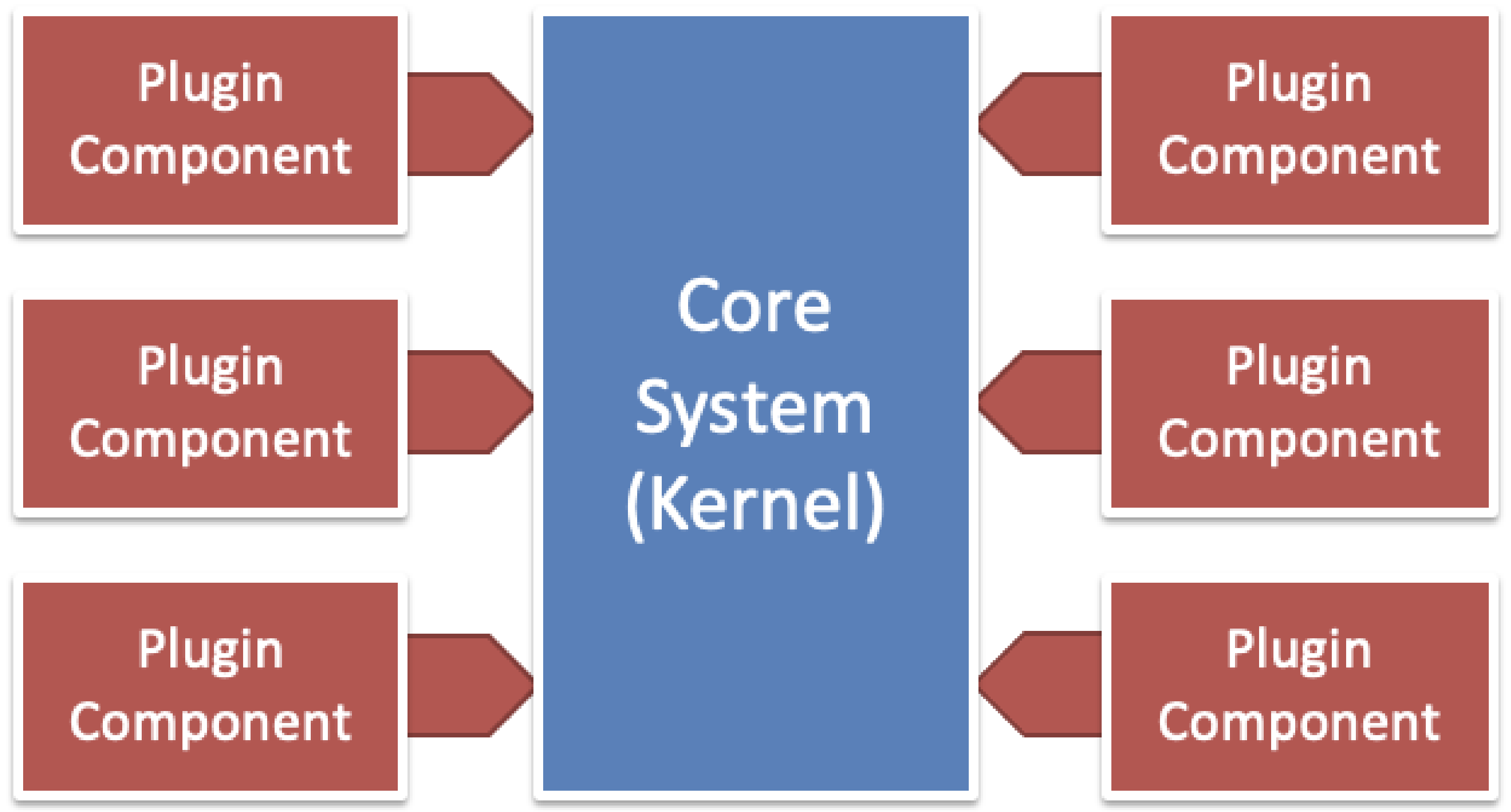


Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Microkernel Architecture

- Constraints:
  - Plugins are independent and cannot communicate to each other.
- Advantages:
  - Evolvability: Plugins can evolve independently without affecting each other.
  - Performance: With respect to memory used, since the user can choose to only use plugins that they need.
- Disadvantages:
  - Complexity: hard in some cases to decide what the bare minimum functionality is and to predict which types of plugins will come in the future.
  - Performance: Depending on the context, plugins might run on other processors or even other components in the network, increasing latency in the communication between plugins & the kernel



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Software Architectural Styles in IoT

- A general, reusable solution to a commonly occurring problem in software architecture
- First step in software design is the selection of the architectural style to be used, based on a given set of quality requirements
- A software architectural style is often described by:
  - a labelled set of components and connectors
  - a set of constraints on how components interact

