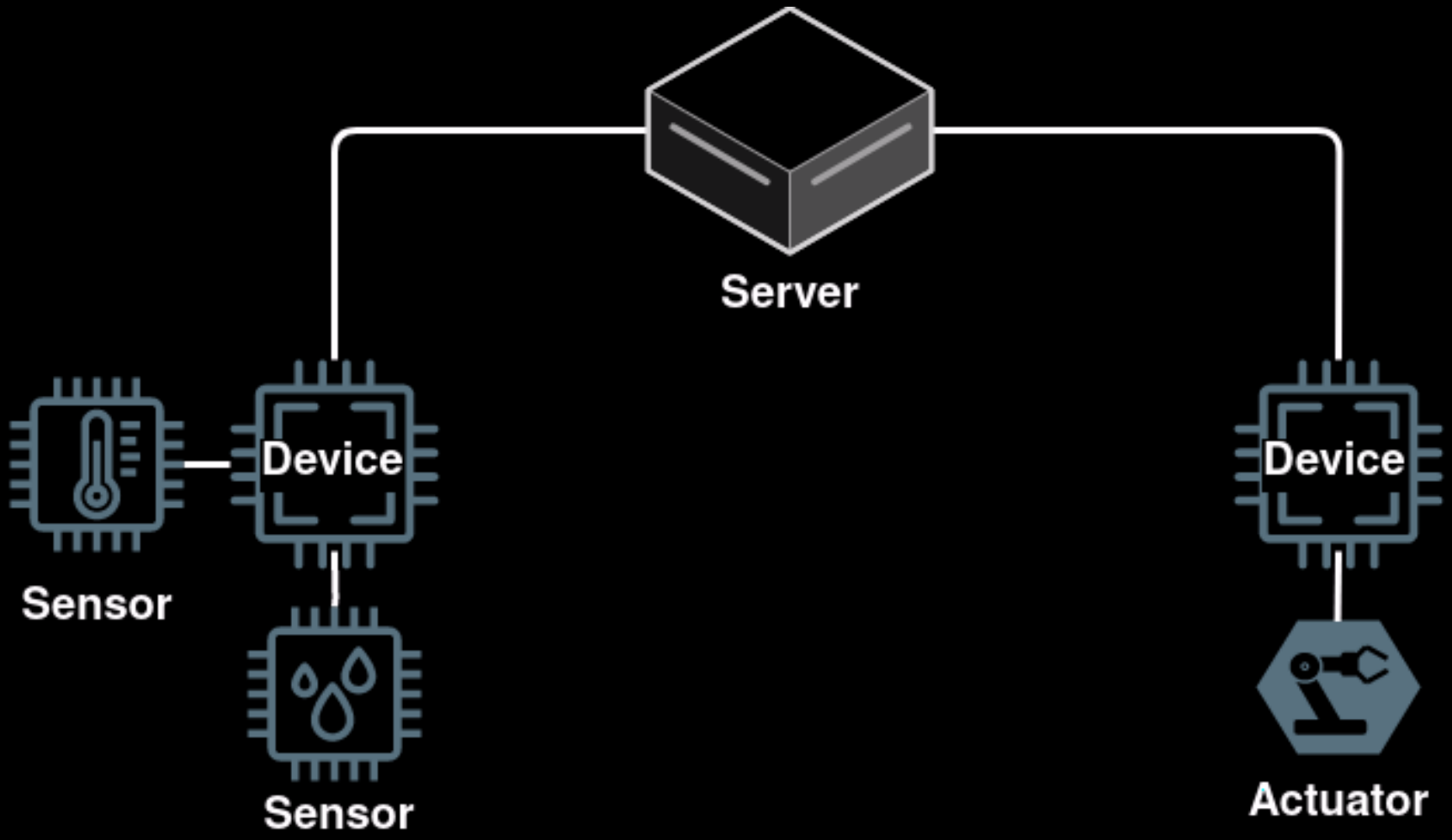
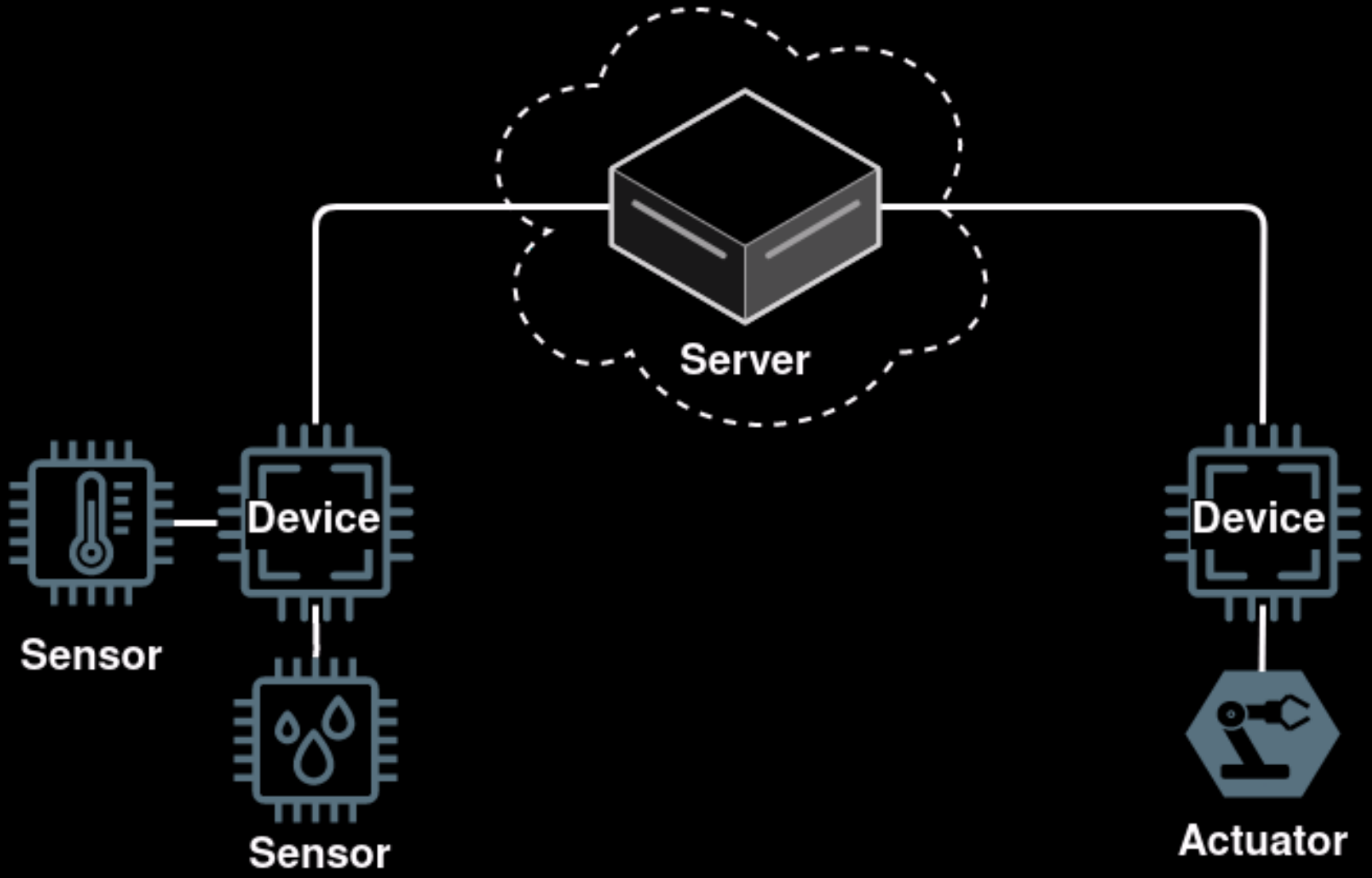
WHY IS MY SMART HOME STILL DUMB?

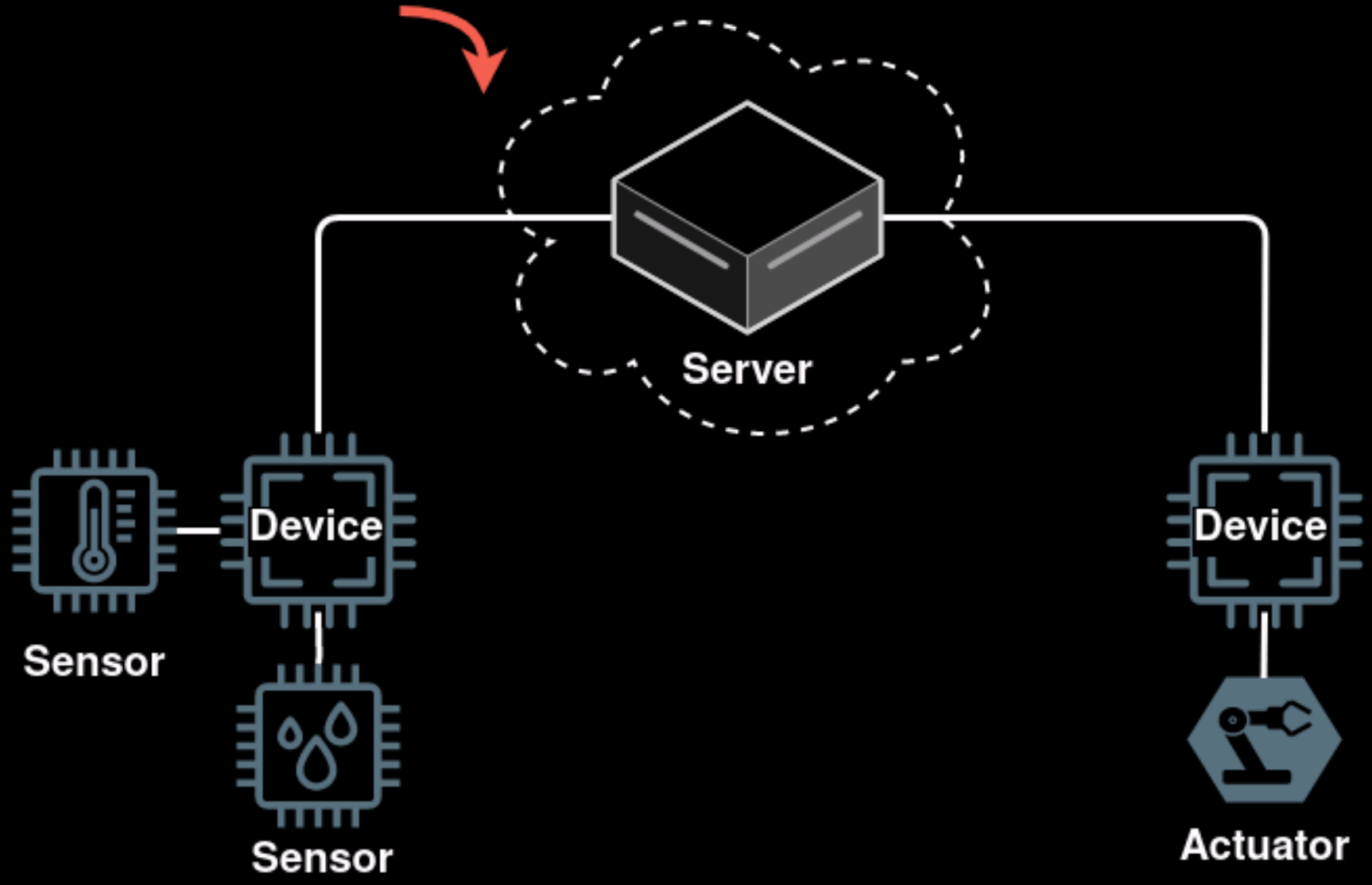
A research agenda towards secure, resilient and evolvable smart buildings

Karolina Skrivankova, Mark Handley, Steve Hailes

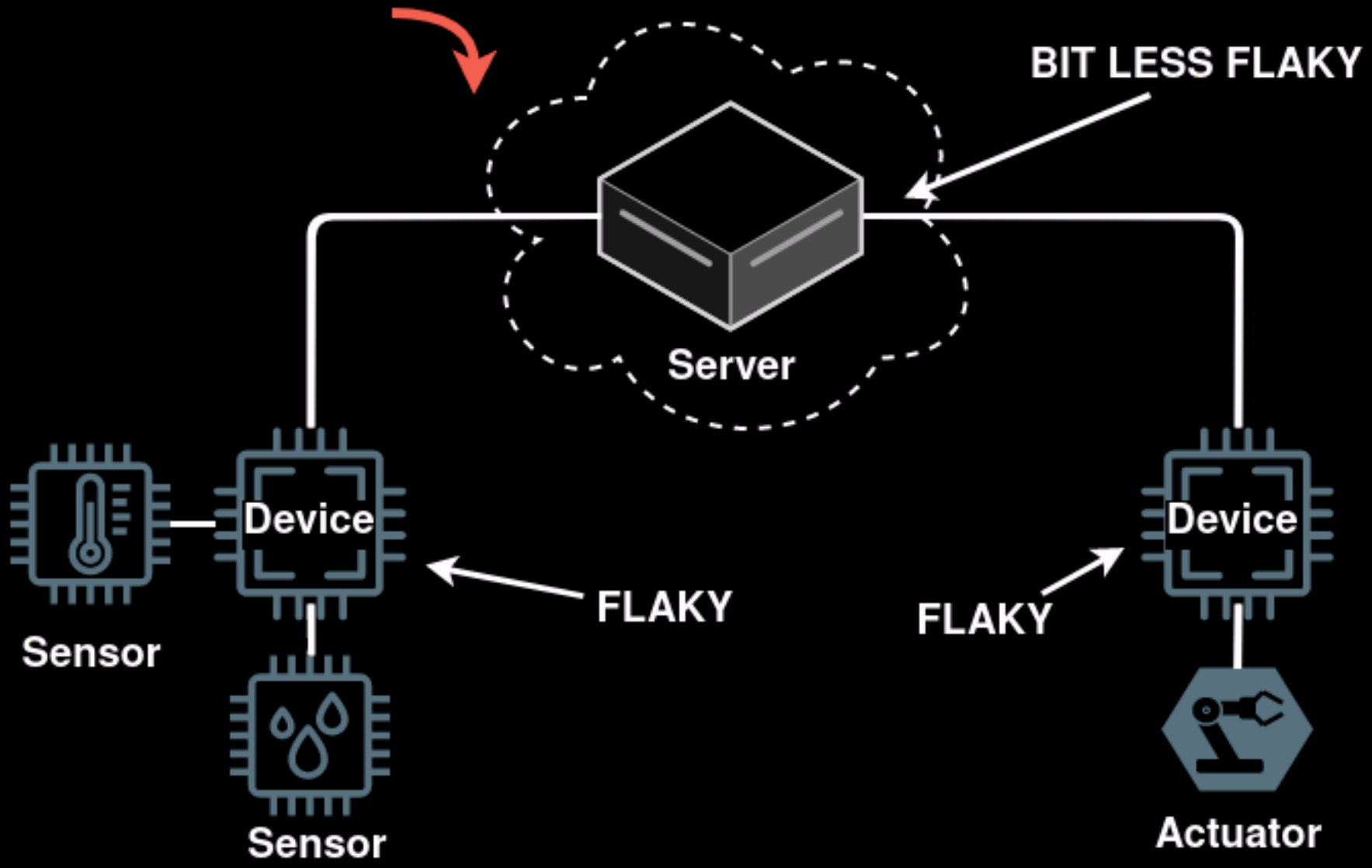

Server
Device
Sensor
Sensor
Device
Actuator


Server
Device
Device
Sensor
Sensor
Actuator

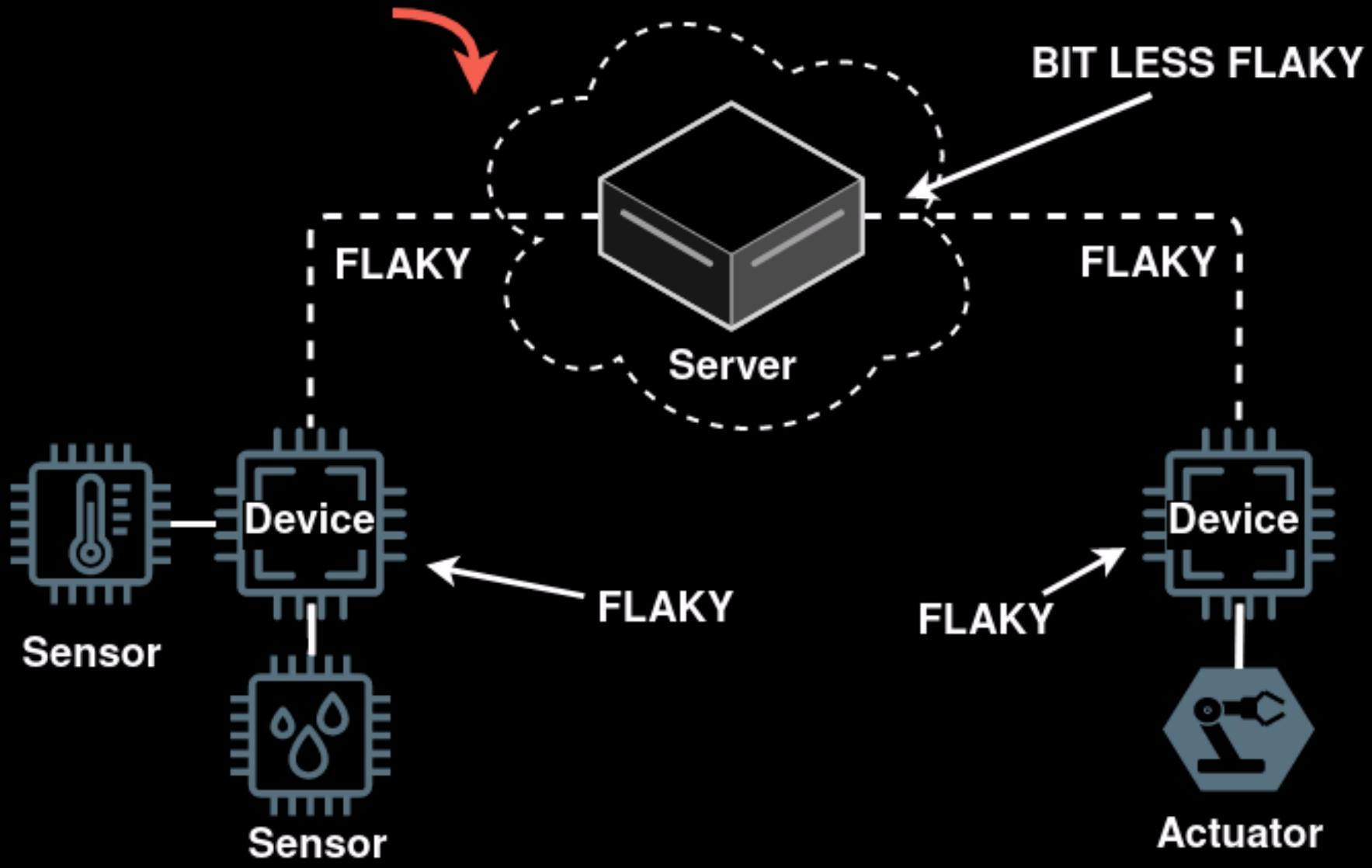
RUN A COMPLEX CONTROL POLICY FOR DECADES

