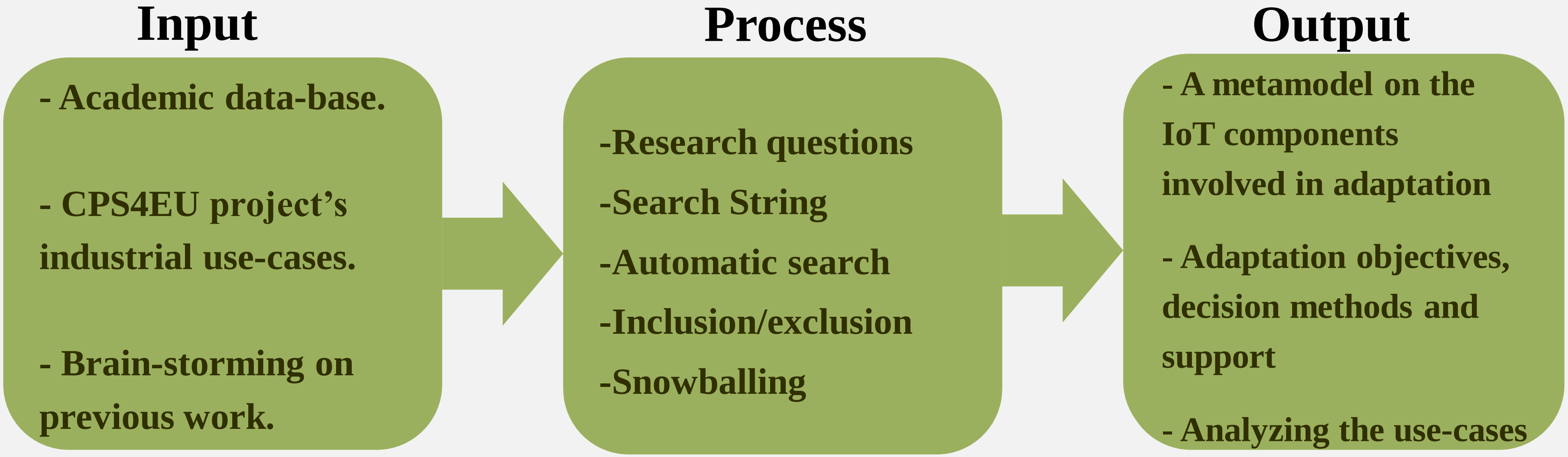
Middleware Support for Self-adaptive IoT Systems

A Systematic Literature Review

Mahyar T. Moghaddam **Eric Rutten** **Guillaume Giraud**

Context and Process

Classifying and analyzing studies on self-adaptation mechanisms for IoT, by focusing on platforms that facilitate such self-adaptation.



Input

- Academic data-base.
- CPS4EU project's industrial use-cases.
- Brain-storming on previous work.