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

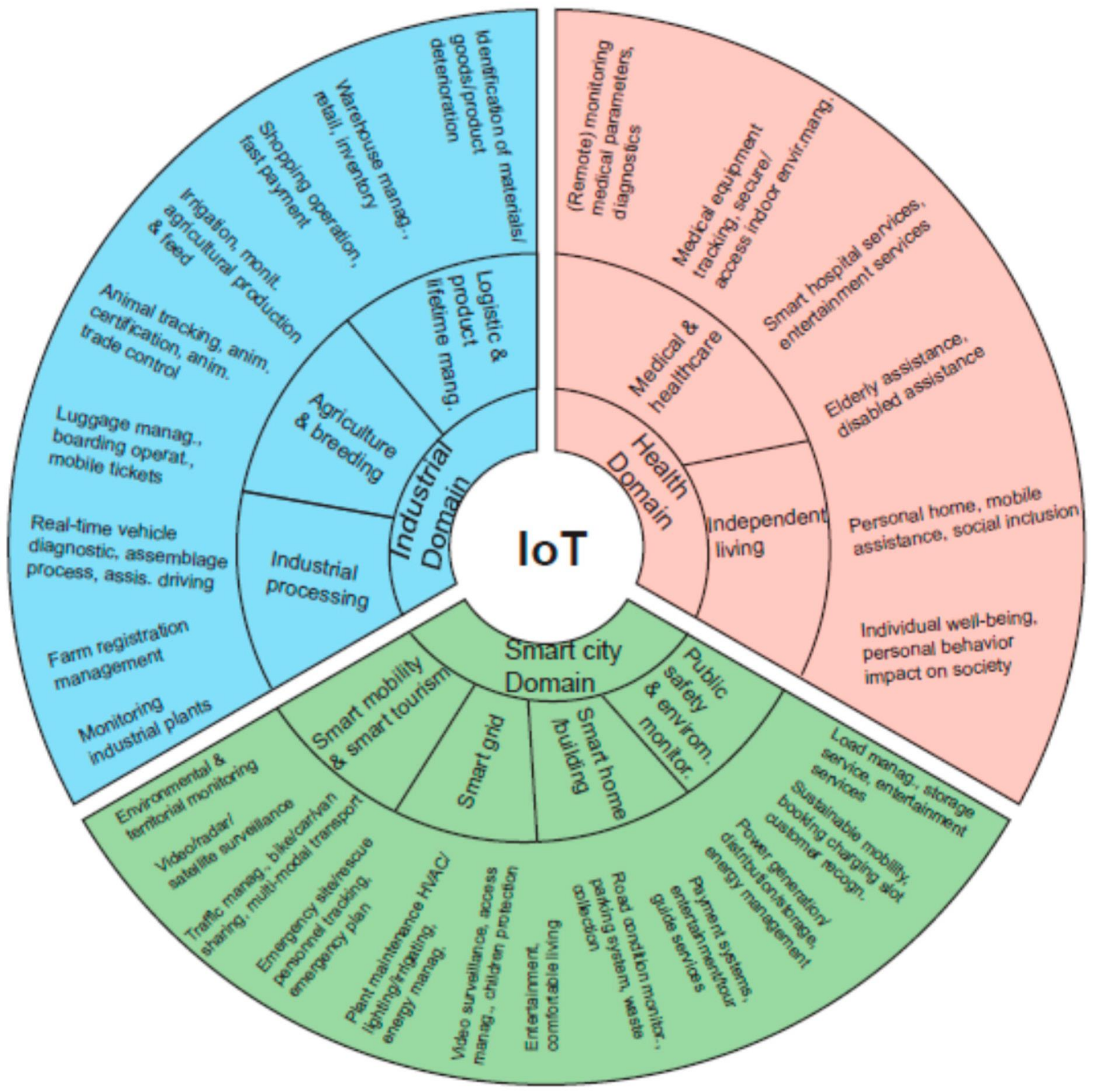
MAI4CAREU Master programmes in Artificial Intelligence 4 Careers in Europe

Major IoT Applications Domains

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Major IoT Application Domains


IoT
Health Domain
Medical & healthcare
Independent living
(Remote) monitoring medical parameters, diagnostics
Medical equipment tracking, secure/ access indoor envir.manag.
Smart hospital services, entertainment services
Elderly assistance, disabled assistance
Personal home, mobile assistance, social inclusion
Individual well-being, personal behavior impact on society
Smart city Domain
Smart mobility & smart tourism
Smart grid
Smart home /building
Public safety & envirom. monitor.
Load manag., storage service, entertainment services
Sustainable mobility, booking charging slot customer recogn.
Power generation/ distribution/storage, energy management
Payment systems, entertainment/tour guide services
Road condition monitor., parking system, waste collection
Entertainment, comfortable living
Video surveillance, access manag., children protection
Plant maintenance HVAC/ lighting/irrigating, energy manag.
Emergency site/rescue personnel tracking, emergency plan
Traffic manag., bike/car/van sharing, multi-modal transport
Video/radar/ satellite surveillance
Environmental & territorial monitoring
Industrial Domain
Logistic & product lifetime mang.
Agriculture & breeding
Industrial processing
Identification of materials/ goods/product deterioration
Warehouse manag., retail, inventory
Shopping operation, fast payment
Irrigation, monit. agricultural production & feed
Animal tracking, anim. certification, trade control
Luggage manag., boarding operat., mobile tickets
Real-time vehicle diagnostic, assemblage process, assis. driving
Farm registration management
Monitoring industrial plants

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Selected IoT Applications Analysis

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Google Nest Learning Thermostat

- Controls home heating system and hot water tank
- Learns the user daily routine from temperature changes in user's house
- Connected to Wi-Fi, remotely controlled using Google Apps (Home, Assistant) or Google Devices (Nest Hub Max, Home Hub)
- Changes heating system and water tank automatically
  - Aware of user presence and location

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Google Nest Learning Thermostat - Characteristics