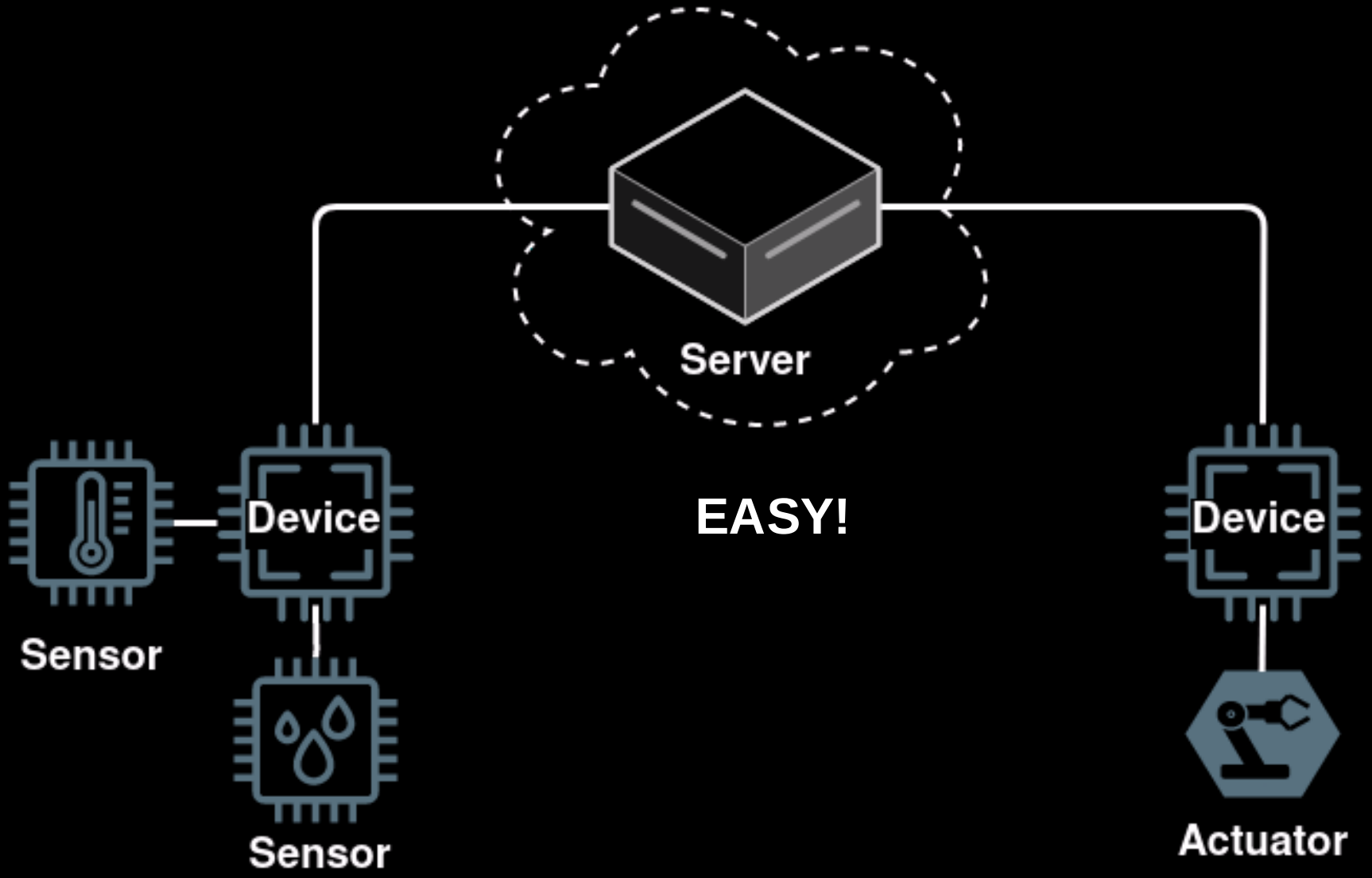


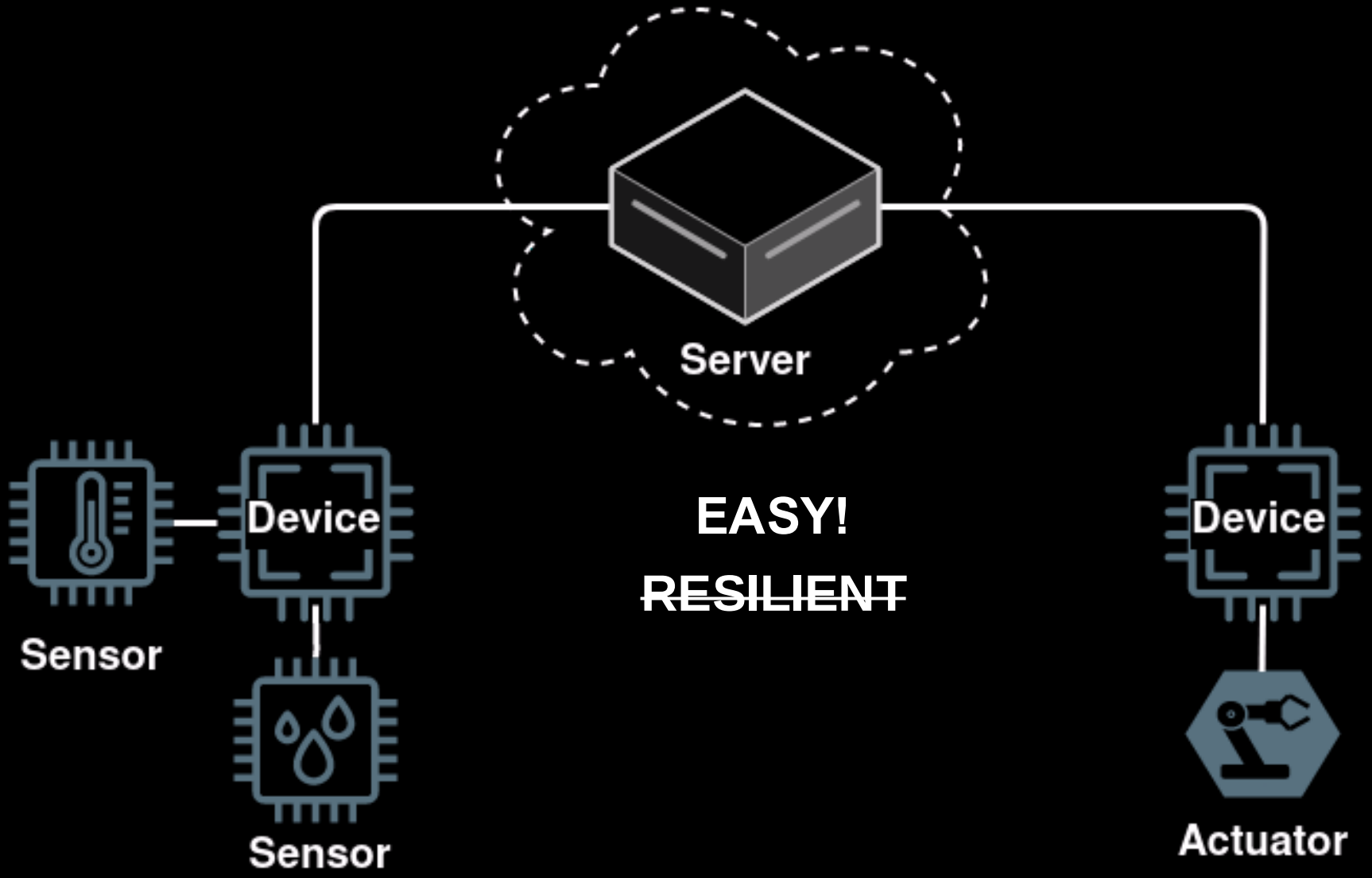
RUN A COMPLEX CONTROL POLICY FOR DECADES



RUN A COMPLEX CONTROL POLICY FOR DECADES

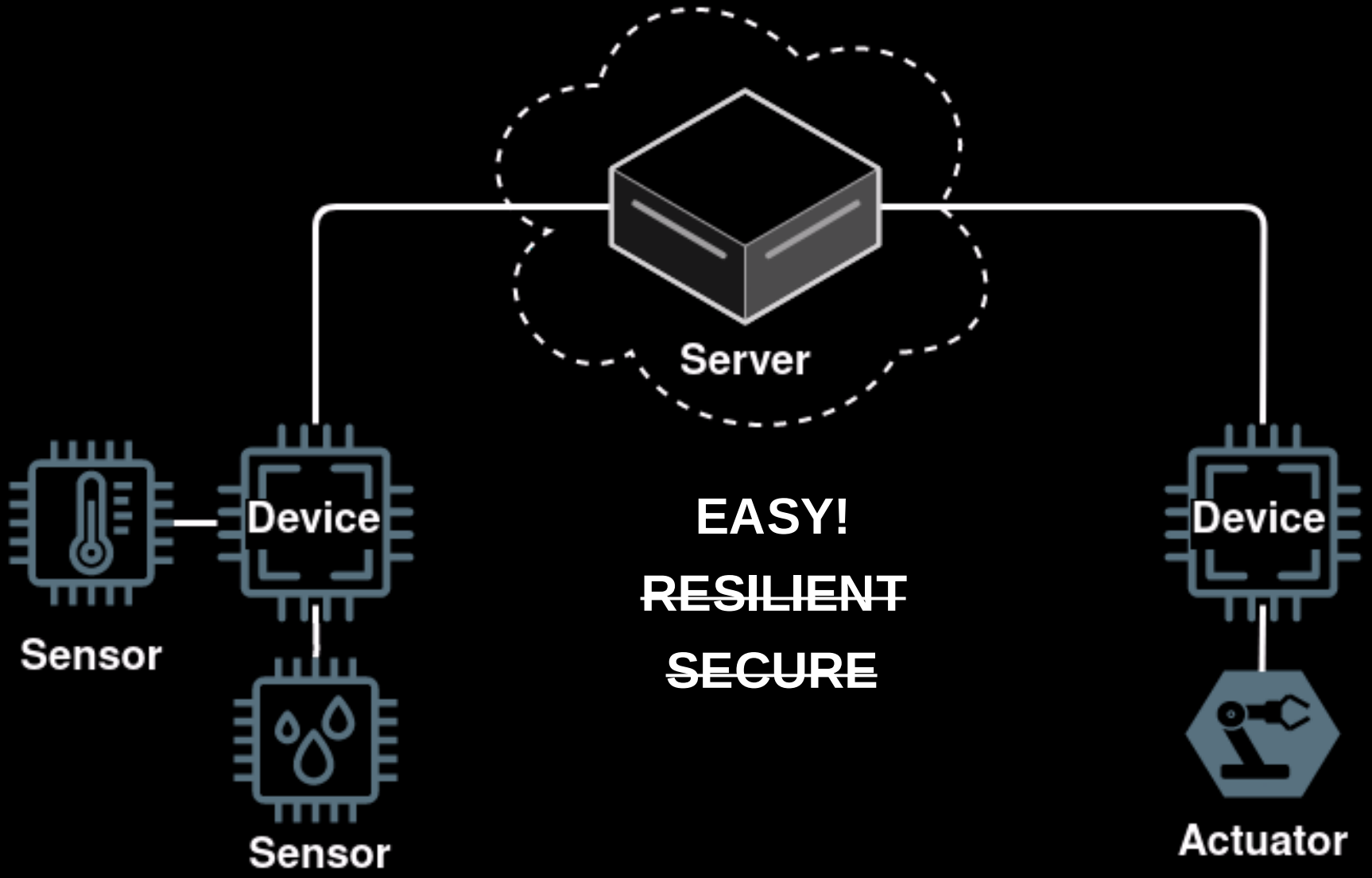



Server
Device
Sensor
Sensor
EASY!
Device
Actuator



EASY!

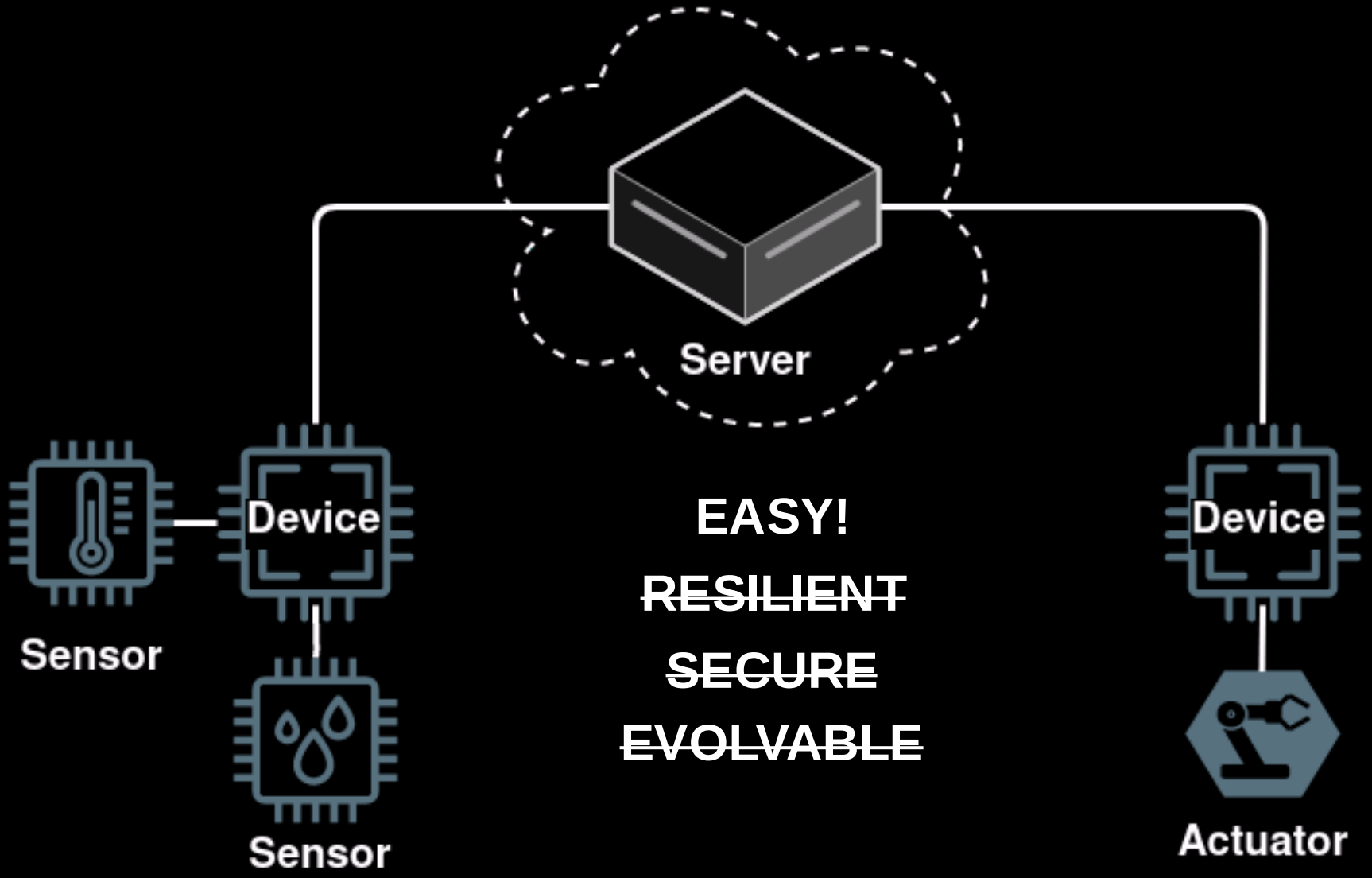
~~RESILIENT~~



EASY!