Process

-Research questions

-Search String

-Automatic search

-Inclusion/exclusion

-Snowballing

Output

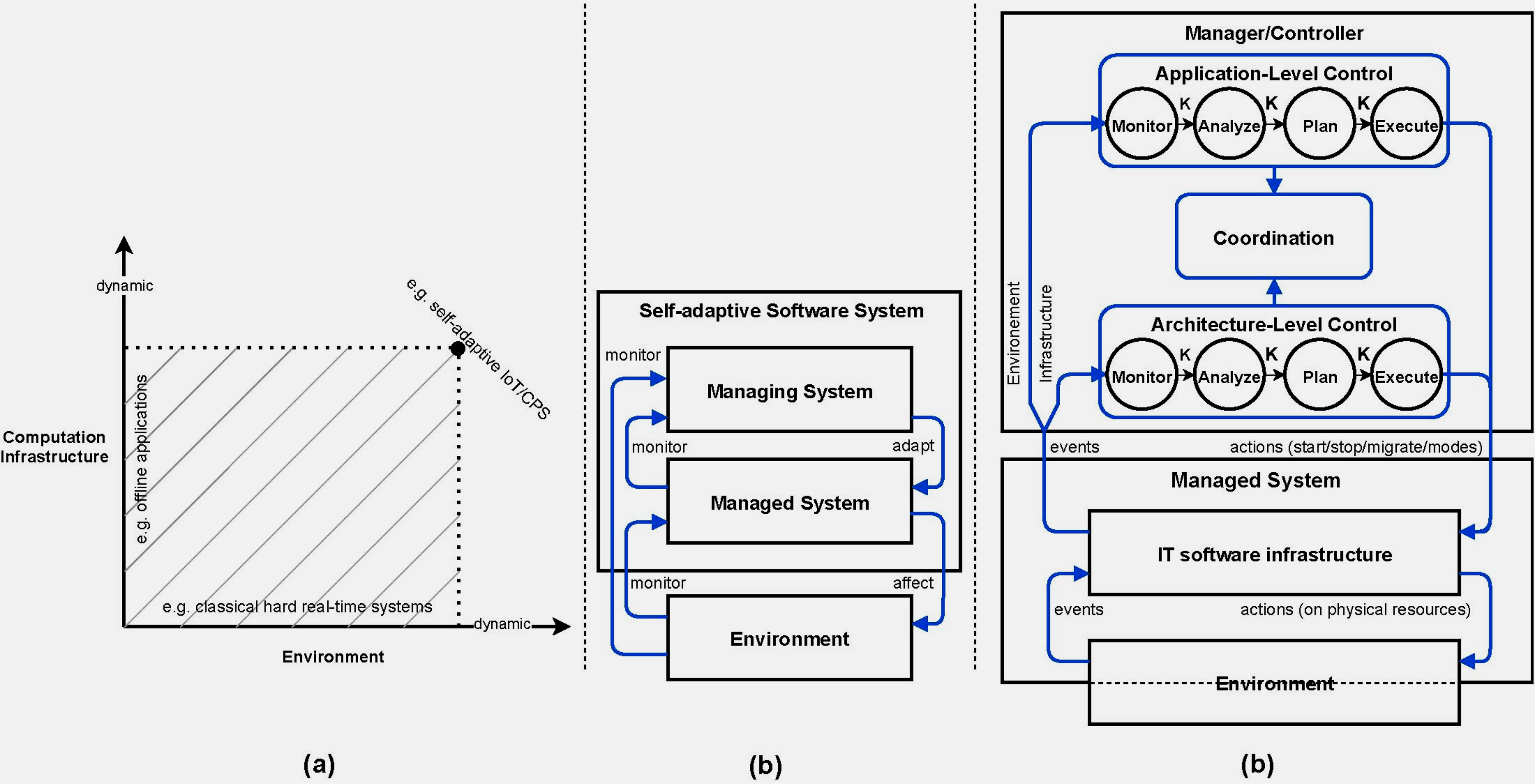
- A metamodel on the IoT components involved in adaptation
- Adaptation objectives, decision methods and support
- Analyzing the use-cases

RQ1. What are the objectives of self-adaptation in IoT systems?