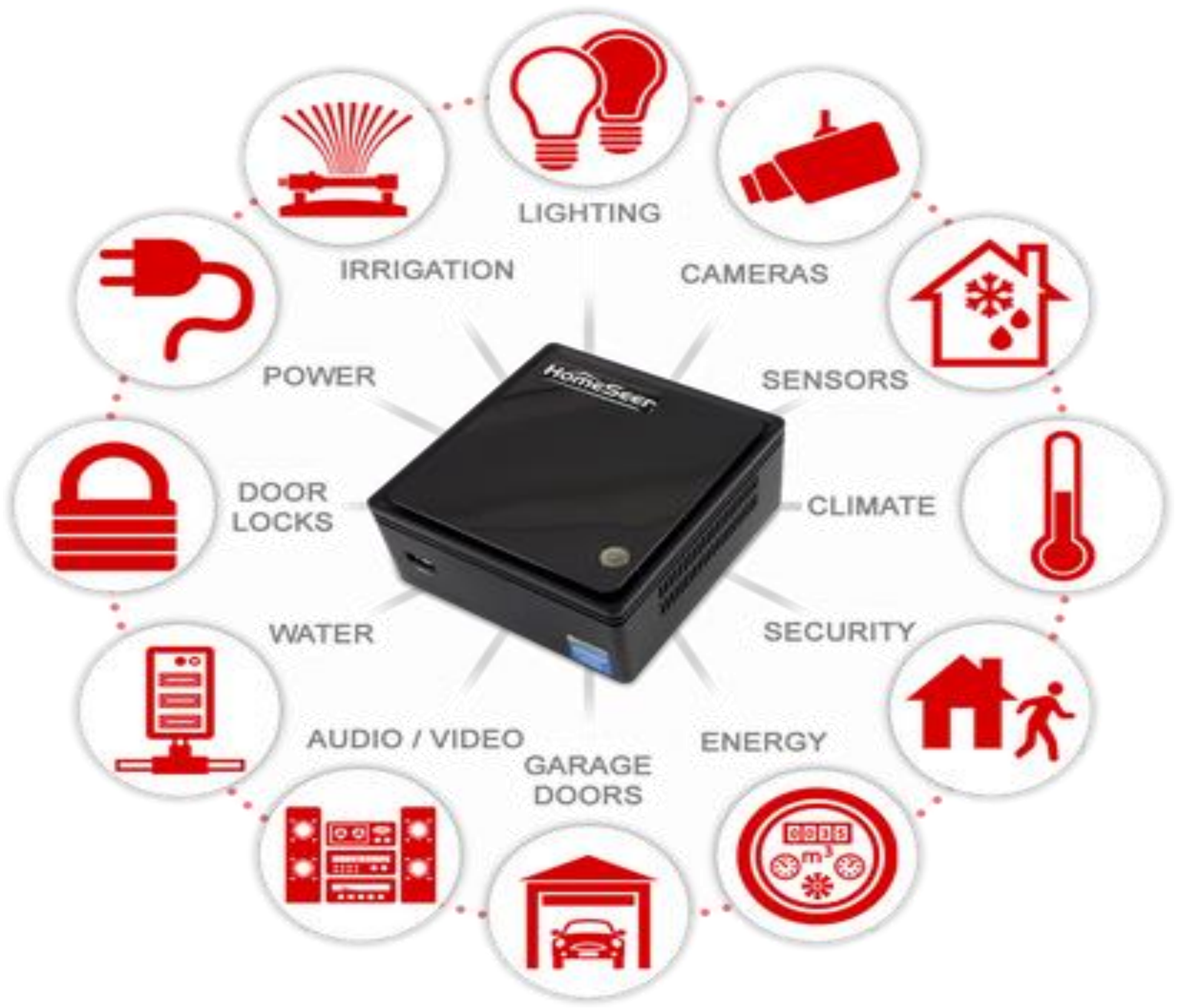
- Physical entity: home (environment), hot water tank
- Attributes: Temperature, Humidity, Proximity, Far-Field Activity, Near-Field Activity, Ambient light
- Edge IoT Device Components: Sensor and Actuator
- Application logic: Thermostat, apps, devices, cloud
- Data storage: Google Cloud
- User interaction: Google mobile apps, Google devices & thermostat LCD display
- Internet dependency: internet-dependent
- Type of invocation: Explicit and Implicit
- Interoperability: Create smart home actions
- Autonomous behavior: Cloud service learns, predicts, and acts autonomously

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Homeseer

- Complete package for home automation
- Automation for lights, temperature, locks, security, water, energy, audio and video
- Does not store data & personal information in the cloud
- HomeTroller controller with HS3 software at the center of the system
- HS3 software can be installed on personal computers
- Mobile app and web service to monitor and control devices
- Integrates a number of technologies



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Homeseer - Characteristics

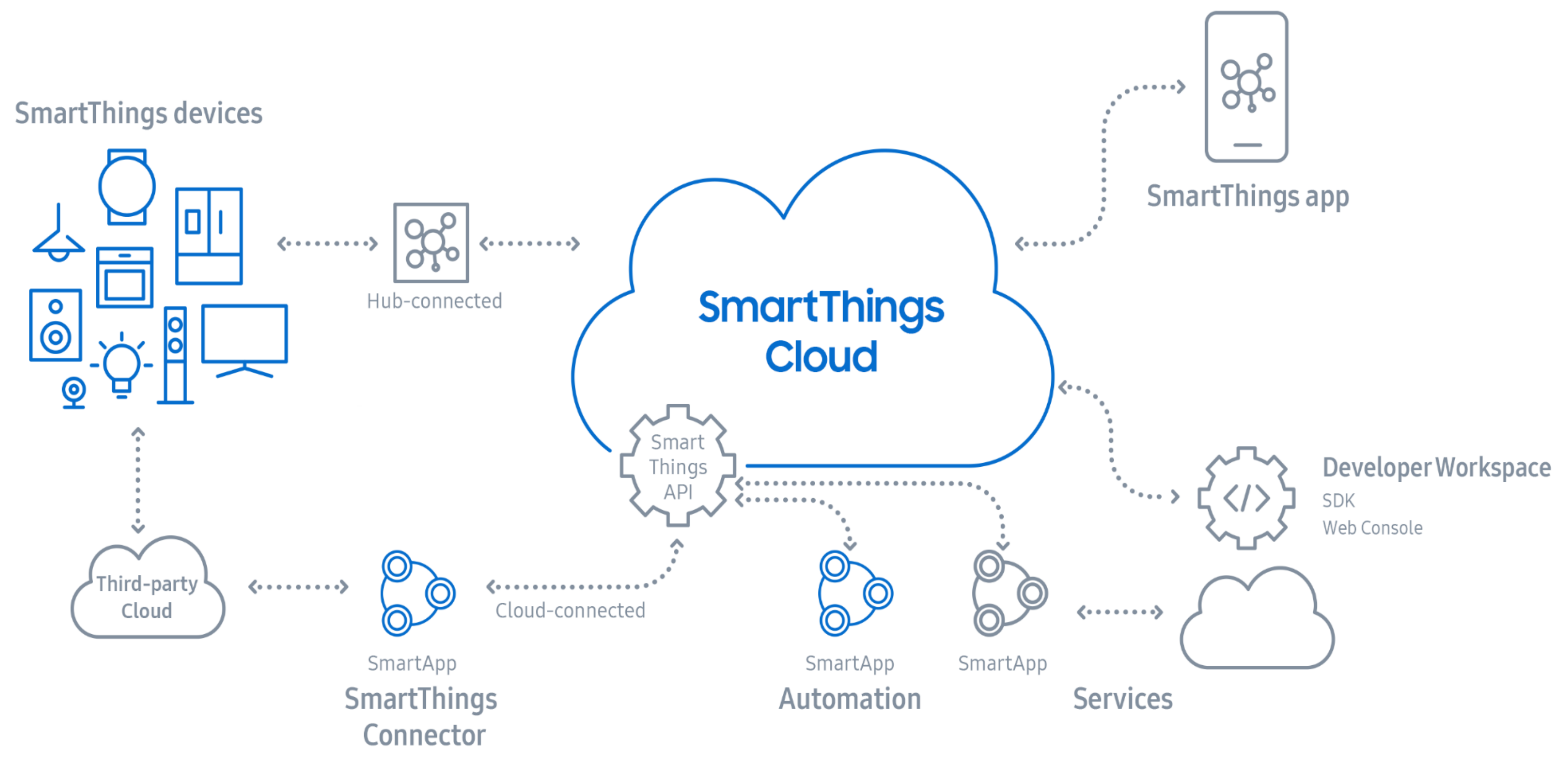
- Physical entity: all connected things in a connected home
- Attributes: Light, Temperature, Access (door, locks), Power, Water, Energy, Occupancy, Audio/Video and more
- Edge IoT Device Components: Sensor and Actuator
- Application logic & Data storage: HS3 software
- User interaction: Mobile app
- Internet dependency: internet-dependent
- Type of invocation: Explicit and Implicit
- Interoperability: Plugins, IFTTT
- Autonomous behavior: Plugins SDK & IFTTT webservice