~~RESILIENT~~

~~SECURE~~

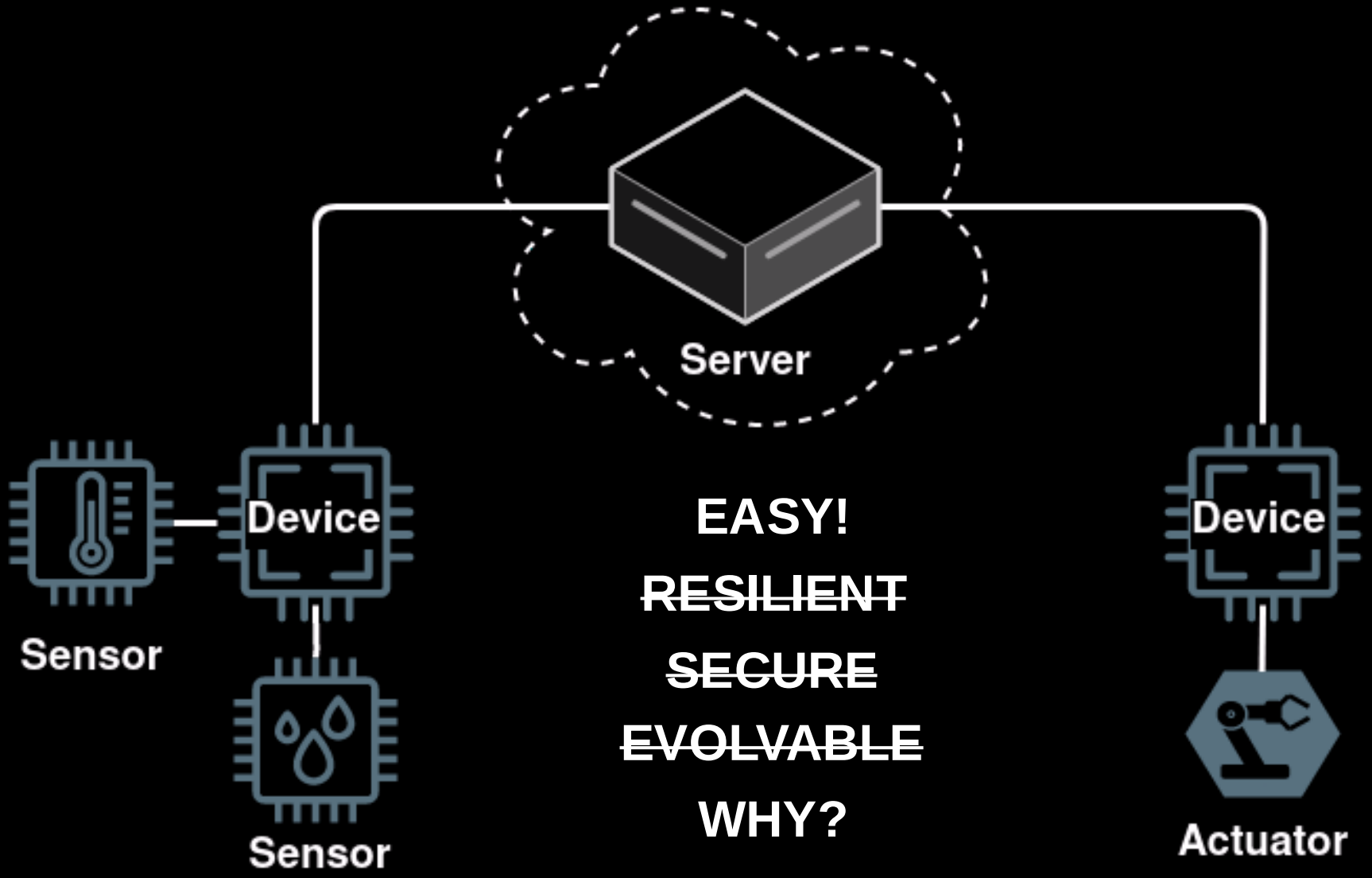


EASY!

~~RESILIENT~~

~~SECURE~~

~~EVOLVABLE~~



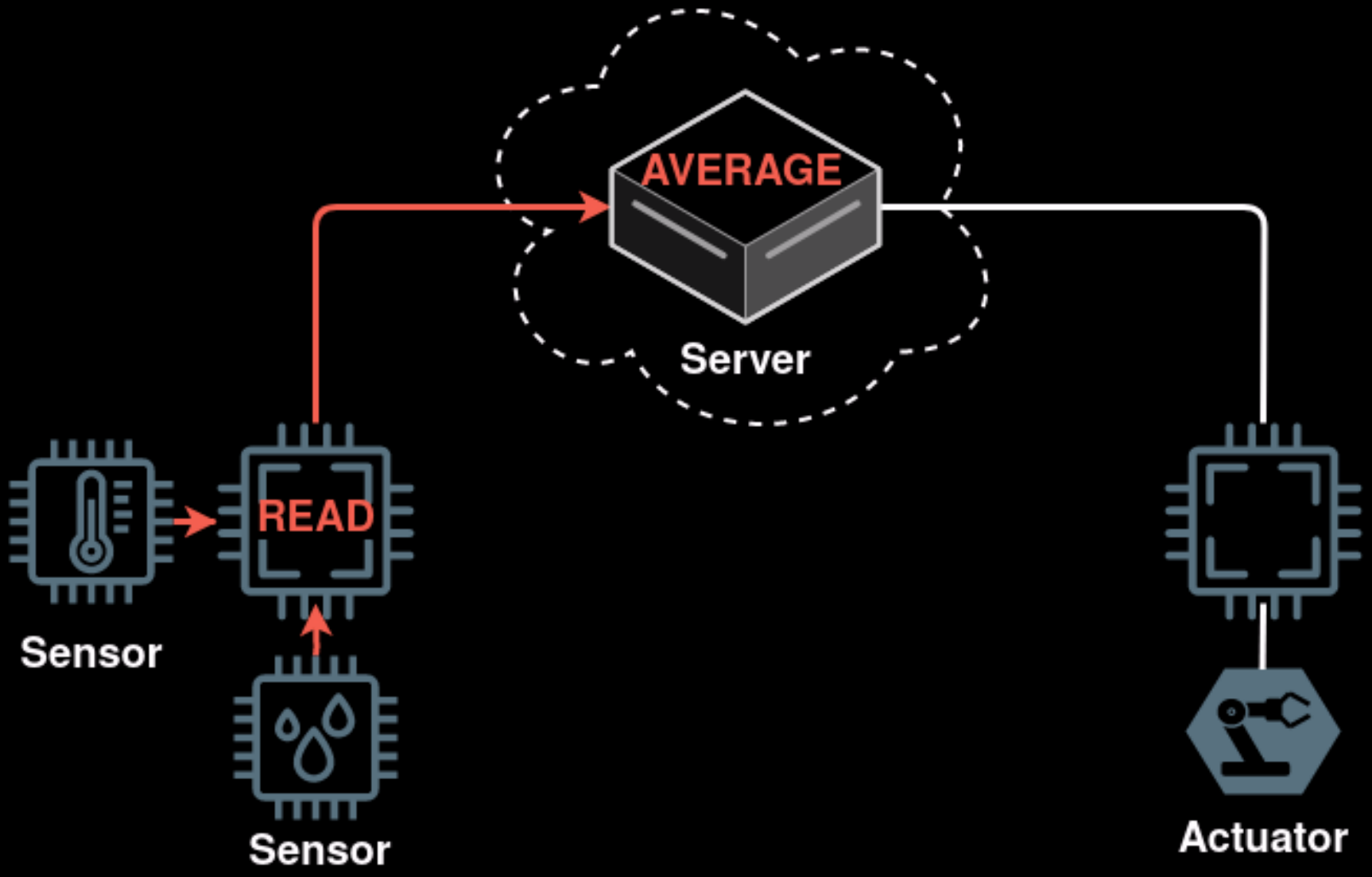
EASY!