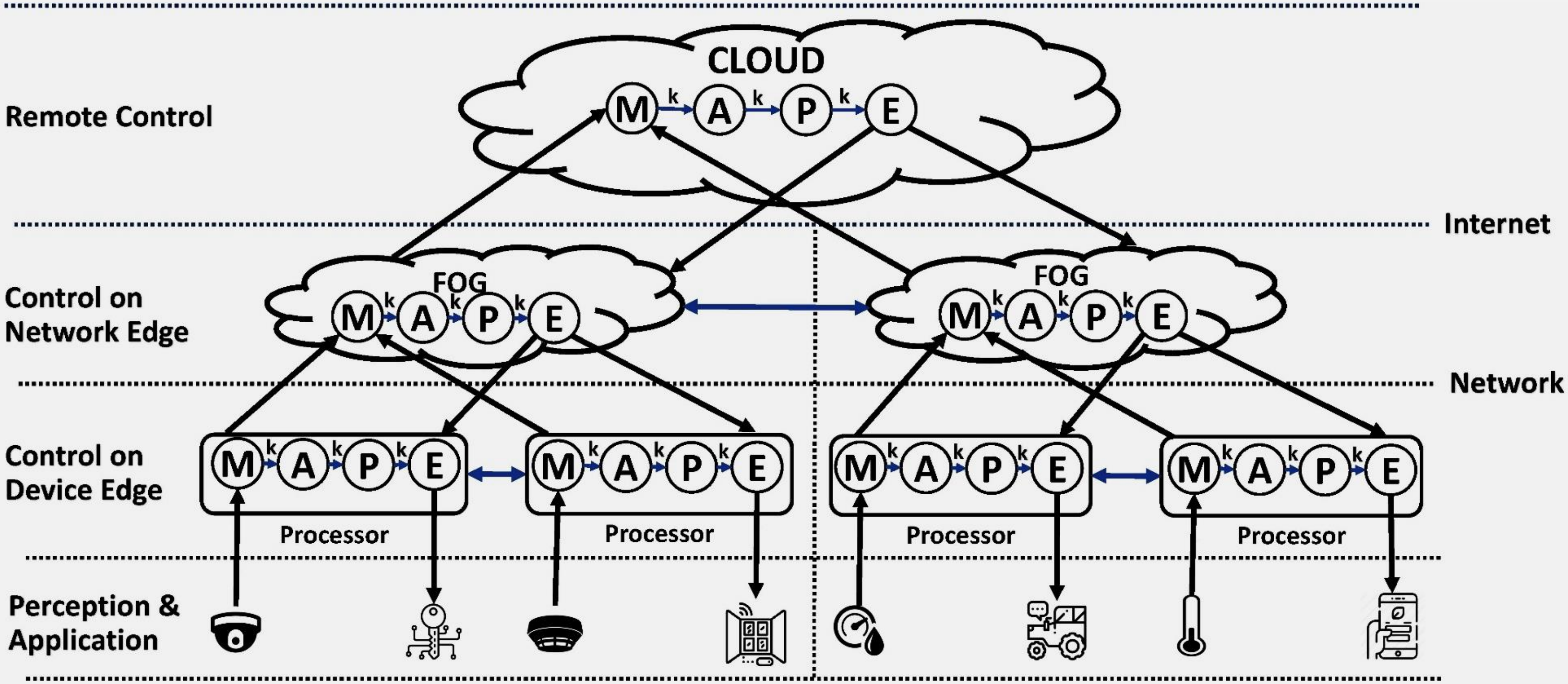
RQ2. What are the adaptation decisions methods that can be adopted?

RQ3. What methods, models, tools are proposed for IoT self-adaptation?

Self-adaptation Concerns (RQ1)


dynamic
e.g. self-adaptive IoT/CPS
Computation Infrastructure
e.g. offline applications
e.g. classical hard real-time systems
dynamic
Environment
(a)
Self-adaptive Software System
monitor
Managing System
monitor
adapt
Managed System
monitor
affect
Environment
(b)
Manager/Controller
Application-Level Control
Monitor
K
Analyze
K
Plan
K
Execute
Coordination
Architecture-Level Control
Monitor
K
Analyze
K
Plan
K
Execute
Environement
Infrastructure
events
actions (start/stop/migrate/modes)
Managed System
IT software infrastructure
events
actions (on physical resources)
Environment
(b)

Self-adaptation Concerns (RQ1)