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

SmartThings

- SmartThings hub, SmartThings mobile app
- Hub connects to home's router
- compatible with ZigBee, Z-Wave, IP-accessible devices
- Mobile app allows users to control, automate, and monitor their home environment
- Data and personal information stored in cloud



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

SmartThings - Characteristics

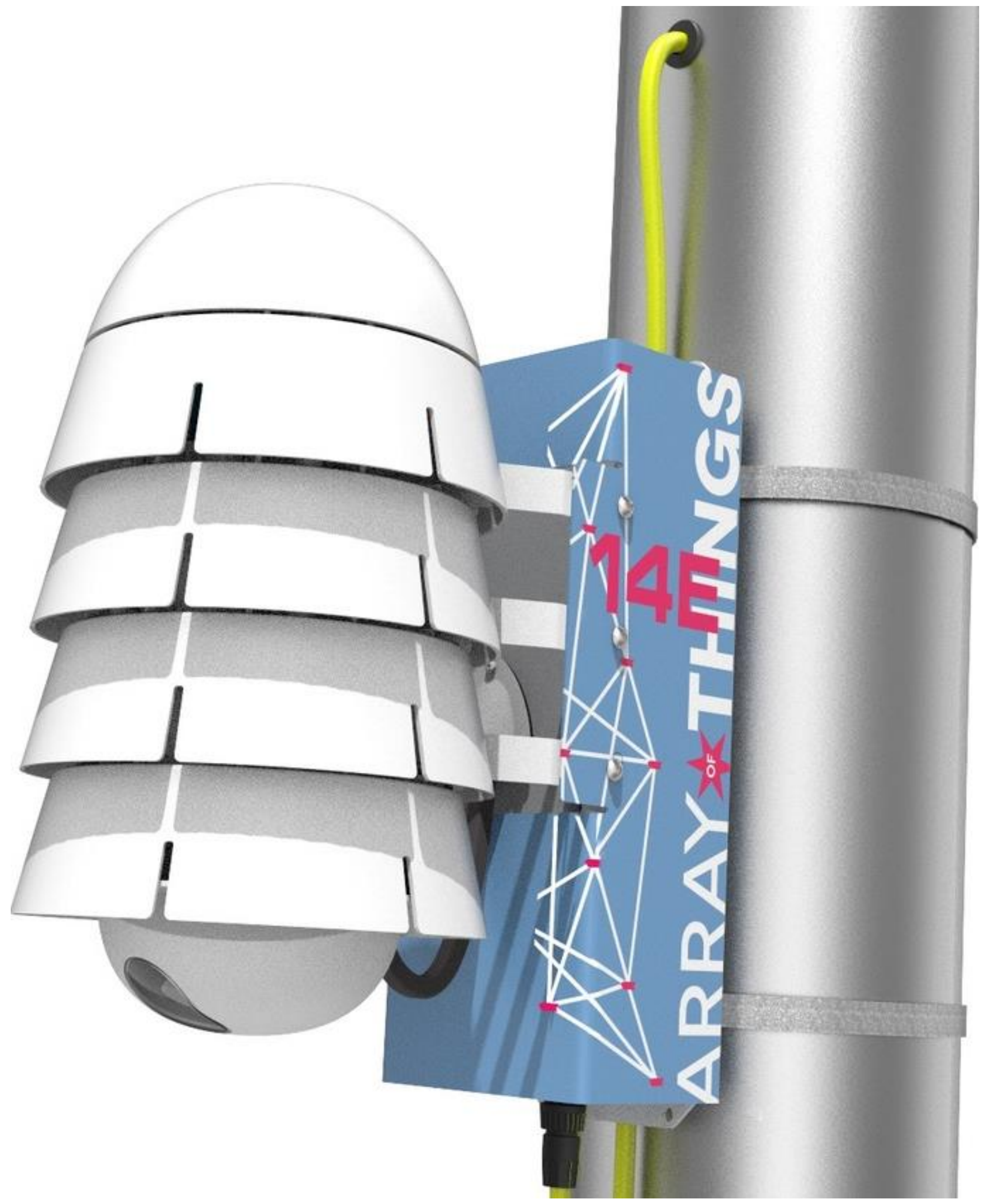
- Physical entity: all connected things in a connected home
- Attributes: Light, Temperature, Access (door, locks), Power, Water, Energy, Occupancy, Audio/Video and more
- Edge IoT Device Components: Sensor and Actuator
- Application logic & Data storage: SmartThings Cloud
- User interaction: Mobile app and Web IDE
- Internet dependency: internet-dependent
- Type of invocation: Explicit and Implicit
- Interoperability: SmartApps
- Autonomous behavior: SmartApps & IFTTT webservice

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Array of Things

- Collect real-time data (temperature, pressure, pedestrian & vehicle traffic, etc.) for research & public use.
- collaborative effort among scientists, universities, local government & communities
- Data published in open DB
- Available as bulk download / API
- Privacy protection built into the design of the sensors



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Array of Things - Characteristics