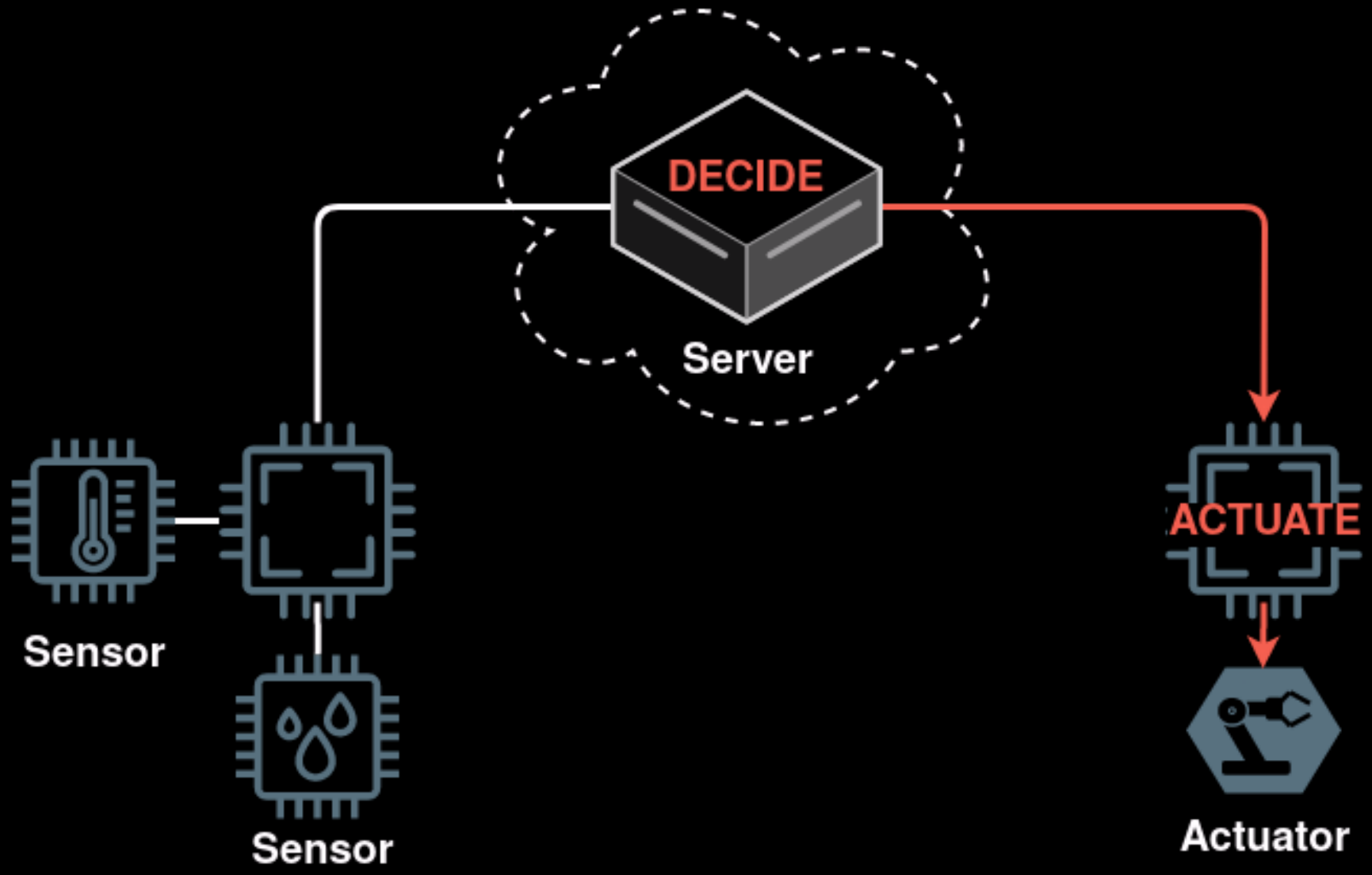
~~RESILIENT~~

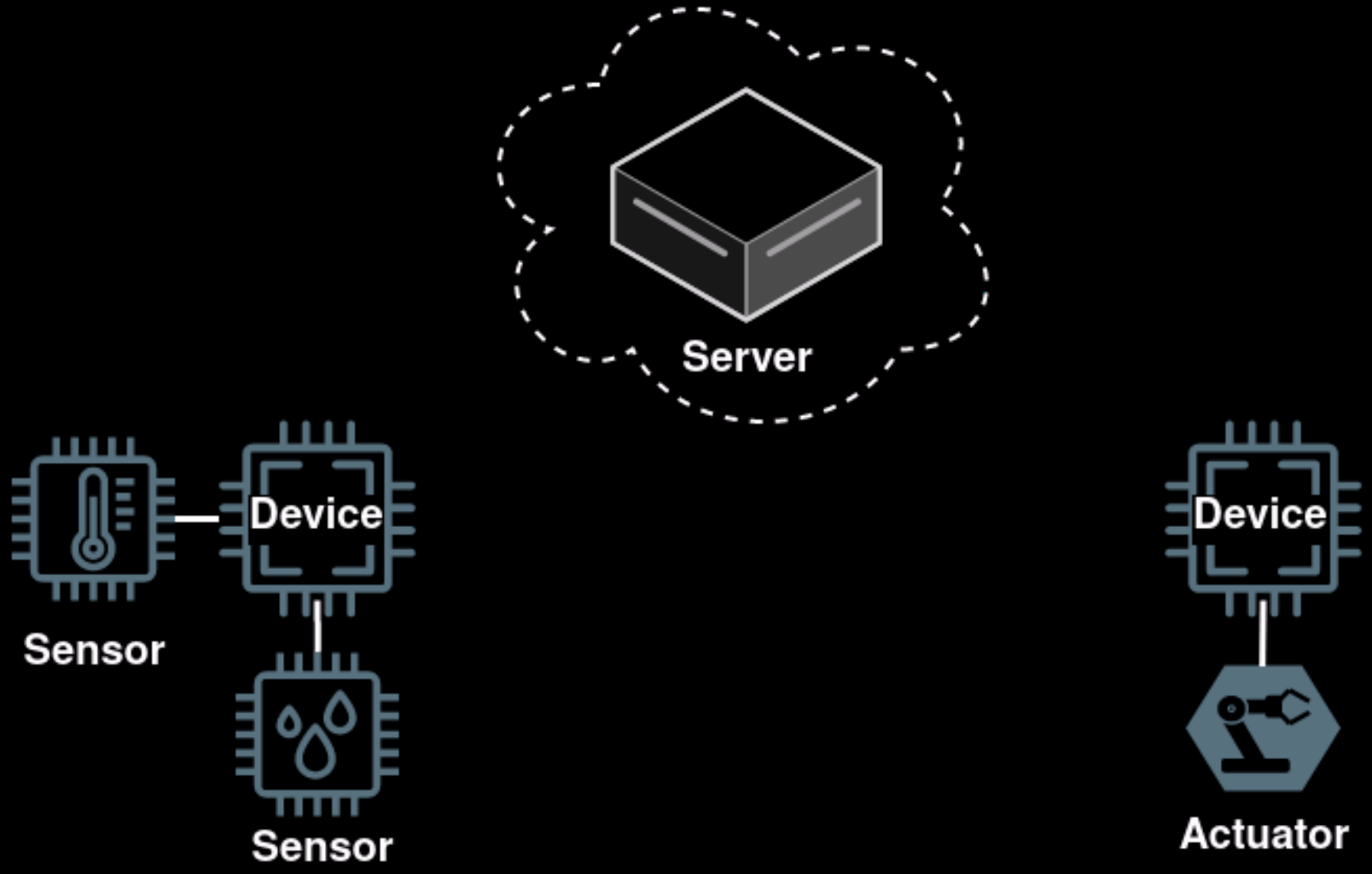
~~SECURE~~

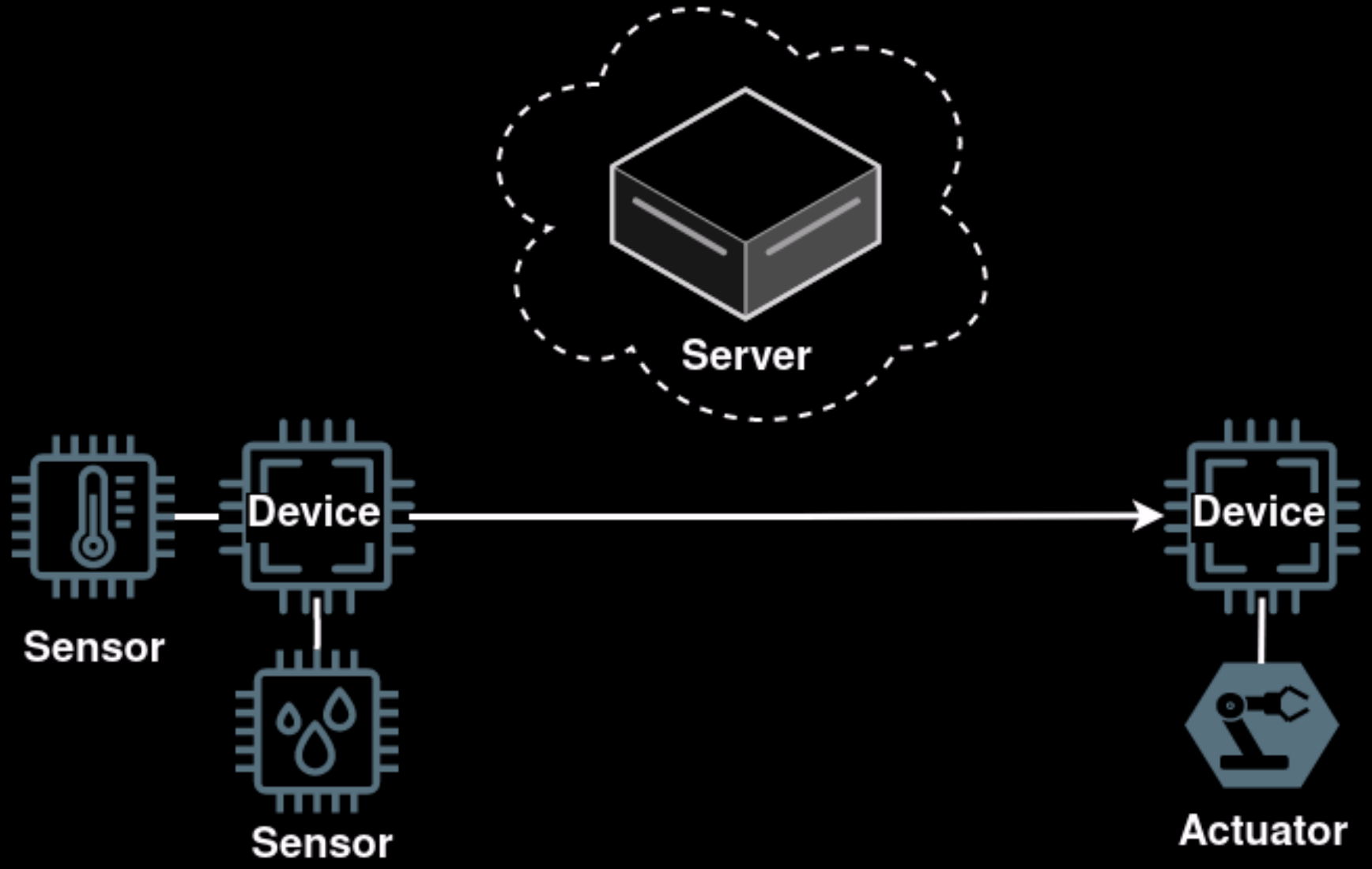
~~EVOLVABLE~~

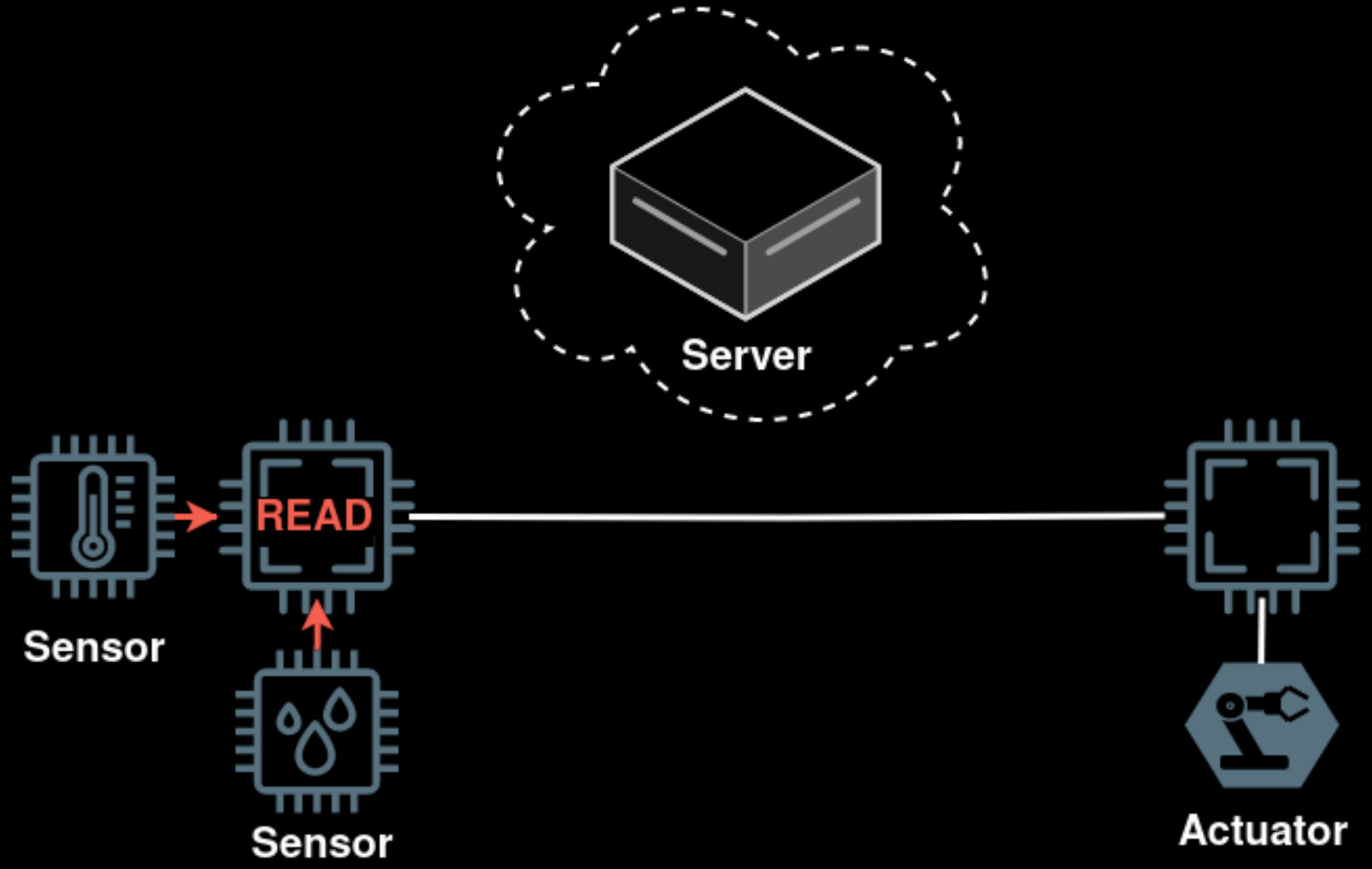
WHY?