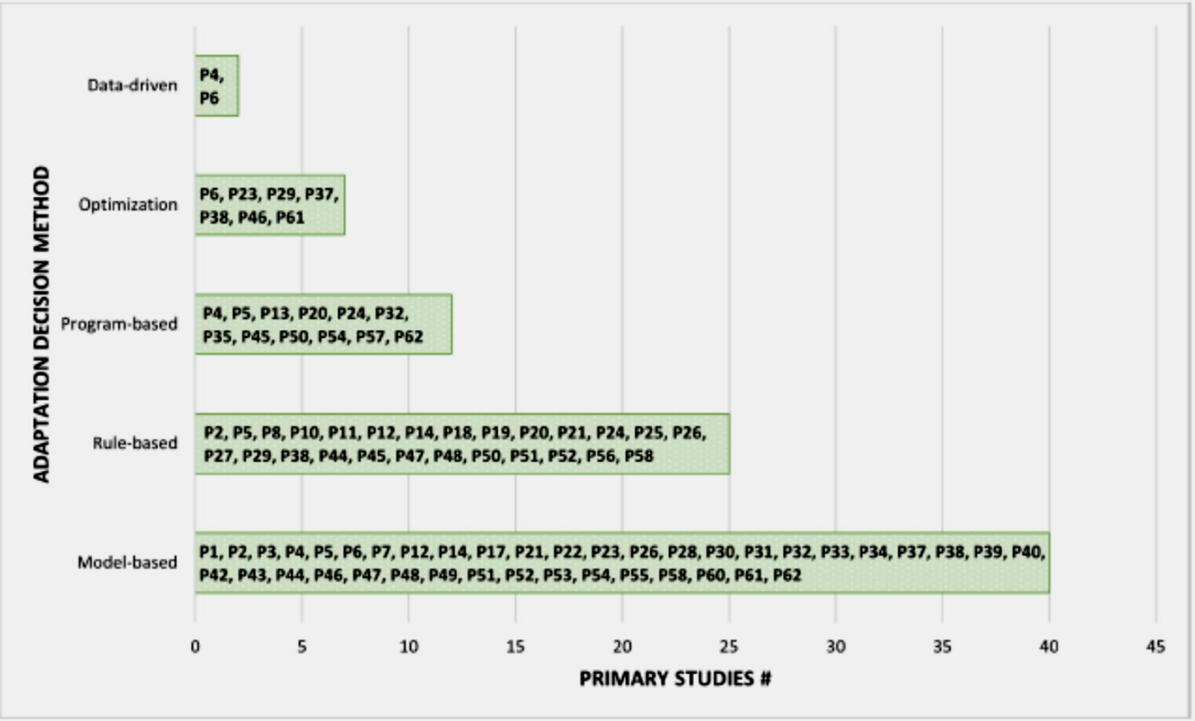


Answer to RQ1:

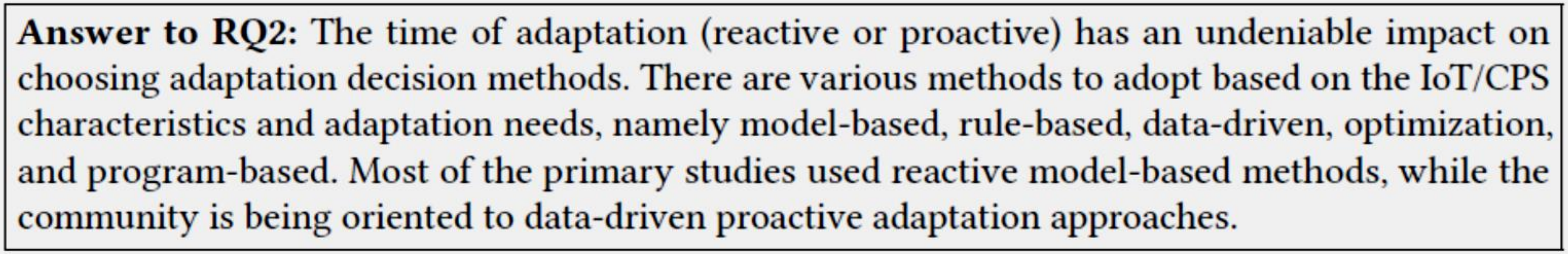
The self-adaptation can occur due to changes required in the system, its surrounding environment, and their coordination. According to our literature-based proposed approach, the system's architectural changes are handled by autonomic control, while the functional control manages the changes related to the environmental context. Our study revealed that while most of the primary studies focus on the adaptation aspects of functional control, the autonomic control topic is recently getting more attention.

Self-Adaptation Decision (RQ2)

- **Time:** Reactive or Proactive
- Decision Methods:



Model-based: Architecture reconfiguration, Feedback loops, Agent-based models, Mathematic models.