- Physical entity: city (environment)
- Attributes: Temperature, barometric pressure, light, vibration, carbon monoxide, nitrogen dioxide, sulfur dioxide, ozone, ambient sound intensity, and pedestrian and vehicle traffic
- Edge IoT Device Components: Sensor
- Application logic: Waggle Node & the cloud
- Data storage: Waggle (raw), cloud (aggregated), public DB (analyzed)
- User interaction: Management console, web portal
- Internet dependency: internet-dependent
- Type of invocation: Explicit
- Interoperability: No interoperability with IoT devices
- Autonomous behavior: No

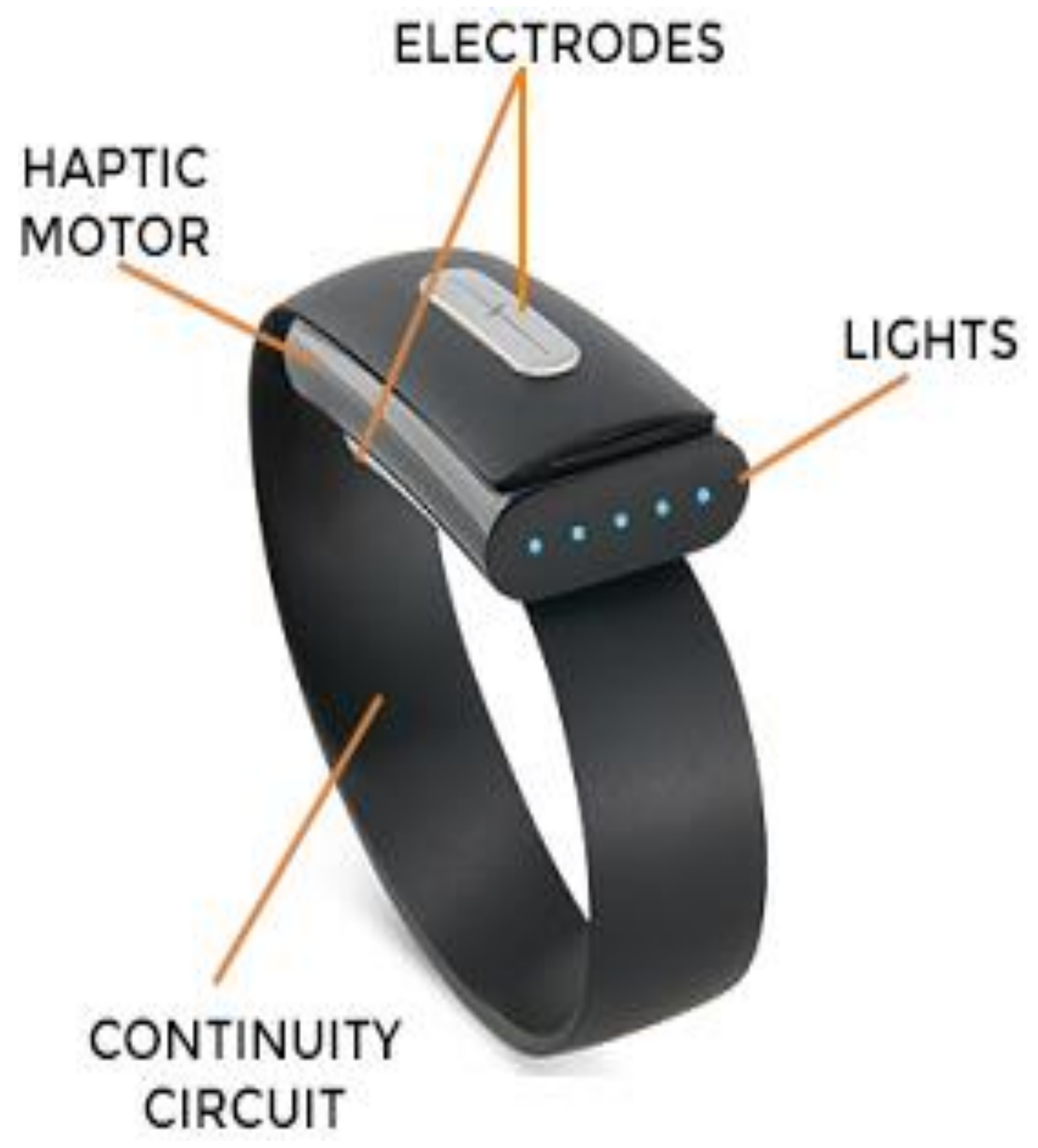
Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Nymi Band

- User’s cardiac rhythm or unique heartbeat to authenticate user and then relays identity to user’s devices via Bluetooth

- At a high level, there are two types of apps interacting with Nymi Band:
  - Nymi Band mobile application
  - Nymi-Enabled Application (NEA) based on Nymi SDK and API