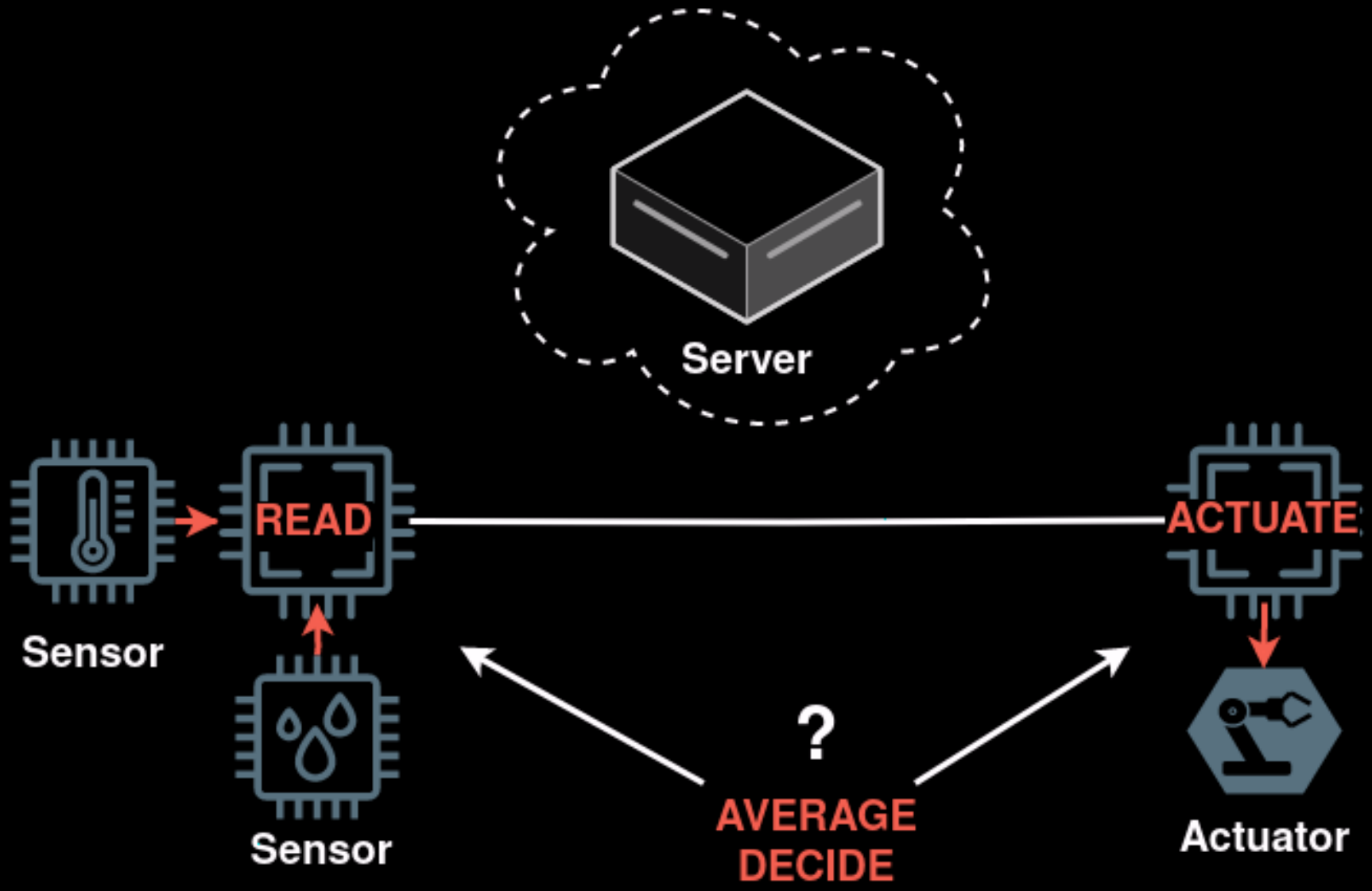

AVERAGE
Server
READ
Sensor
Sensor
Actuator

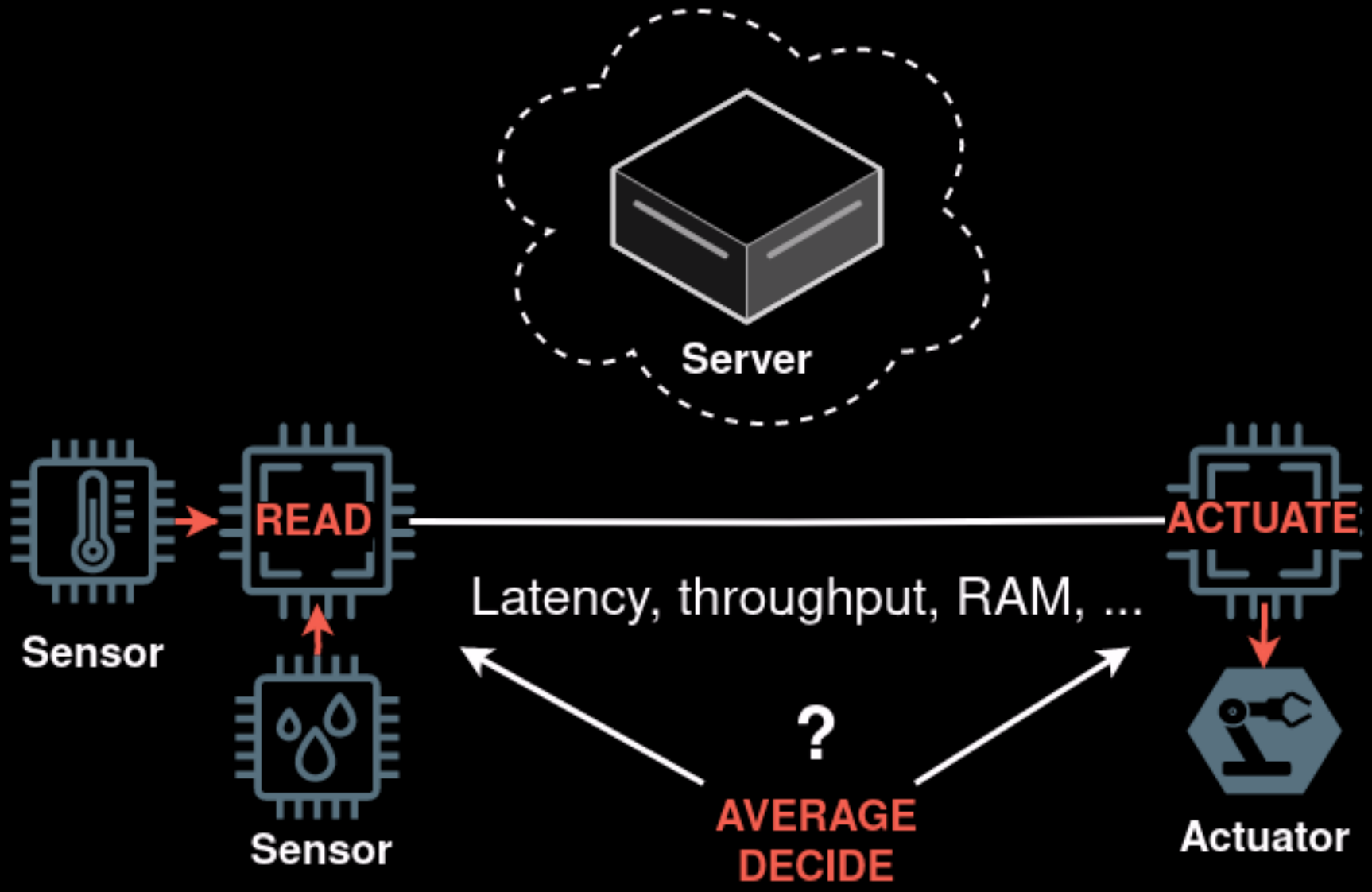

DECIDE
Server
Sensor
Sensor
ACTUATE
Actuator

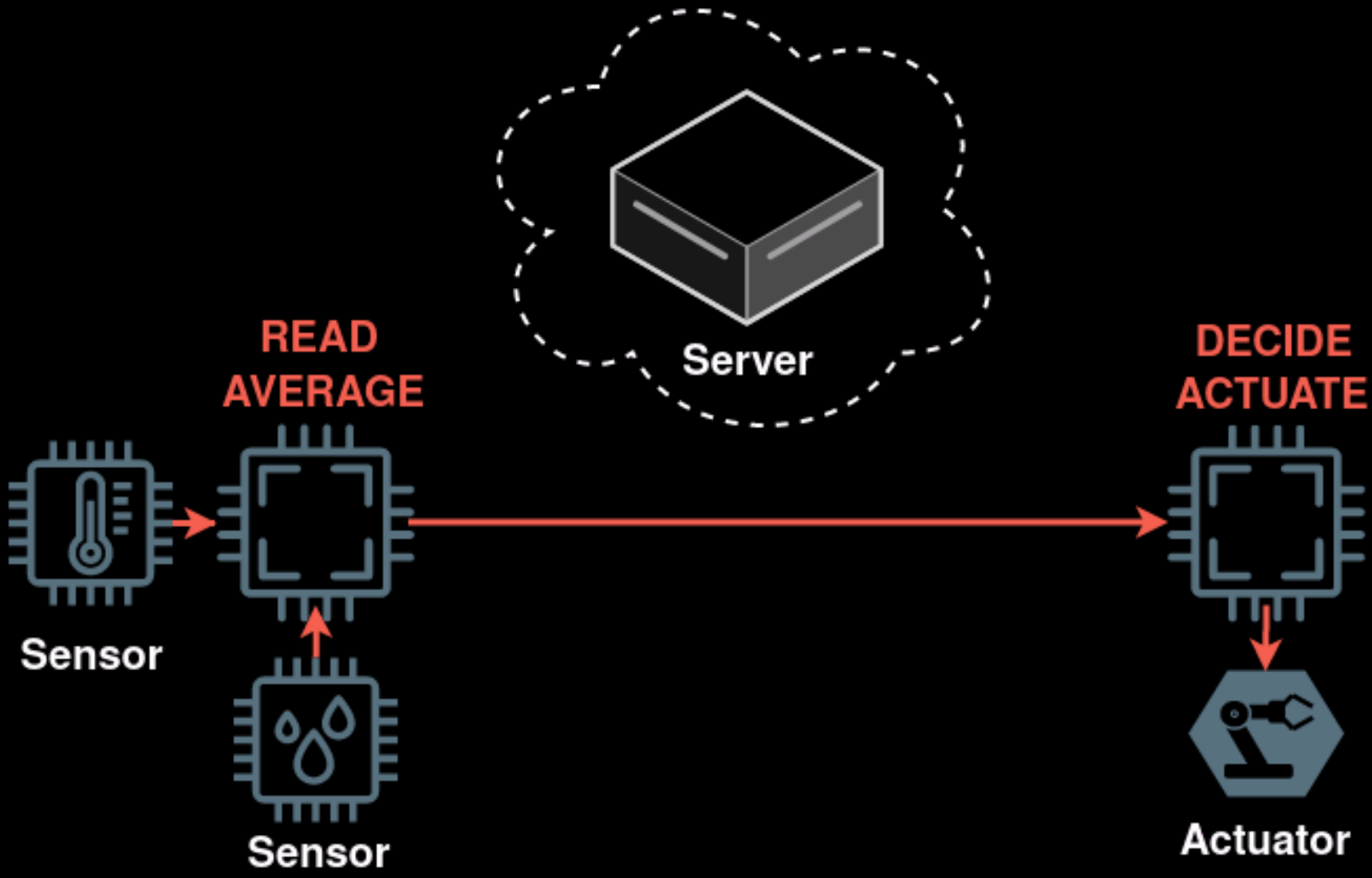

Server
Device
Sensor
Sensor
Device
Actuator

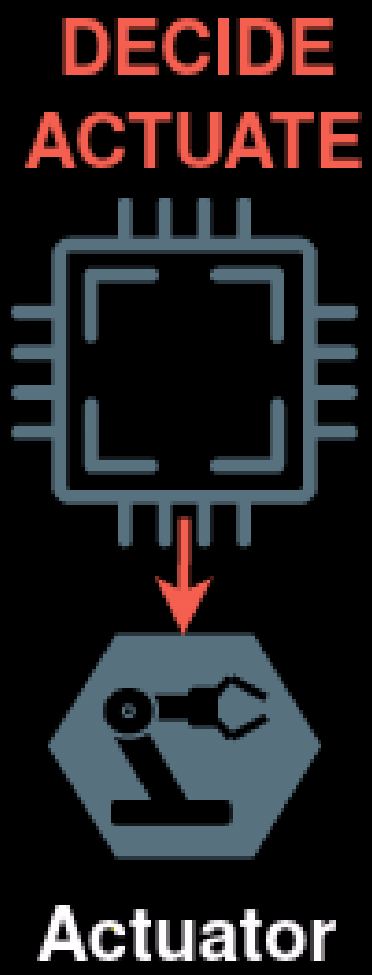

Server
Device
Sensor
Sensor
Device
Actuator


Server
READ
Sensor
Sensor
Actuator


Server
READ
ACTUATE
Sensor
Sensor
?
AVERAGE
DECIDE
Actuator


Server
READ
ACTUATE
Latency, throughput, RAM, ...
Sensor
Sensor
?
AVERAGE
DECIDE
Actuator