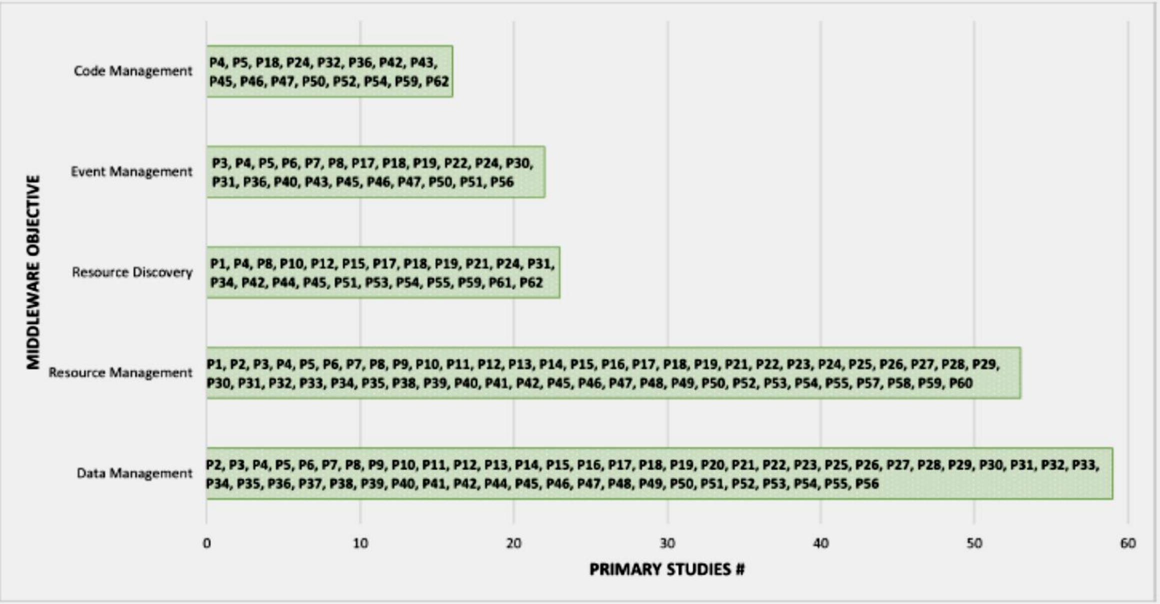
Answer to RQ2: The time of adaptation (reactive or proactive) has an undeniable impact on choosing adaptation decision methods. There are various methods to adopt based on the IoT/CPS characteristics and adaptation needs, namely model-based, rule-based, data-driven, optimization, and program-based. Most of the primary studies used reactive model-based methods, while the community is being oriented to data-driven proactive adaptation approaches.

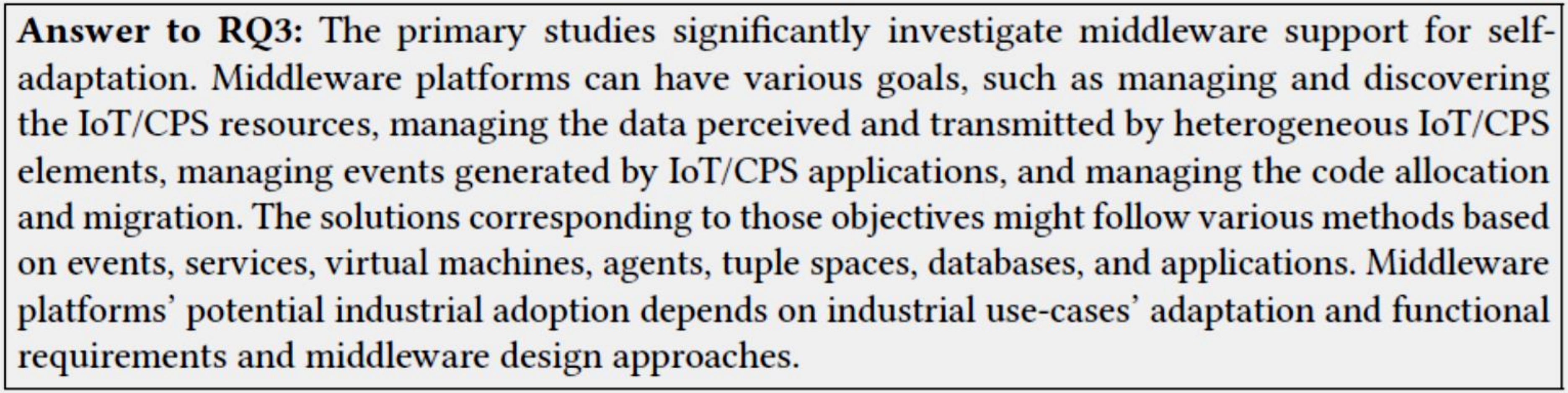
Middleware Support (RQ3)

Middleware Solutions:

- Event-based
- Service-oriented
- Virtual machine-based
- Agent-based
- Tuple-spaces
- Database-oriented
- Application specific