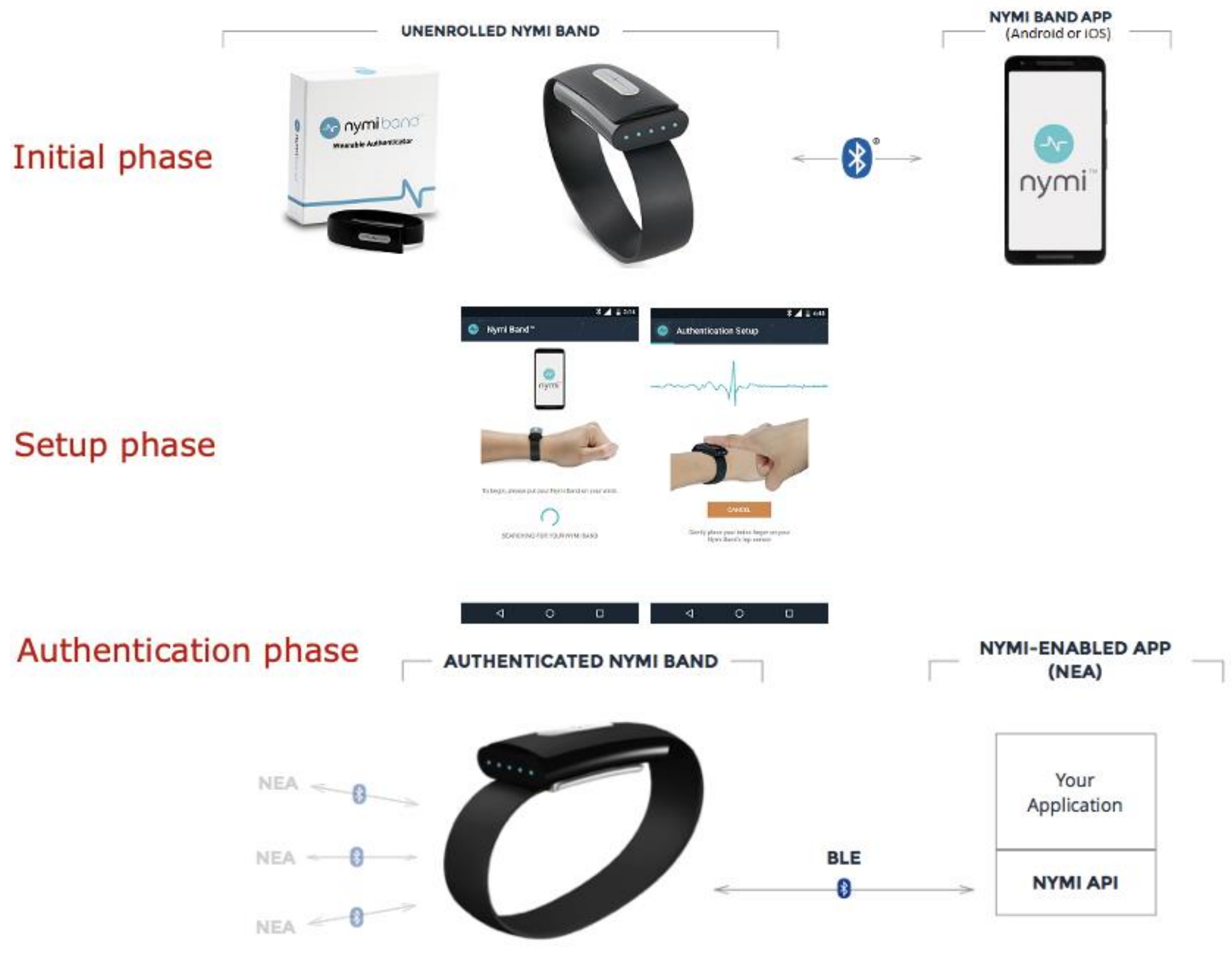


Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Nymi Band - Characteristics

- Physical entity: Human body
- Attributes: Heart signature
- Edge IoT Device Components: Sensor
- Application logic: Nymi Band
- Data storage: Nymi Band
- User interaction: vibration, LED lights
- Internet dependency: internet-independent
- Type of invocation: Explicit
- Interoperability: Nymi SDK
- Autonomous behavior: No