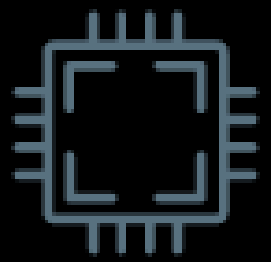

Server
READ
AVERAGE
DECIDE
ACTUATE
Sensor
Sensor
Actuator

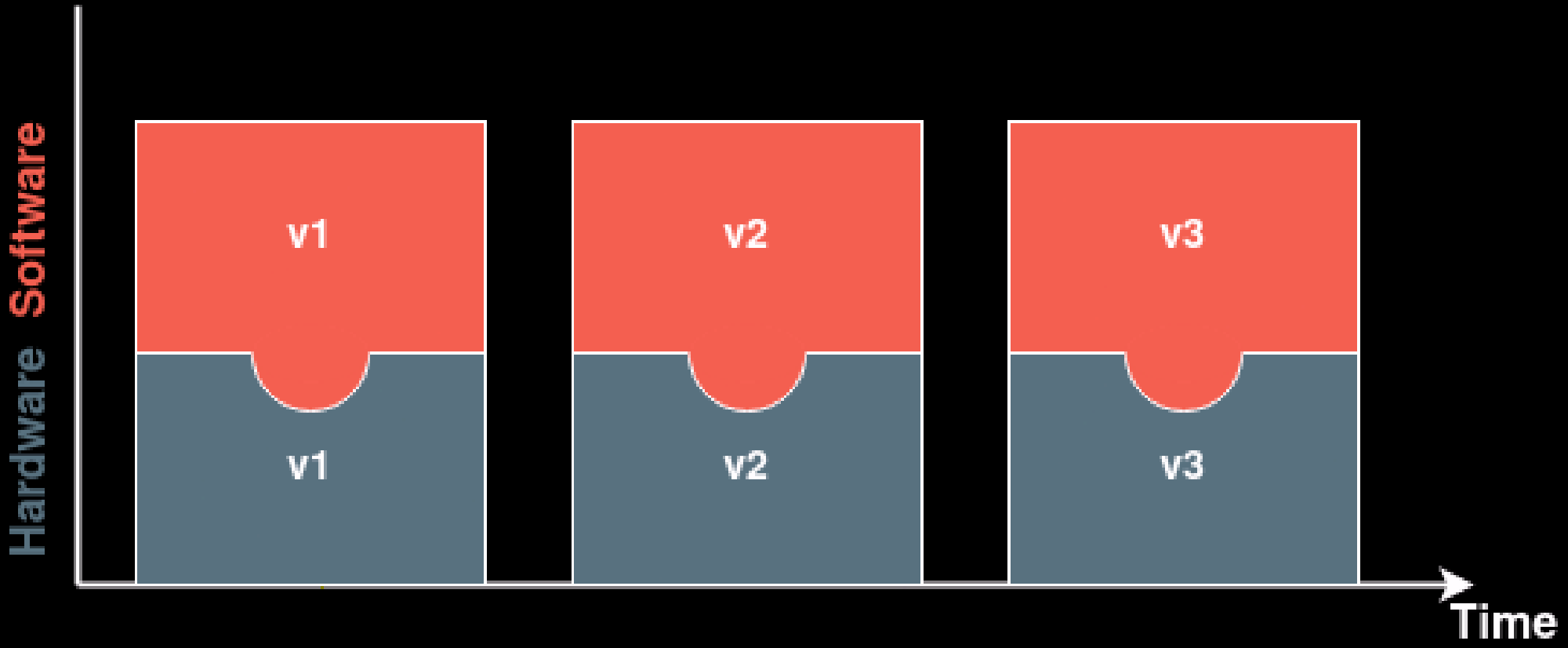


Hardware 🔒 Software

Hardware ☠ Software

DECIDE

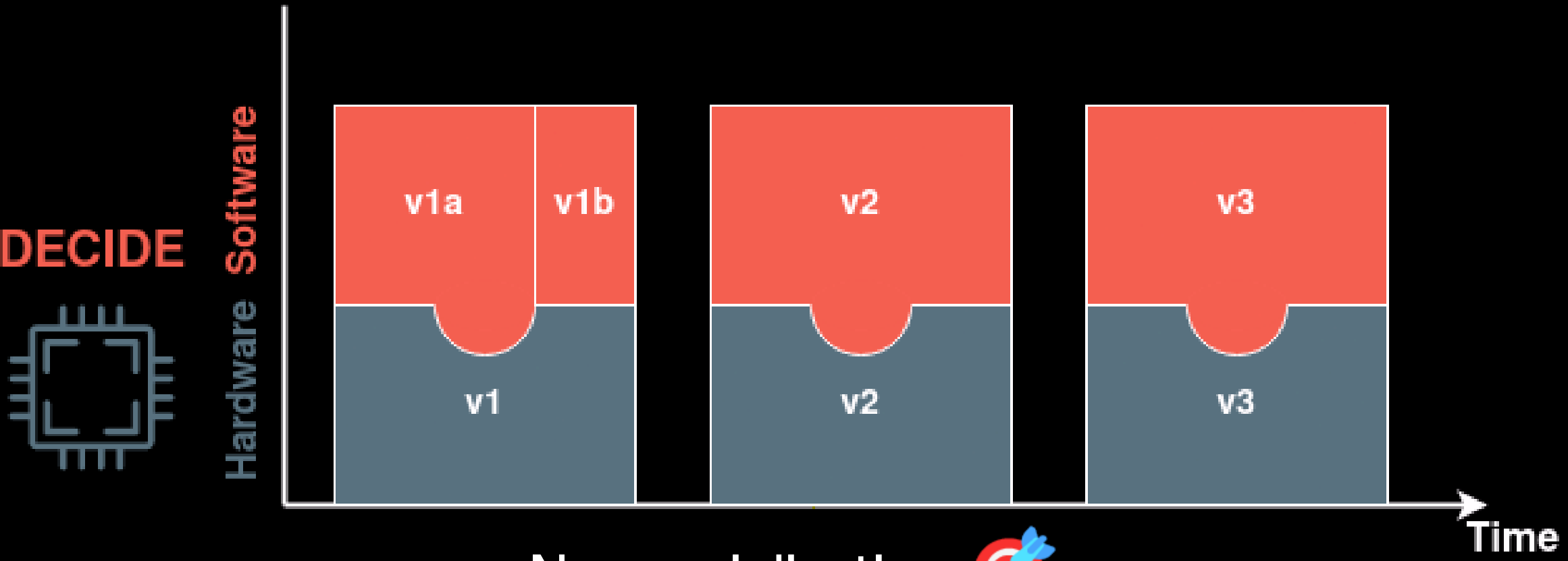




Hardware 🔒 Software

Hardware ☠ Software

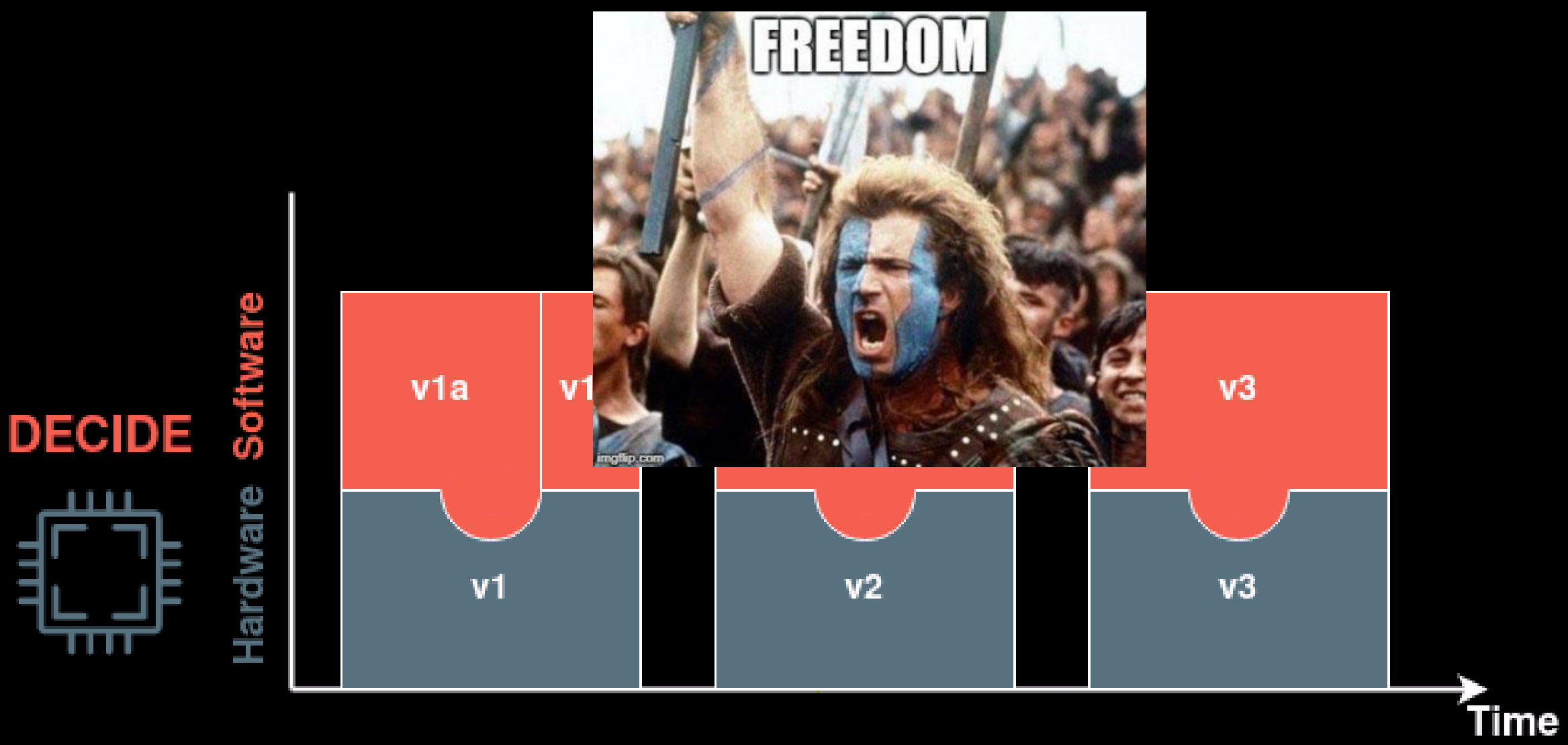
DECIDE



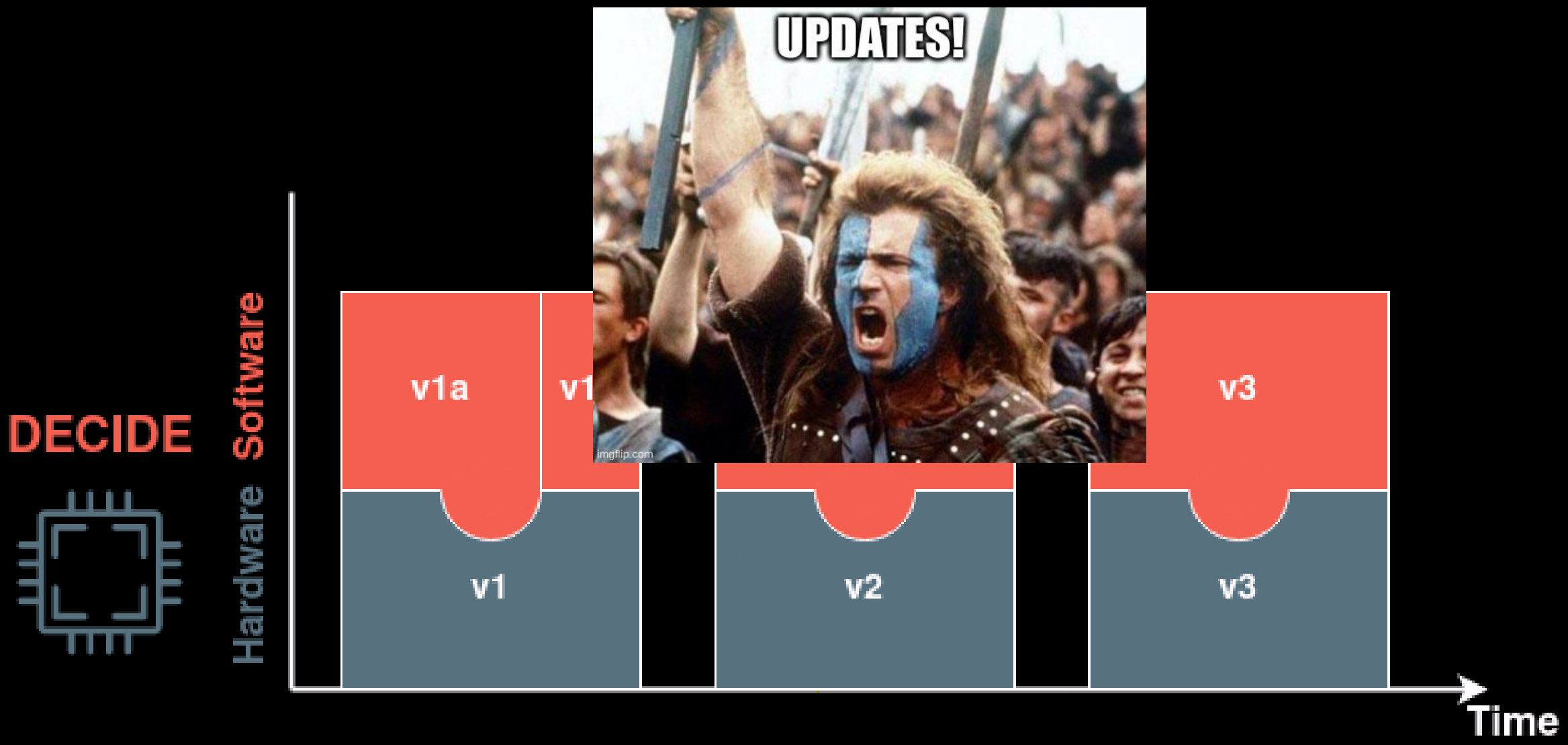
No specialization 🎯

Hardware 🔒 Software

Hardware ☠️ Software

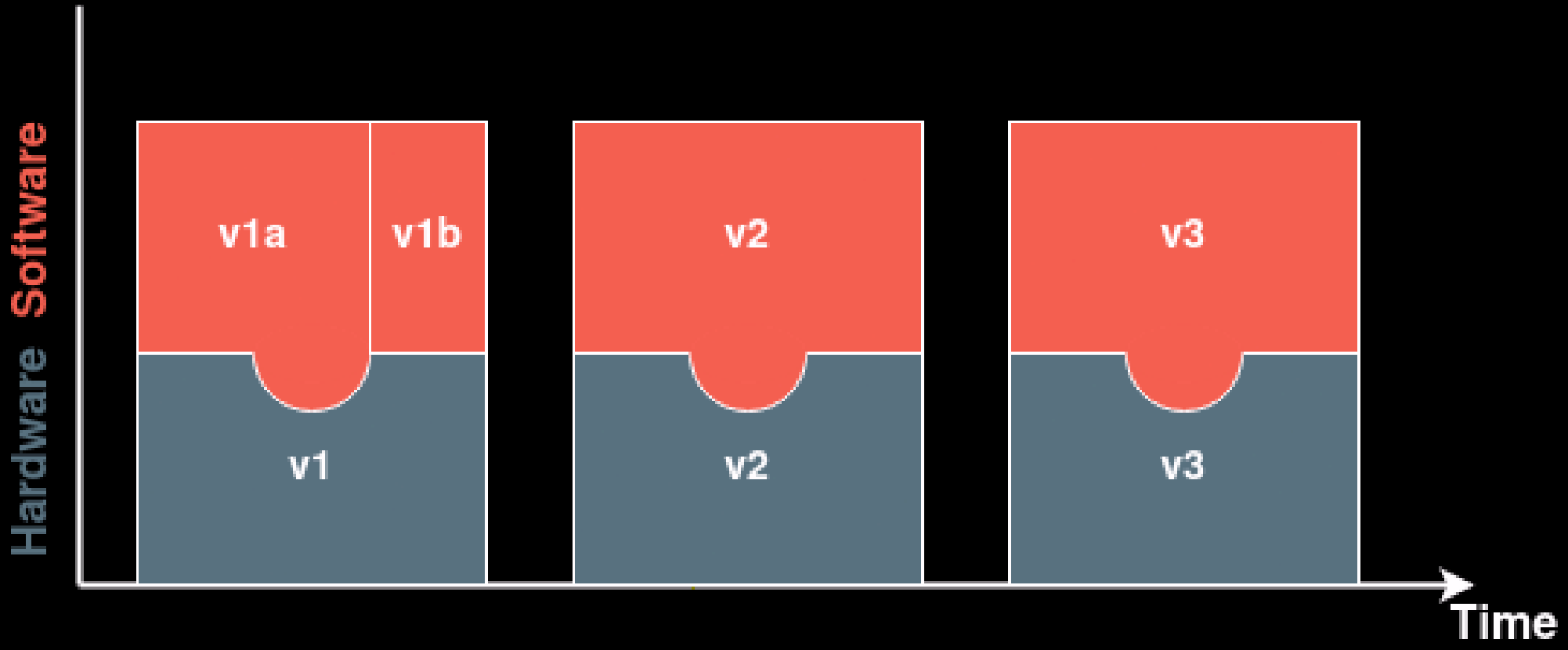


Software must become independent



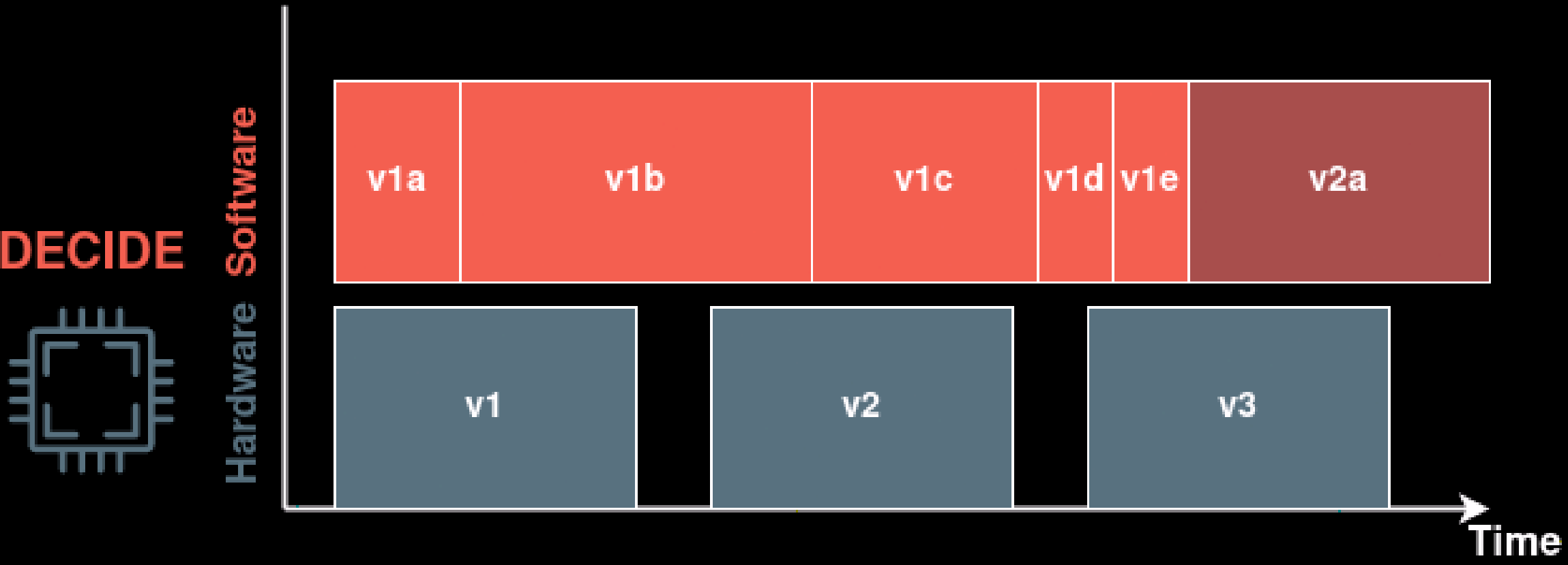
Software must become independent … and evolve

DECIDE

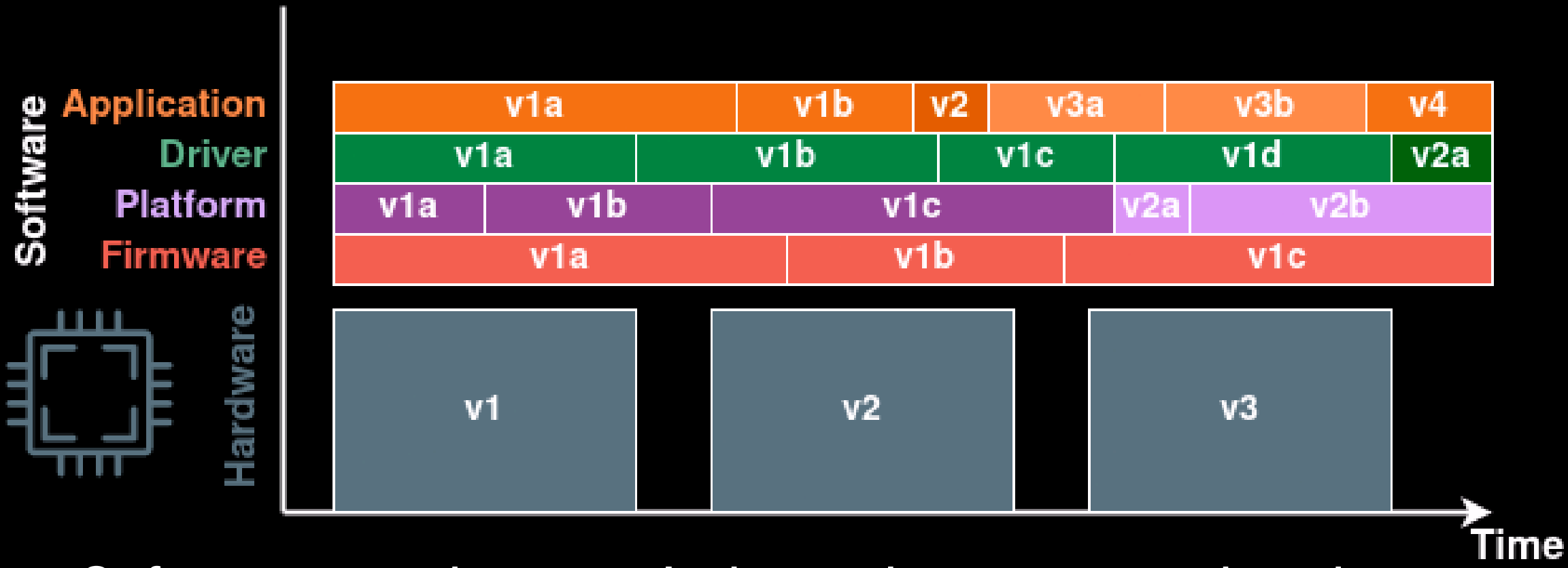


Software must become independent … and evolve

DECIDE



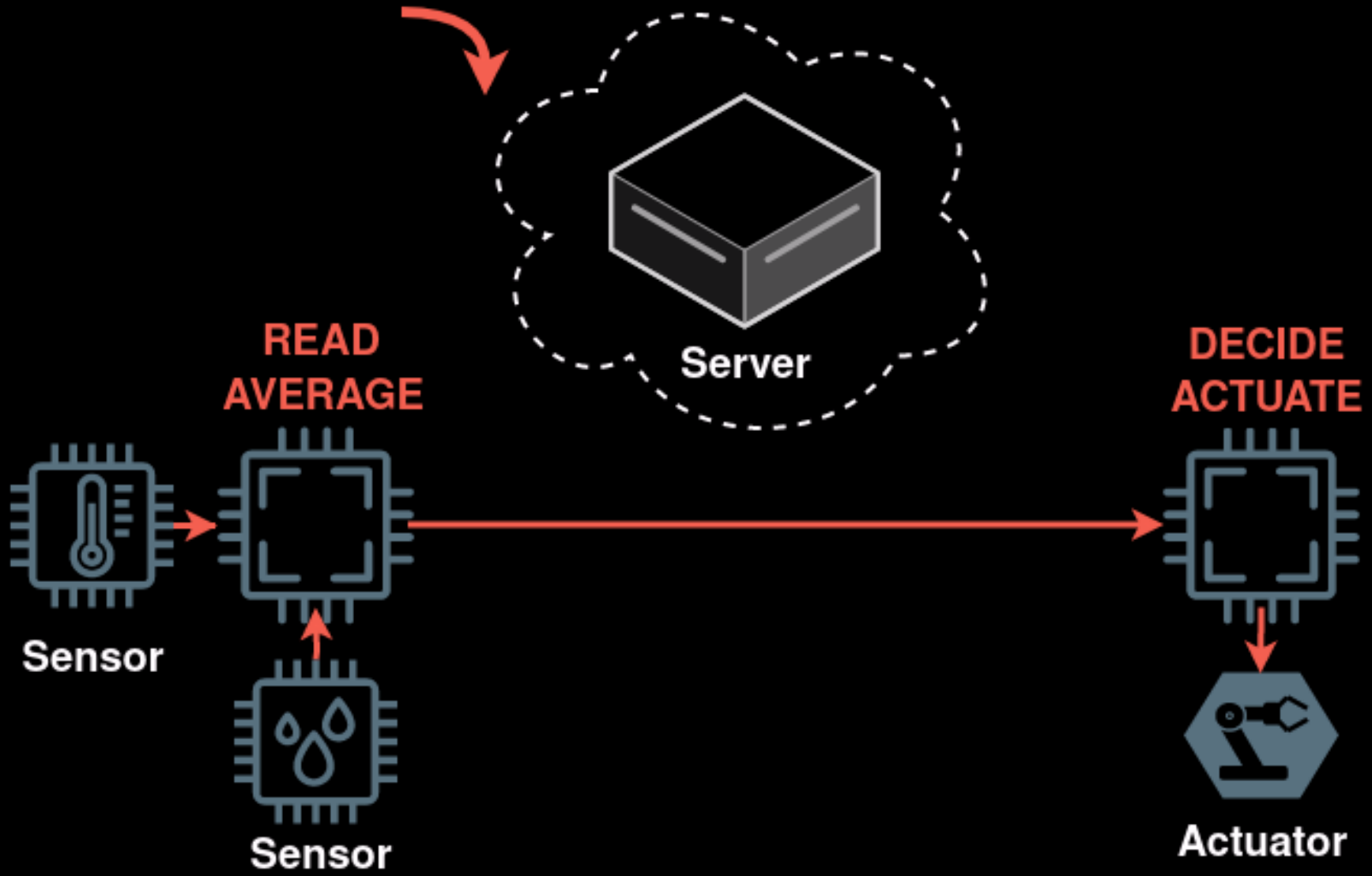
Software must become independent … and evolve

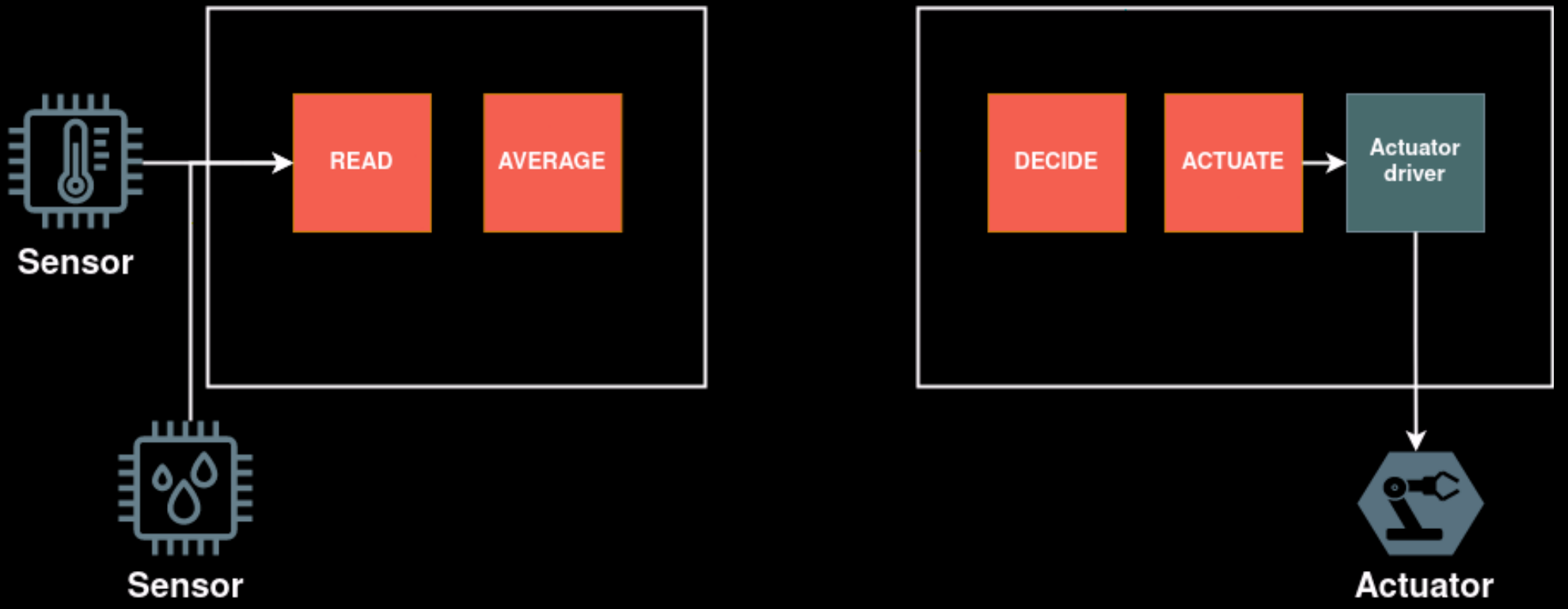


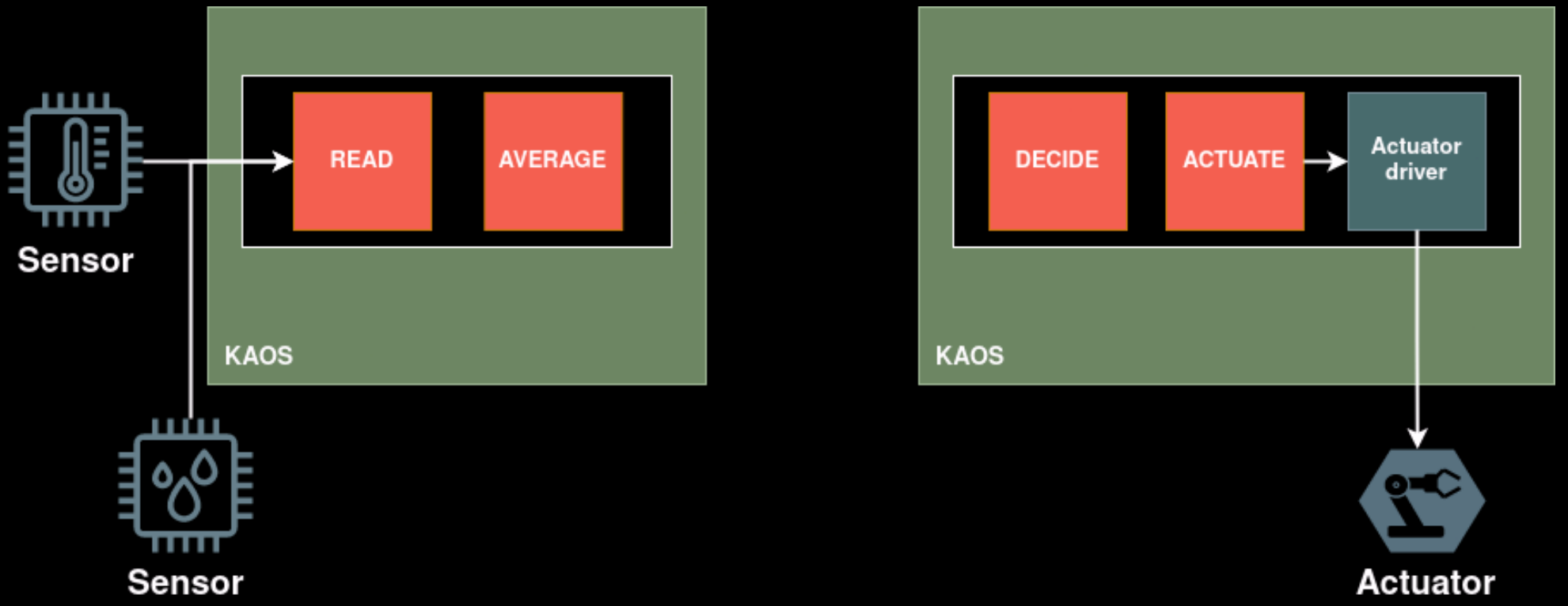
Software must become independent … and evolve