Middleware Goals

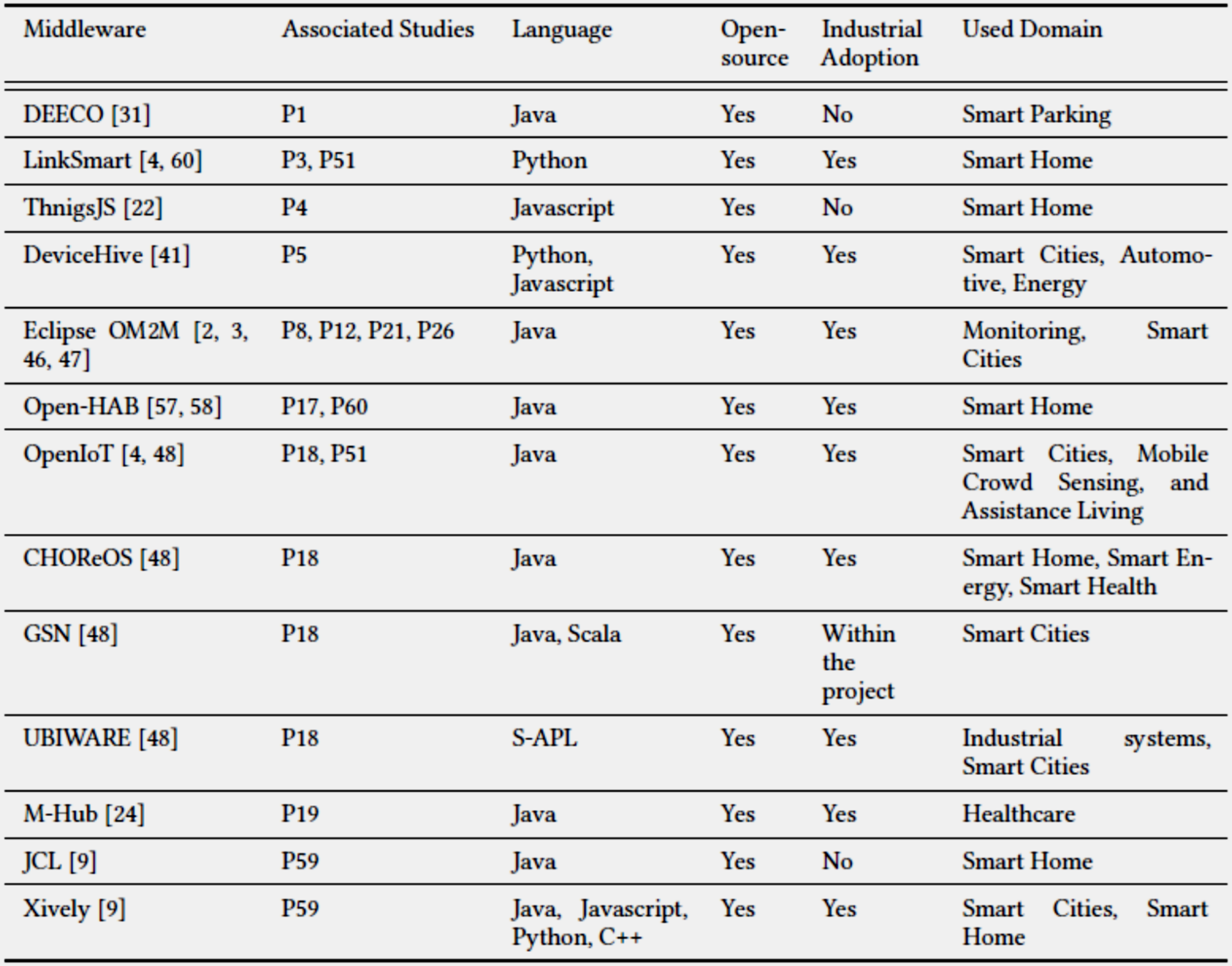




Answer to RQ3: The primary studies significantly investigate middleware support for self-adaptation. Middleware platforms can have various goals, such as managing and discovering the IoT/CPS resources, managing the data perceived and transmitted by heterogeneous IoT/CPS elements, managing events generated by IoT/CPS applications, and managing the code allocation and migration. The solutions corresponding to those objectives might follow various methods based on events, services, virtual machines, agents, tuple spaces, databases, and applications. Middleware platforms' potential industrial adoption depends on industrial use-cases' adaptation and functional requirements and middleware design approaches.