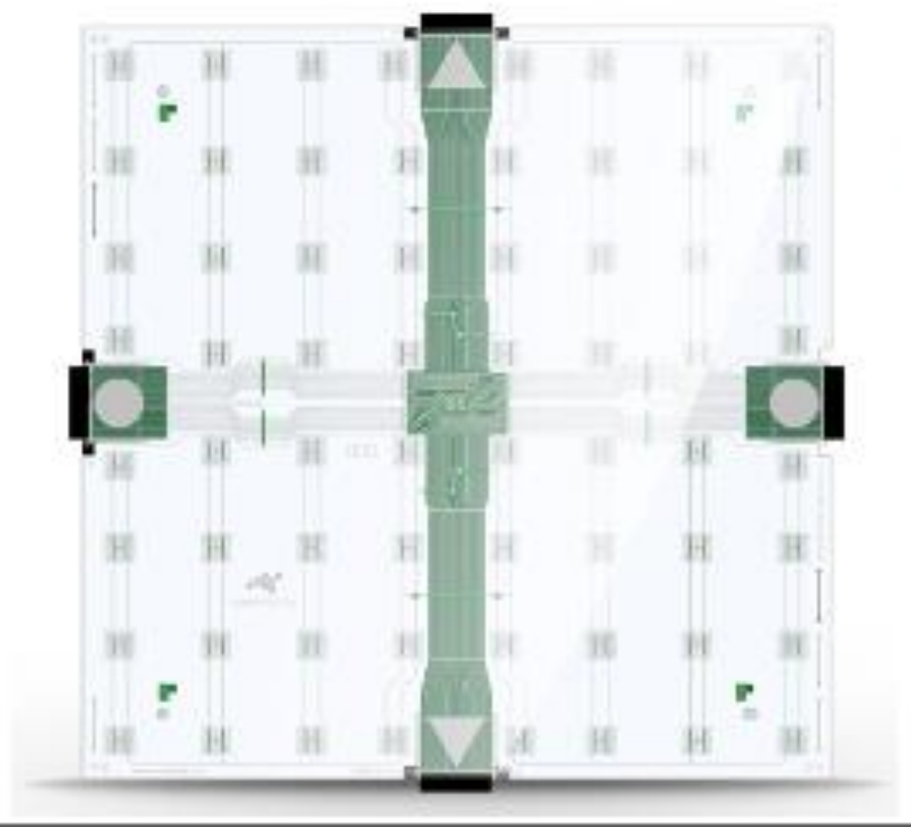


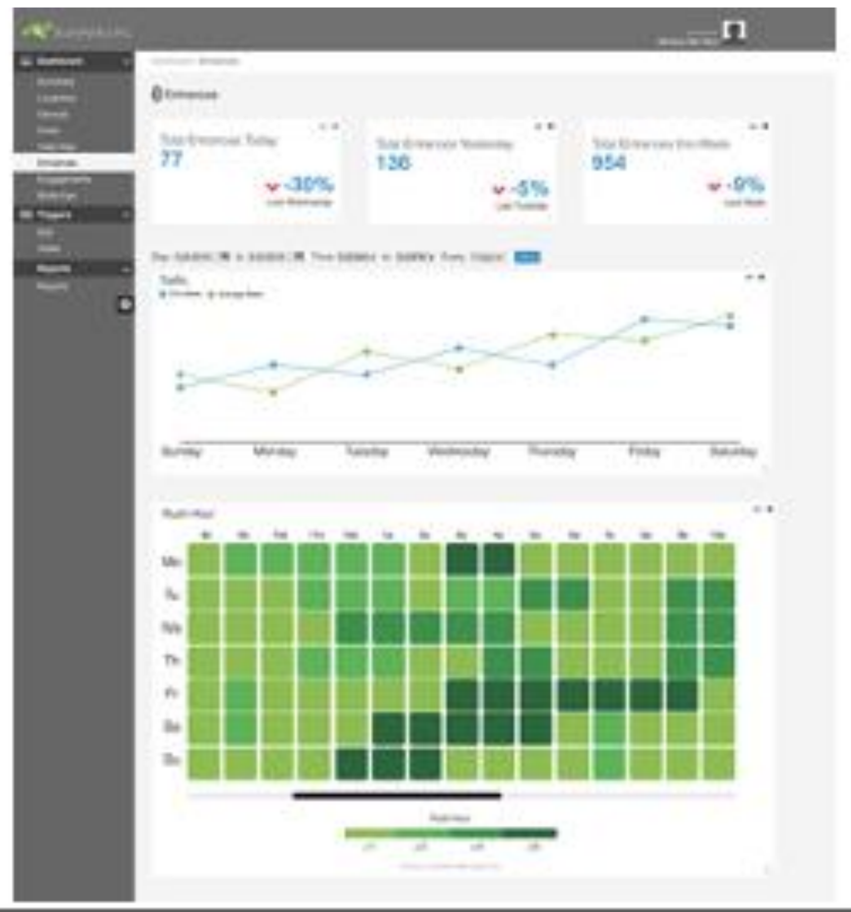
Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Scanalytics smart floor sensors

- Smart floor sensors for commercial buildings, retail and events
- Measure foot traffic to optimize the use of space and enhance visitor experience





Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Scanalytics smart floor sensors - Characteristics

- Physical entity: Floor
- Attributes: Occupancy, Bottleneck Areas, Abandonment Rates, Conversion Rates, Wait Times, Rush Hours, Engagement Times, Traffic Flow
- Edge IoT Device Components: Sensor
- Application logic: Cloud
- Data storage: Cloud
- User interaction: Online dashboard
- Internet dependency: Internet-independent
- Type of invocation: Implicit
- Interoperability: Scanalytics SDK
- Autonomous behavior: Yes

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

MAI4CAREU | Master programmes in Artificial Intelligence 4 Careers in Europe

Brickstream

- Standalone sensor devices for People Counting, People Tracking, and Queue Management
- Stereo-vision and advanced object tracking algorithms
- Brickstream Device Manager for device management
- Centralized real-time dashboard for real-time status display, configuration, maintenance, historical & predictive graphs

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Brickstream - Characteristics

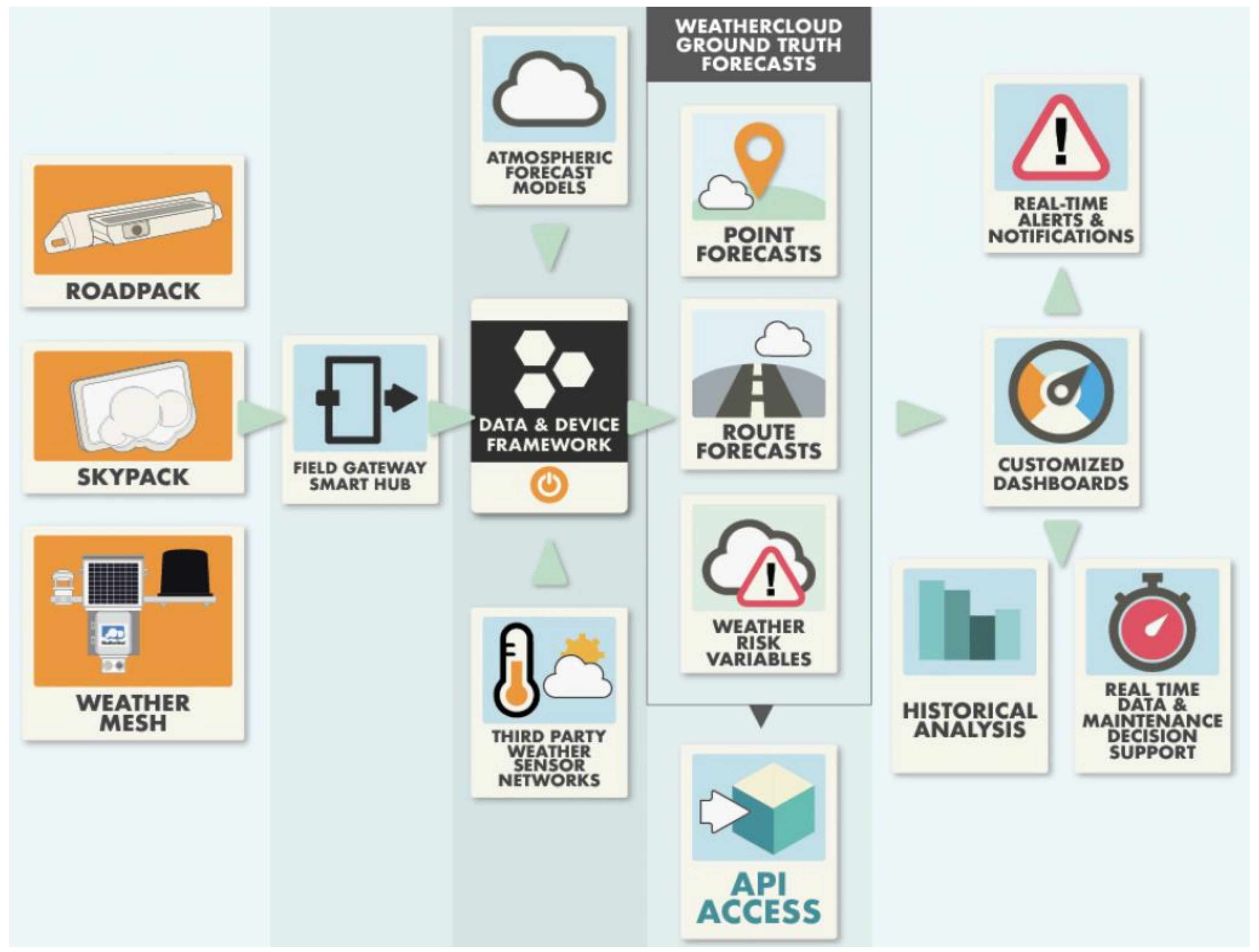
- Physical entity: Room (environment)
- Attributes: Occupancy, People counting, Height measurement, Queue management
- Edge IoT Device Components: Sensor
- Application logic: Brickstream device, data server
- Data storage: Brickstream device
- User interaction: Dashboard
- Internet dependency: Internet-independent
- Type of invocation: Explicit, device records continuously
- Interoperability: No
- Autonomous behavior: Yes