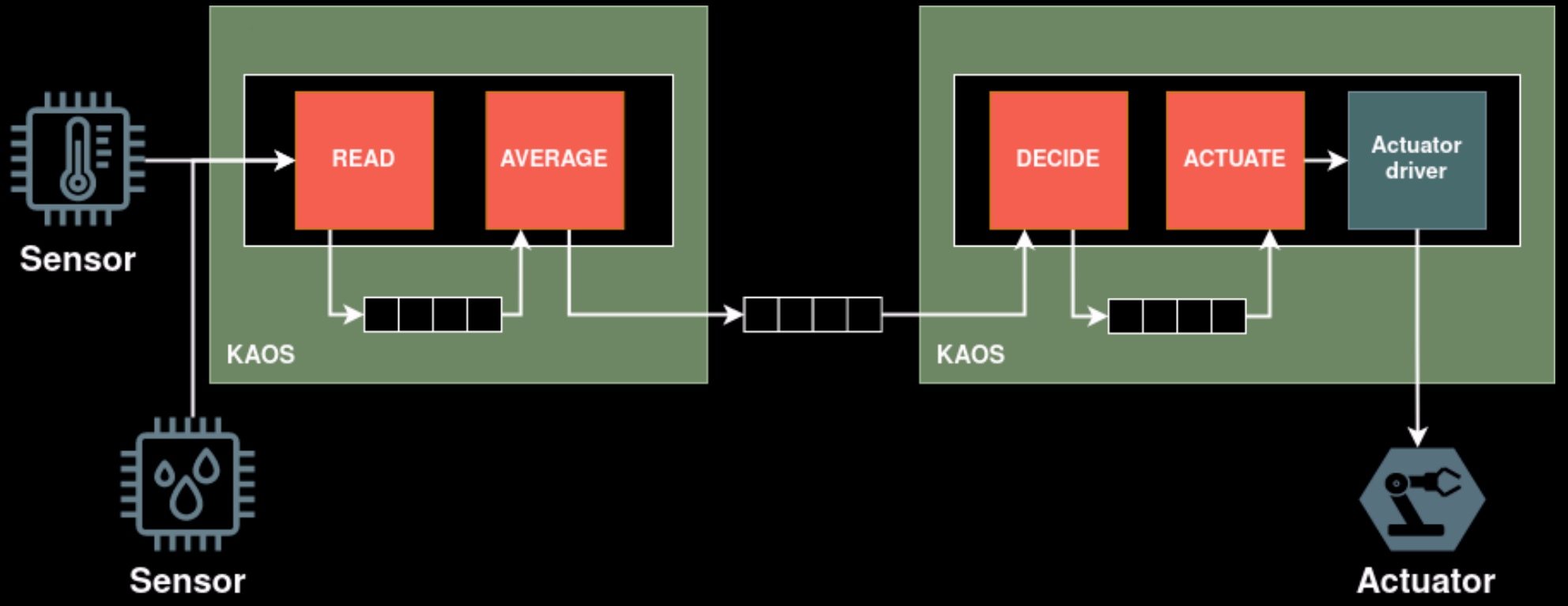
ECOSYSTEM that supports evolvability

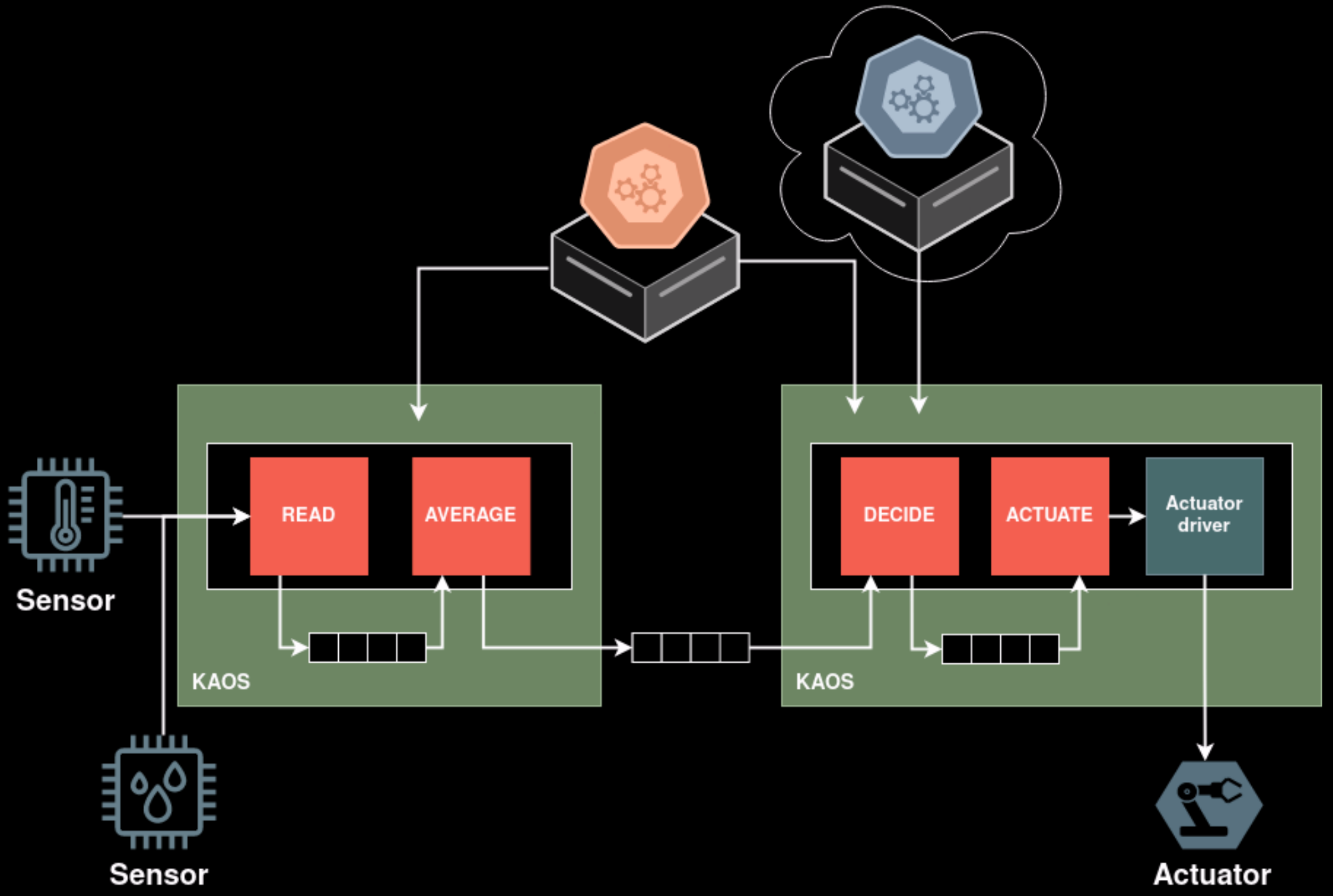
RUN A COMPLEX CONTROL POLICY FOR DECADES




READ
AVERAGE
Sensor
Sensor
DECIDE
ACTUATE
Actuator driver
Actuator


READ
AVERAGE
Sensor
KAOS
Sensor
DECIDE
ACTUATE
Actuator driver
KAOS
Actuator


Sensor
READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Actuator


READ
AVERAGE
DECIDE
ACTUATE
Actuator driver
KAOS
KAOS
Sensor
Sensor
Actuator

ESP32

Small but mighty

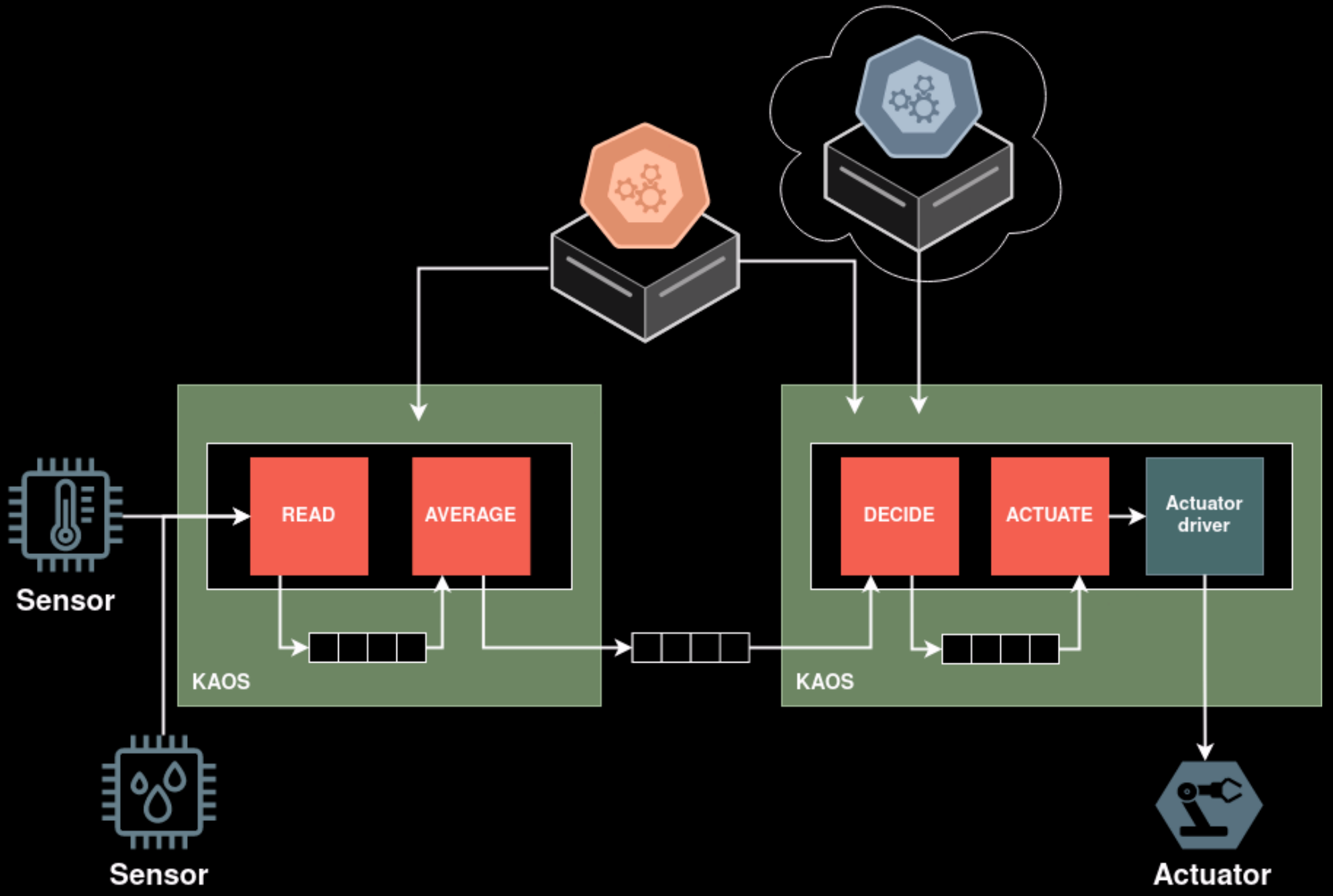
Cheap devices (~30p) with plenty of resources

Multi-core, co-processors, megabytes* of RAM

WiFi + Bluetooth

Very popular platform for smart devices

We can do stuff!