Middleware Support (RQ3)

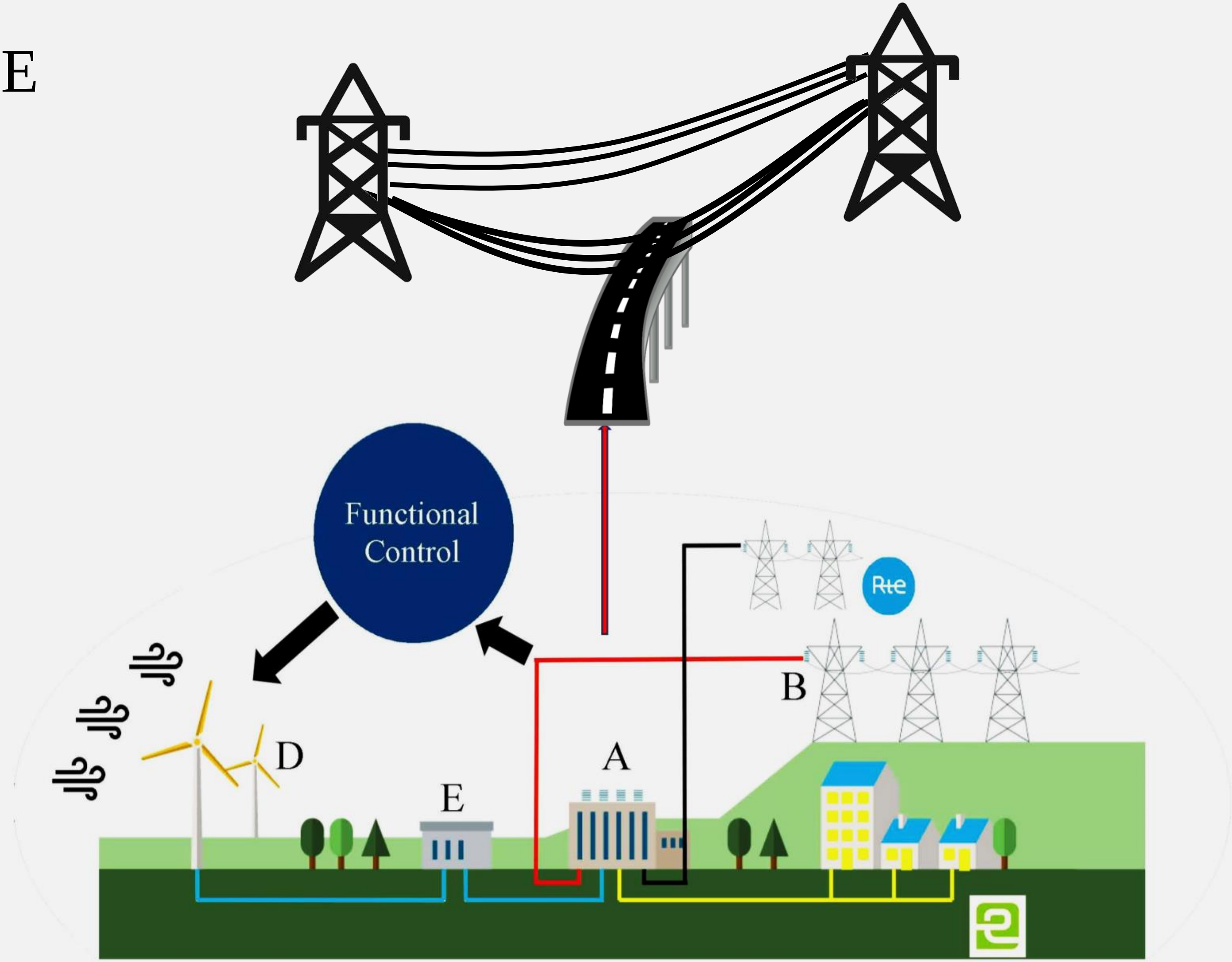
- Platforms



Middleware	Associated Studies	Language	Open-source	Industrial Adoption	Used Domain
DEECO [31]	P1	Java	Yes	No	Smart Parking
LinkSmart [4, 60]	P3, P51	Python	Yes	Yes	Smart Home
ThnigsJS [22]	P4	Javascript	Yes	No	Smart Home
DeviceHive [41]	P5	Python, Javascript	Yes	Yes	Smart Cities, Automotive, Energy
Eclipse OM2M [2, 3, 46, 47]	P8, P12, P21, P26	Java	Yes	Yes	Monitoring, Smart Cities
Open-HAB [57, 58]	P17, P60	Java	Yes	Yes	Smart Home
OpenIoT [4, 48]	P18, P51	Java	Yes	Yes	Smart Cities, Mobile Crowd Sensing, and Assistance Living
CHOReOS [48]	P18	Java	Yes	Yes	Smart Home, Smart Energy, Smart Health
GSN [48]	P18	Java, Scala	Yes	Within the project	Smart Cities