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

WeatherCloud

- WeatherCloud utilizes vehicles as weather sensors and combines that data with high-density stationary weather stations to produce an accurate surface weather model.
- The Skypack is affixed to the inside of the windshield. This sensor measures precipitation rate and type, and ambient lighting conditions.
- The Roadpack is attached to the license plate frame and measures dew point, ambient temperature, pavement temperature and road conditions.
- The in-vehicle Field Gateway smart hub aggregates the data collected by the sensors and adds vehicle dynamics data collected by the gateway itself.
- WeatherCloud's high-density WeatherMesh can be installed on existing infrastructure, such as light poles.



Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

WeatherCloud - Characteristics

- Physical entity: Vehicle
- Attributes: Precipitation rate and type, ambient lighting, dew point, ambient temperature, pavement temperature, road conditions such as wet, icy or snow-packed, wind speed and direction, rain fall, visibility
- Edge IoT Device Components: Sensor
- Application logic: Field Gateway, Cloud
- Data storage: Field Gateway, Cloud
- User interaction: Desktop app, Mobile app
- Internet dependency: Internet-dependent
- Type of invocation: Explicit
- Interoperability: No
- Autonomous behavior: Yes

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

DAQRI Smart Glasses

- The DAQRI smart glasses can operate wirelessly, using bluetooth and wi-fi technology to stream audio and video, with a wait-mounted control box providing most of the required computing power

- DAQRI glasses come with a belt pack that contains the battery as well as the processing unit.

- DAQRI glasses are supported by a suite of augmented reality productivity apps.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

DAQRI Smart Glasses - Characteristics

- Physical entity: Industrial workplace
- Attributes: State (combination of attributes)
- Edge IoT Device Components: Sensor
- Application logic: smart glasses
- Data storage: smart glasses, server
- User interaction: smart glasses, Worksense web app
- Internet dependency: Internet-dependent
- Type of invocation: Not-specified
- Interoperability: VOS extension to develop Unity apps
- Autonomous behavior: Yes

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

MAI4CAREU

Master programmes in Artificial Intelligence 4 Careers in Europe

Quality Attributes Analysis in IoT

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Interoperability

- Hub/Gateway: The Hub/Gateway middleware node contains the logic to achieve semantic interoperability between devices.
- Cloud: The Cloud component contains the logic to achieve semantic interoperability between devices.
- Third Party Service: A third-party service is used as a mediator between two IoT applications.
- SDK: An SDK is available to develop for the platform.
- Open Database: The data measured by the IoT application is available to other systems.
- No Interoperability: The solution currently does not support interoperability with other systems

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Evolvability

- In IoT, evolvability is related to the ability to push software updates to a device via the network
- Applications with “thin” edges: sensor sends data, logic at the cloud
  - software only needs to be updated in one central location (cloud)
- Applications with “smart” edges: logic at the edge of the network
  - software updates have to be propagated to all decentralized (edge) nodes
- **Centralized or decentralized application logic??**
  - High rate of updates & number of nodes? centralized logic (cloud)
  - Low rate of updates & number of nodes decentralized logic (edge)
    - Higher performance and avilability

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Performance

- Decentralized logic => low-latency, increased performance
- Half of applications examined follow centralized intelligence:
  - Low latency is not priority
  - “eventually consistent” data is enough
- Smart home environments with stationary edge devices and reliable Internet connection => high performance even logic at the cloud (SmartThings, Google Home)
- Edge devices in motion and unstable connectivity => logic at the edges higher impact on performance (Array of Things, DAQRI)

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Scalability

- Decentralization is good for scalability, especially if the central node is not a cloud component
  - Logic in edge of network decreases workload for central node (Array of Things, Brickstream, DAQRI)
- For IoT apps with fixed number of devices or fixed number of users scalability less of a priority
  - Zebra Motionworks Sport
- Some IoT apps do not need to scale at all
  - Communicate with one or few devices at a time (Nymi)
- Scalability in the IoT becomes more interesting when interoperability between applications come into play
  - high magnitude of users and edge devices (SmartThings, Google Home)

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Availability

- Usually attributed to battery exhaustion
  - Majority of solutions analyzed either plug in to a power outlet or have pretty long battery life
    - Only DAQRI smart glasses battery lasts for some hours

- Affected by connectivity issues
  - Allow device to locally store data; upload when reliable connection established
    - Brickstream, DAQRI

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Resiliency

- Centralization of logic introduces a single point of failure
  - DAQRI smart glasses is able to continue functioning while the central node has time to recover
  - SmartThings home environment becomes inoperable if central node (cloud) fails
- Replication of central nodes or decentralization of logic can remove a single point of failure, increasing overall resiliency
- Replication of edge devices increase system resiliency
  - In WeatherCloud multiple nodes measure the same thing; if a car fails, weather conditions obtained from other WeatherCloud enabled cars
  - If Google Nest Thermostat fails, particular house becomes inoperable

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Security

- Hard to say anything about security measures that IoT applications use => limited info given

- IoT apps usually use widely accepted security protocols
  - Unsuitable for resource-limited IoT devices

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Privacy

- Privacy is promised by assuring the users that only necessary data is gathered and not personal data (data minimization).
- Decentralization of data storage is deemed privacy friendly.
- There are some solutions where privacy does not play a role at all because of the nature of the data being gathered and processed.
- There is a trade-off between Privacy and Functionality in the IoT.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action No 2020-EU-IA-0087, co-financed by the EU CEF Telecom under GA nr. INEA/CEF/ICT/A2020/2267423

Summary

- ❑ Design Requirements for IoT Applications
- ❑ Software architectural styles in IoT
- ❑ Major IoT Applications Domains
- ❑ Selected IoT Applications Analysis
- ❑ Quality Attributes Analysis in IoT

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423