UBIWARE [48]	P18	S-APL	Yes	Yes	Industrial systems, Smart Cities
M-Hub [24]	P19	Java	Yes	Yes	Healthcare
JCL [9]	P59	Java	Yes	No	Smart Home
Xively [9]	P59	Java, Javascript, Python, C++	Yes	Yes	Smart Cities, Smart Home

Some Industrial Use-cases

- RTE



Some Industrial Use-cases

- ACOEM

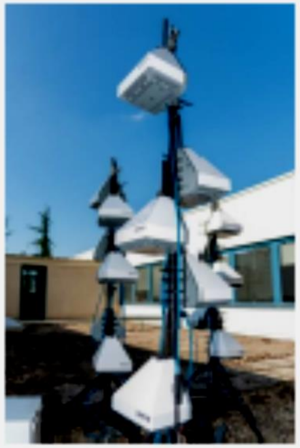


Figure 6- Photo of the Environmental PODs used to collect data

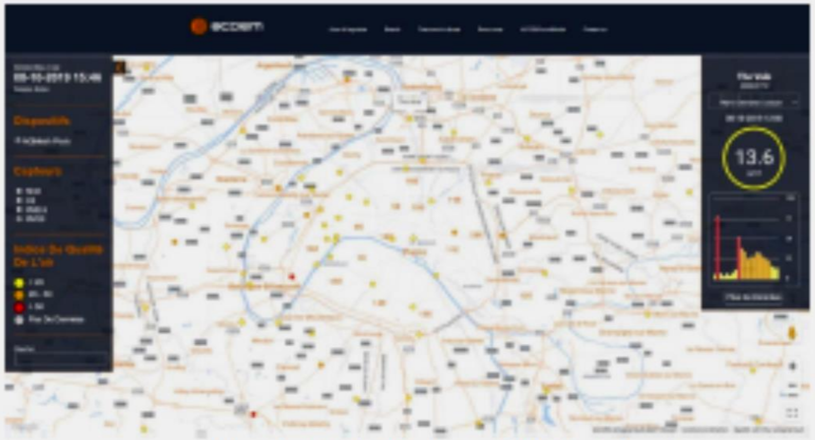


Figure 5 - Example of Environmental Quality Index map

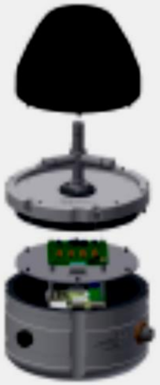


Figure 8- First version of the Gunshot Detection Pod composed of a 4 microphones antenna

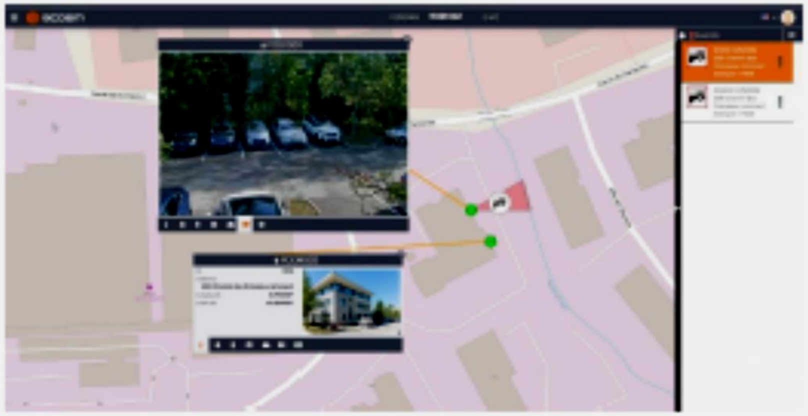


Figure 7 - Example of Gun Shot Detection System

Thanks!
Any Questions?