READ
AVERAGE
DECIDE
ACTUATE
Actuator driver
KAOS
KAOS
Sensor
Sensor
Actuator

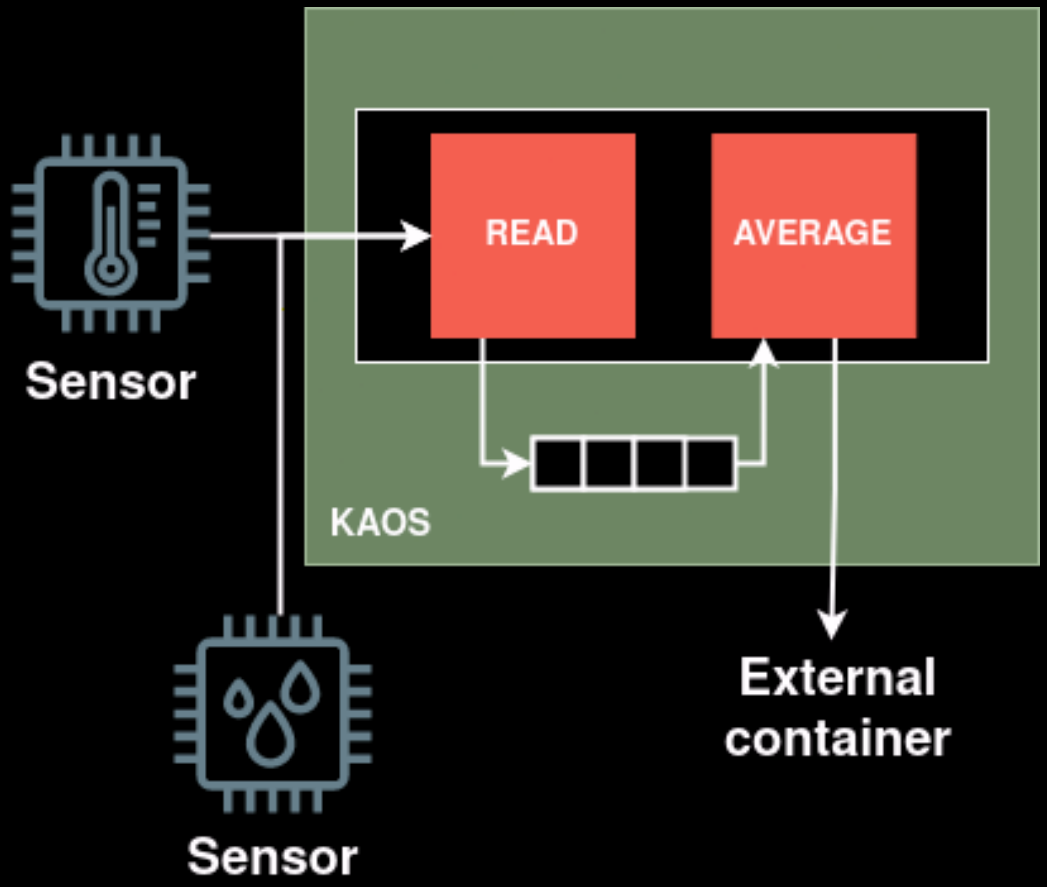
PORTABLE COMPONENTS

Decoupling hardware and software

Specialize software development

Accommodate multiple stakeholders

Run on various hardware platforms



PORTABLE COMPONENTS

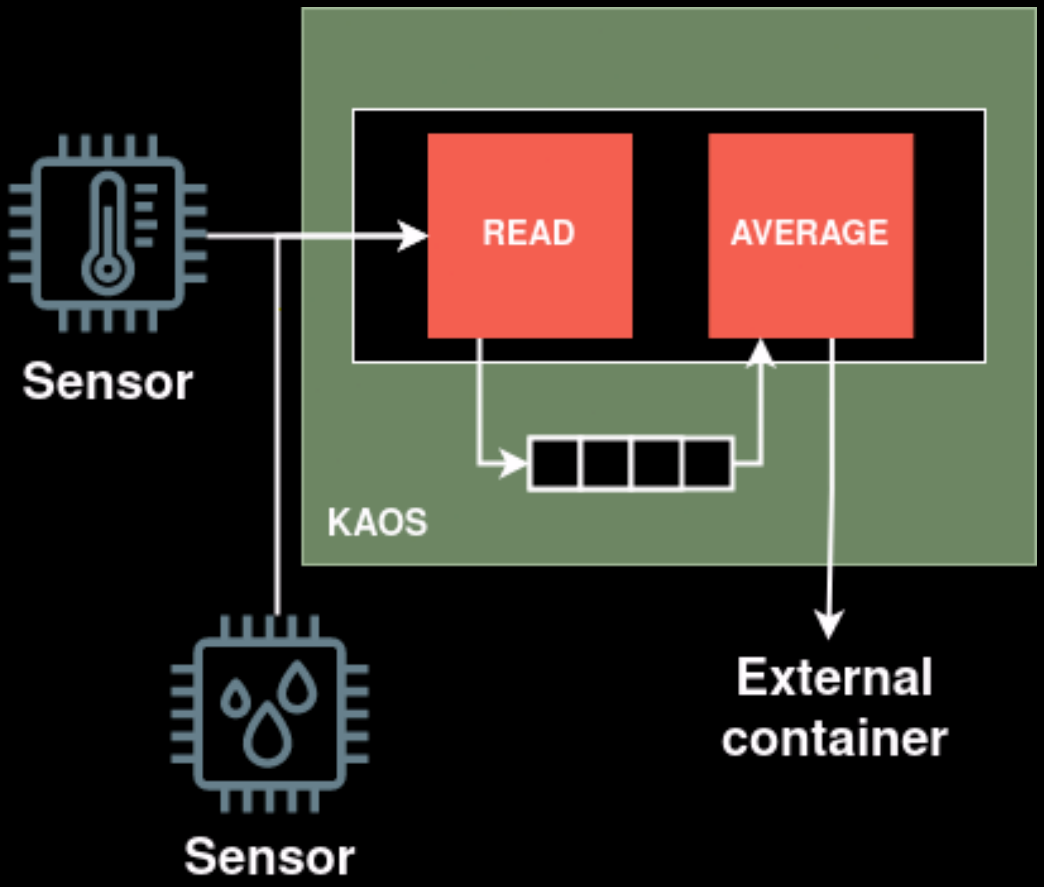
Decoupling hardware and software

Specialize software development

Accommodate multiple stakeholders

Run on various hardware platforms

$$$



PORTABLE COMPONENTS

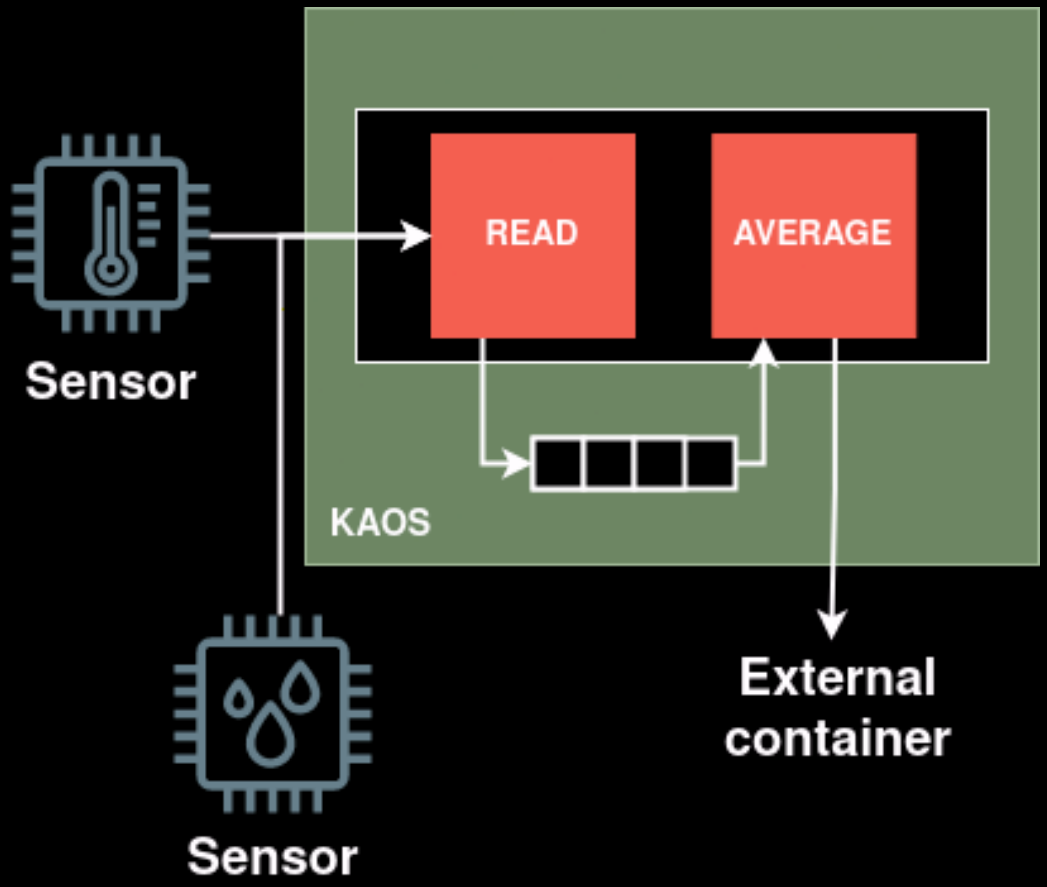
Decoupling hardware and software

Specialize software development

Accommodate multiple stakeholders

Run on various hardware platforms

→ **EVOLVABILITY**



PORTABLE COMPONENTS

Decoupling hardware and software

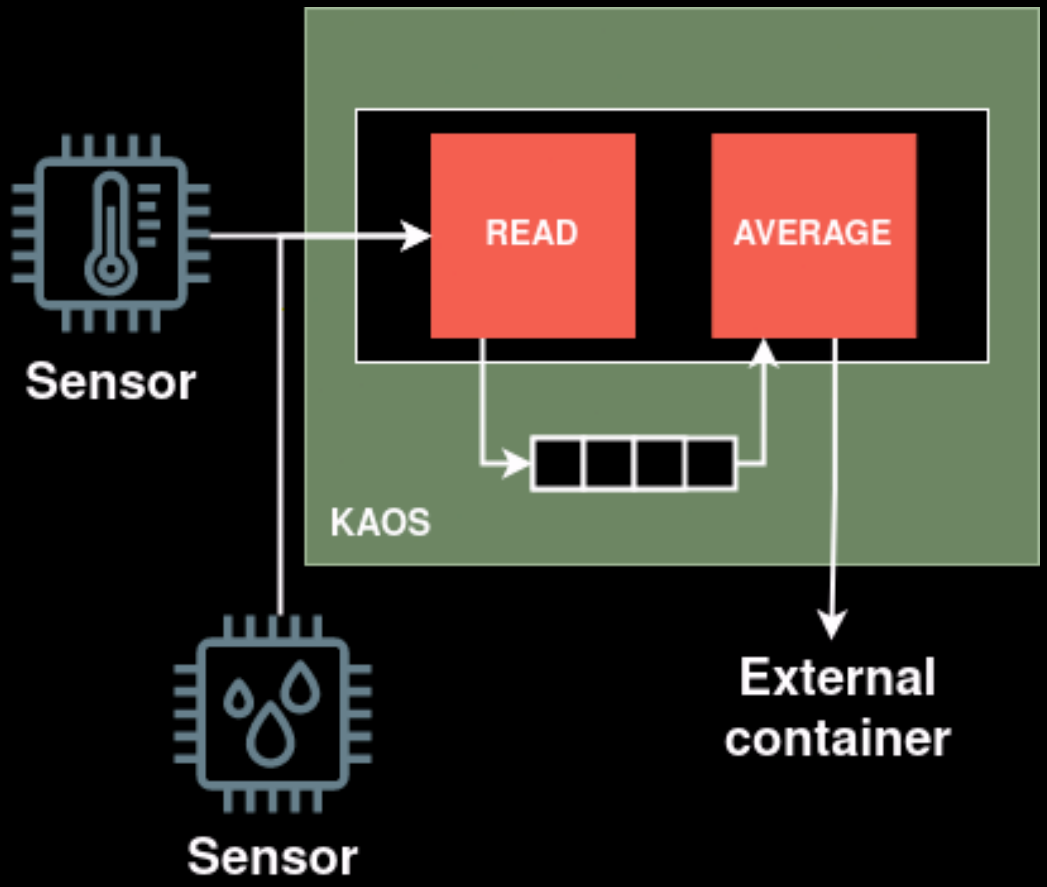
Specialize software development

Accommodate multiple stakeholders

Run on various hardware platforms

→ **EVOLVABILITY**

Restart computation and move it around



PORTABLE COMPONENTS

Decoupling hardware and software

Specialize software development

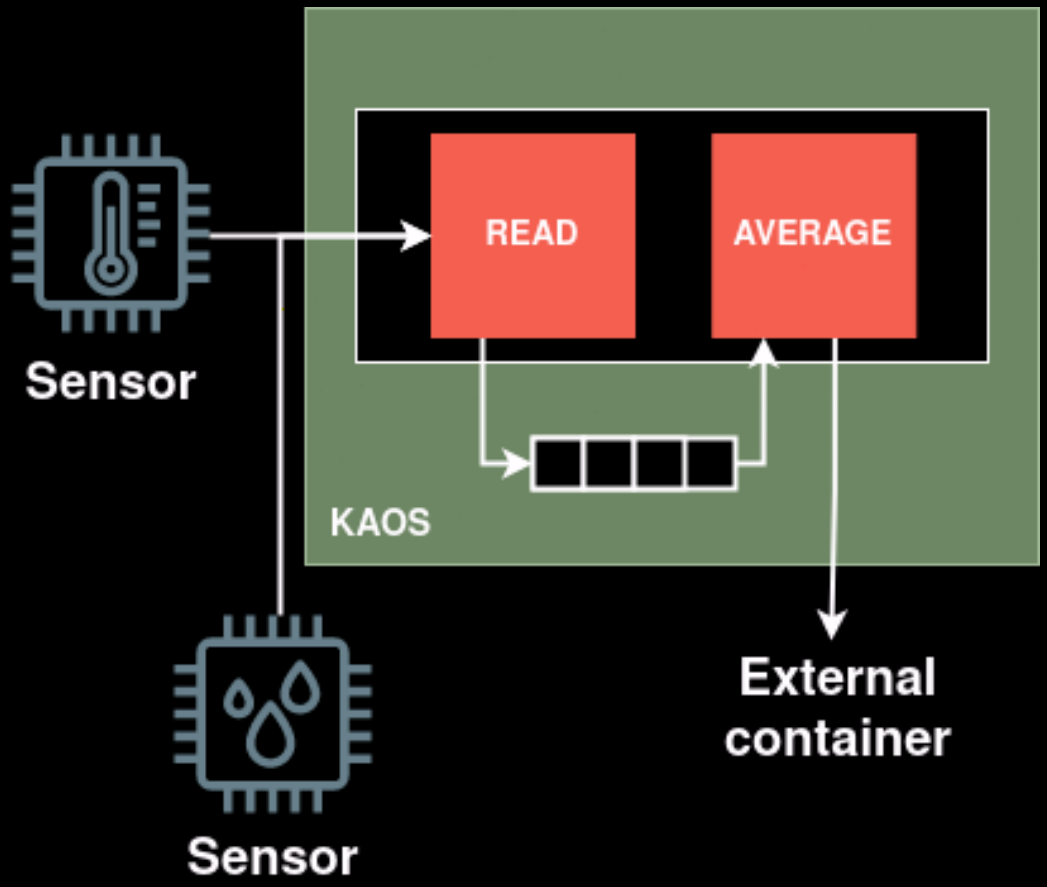
Accommodate multiple stakeholders

Run on various hardware platforms

→ **EVOLVABILITY**

Restart computation and move it around

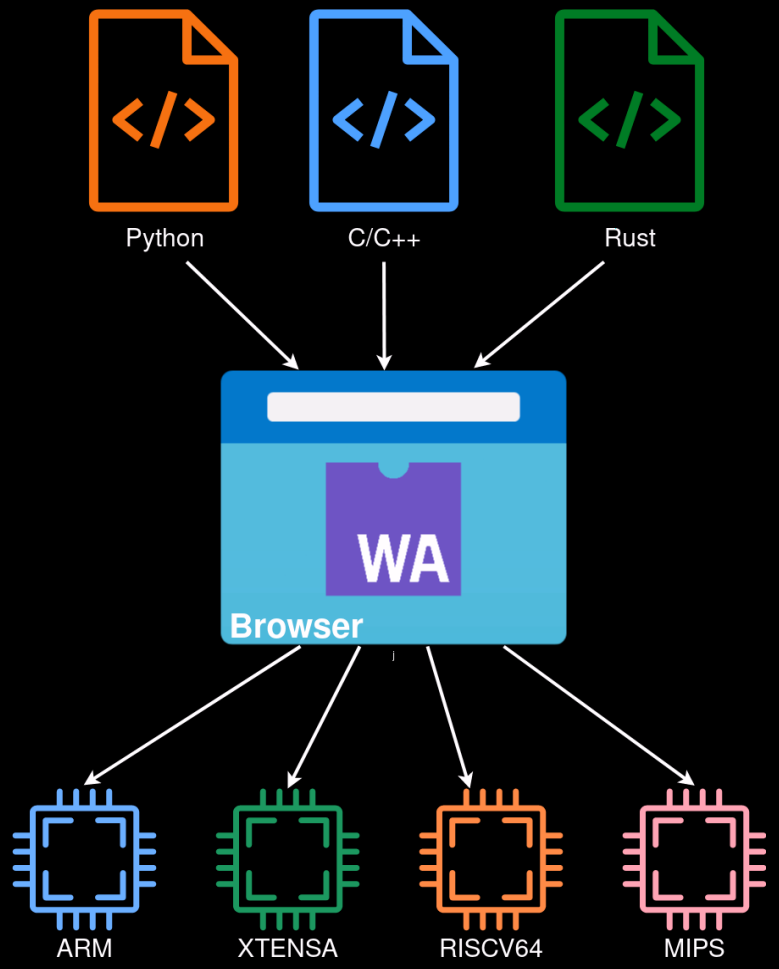
→ **RESILIENCE**



WEB ASSEMBLY ON ESP32?

Open portable binary standard

High performance computation in the browser

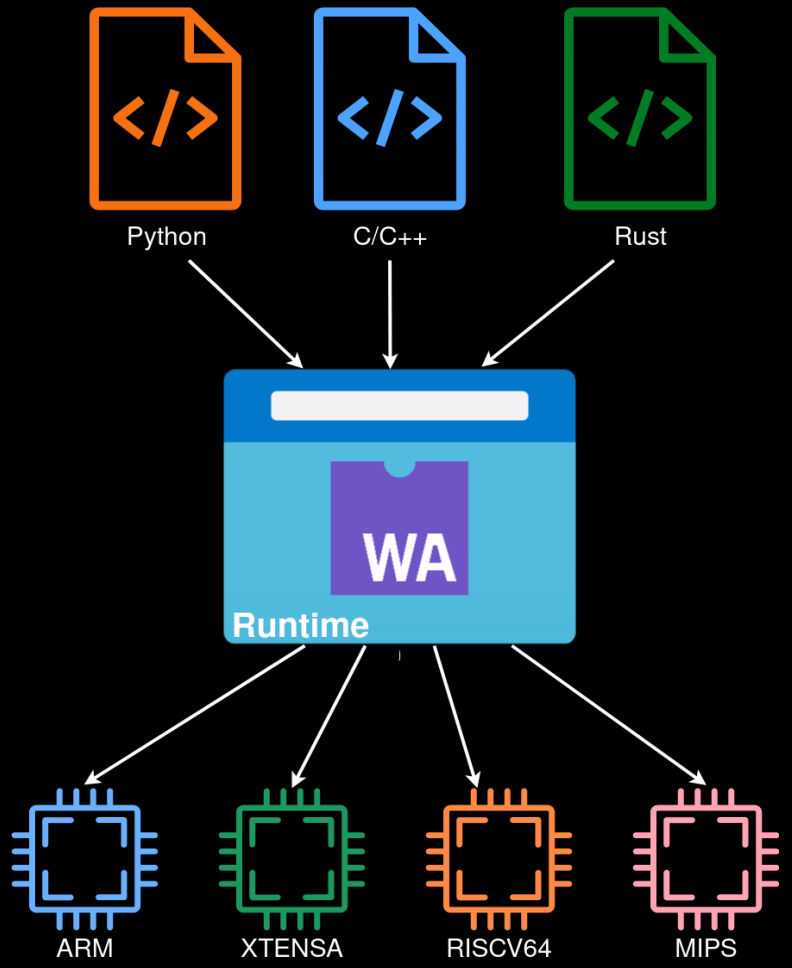


WEB ASSEMBLY ON ESP32?

Open portable binary standard

High performance computation ~~in the browser~~

Web Assembly Micro Runtime: small footprint



WEB ASSEMBLY ON ESP32?

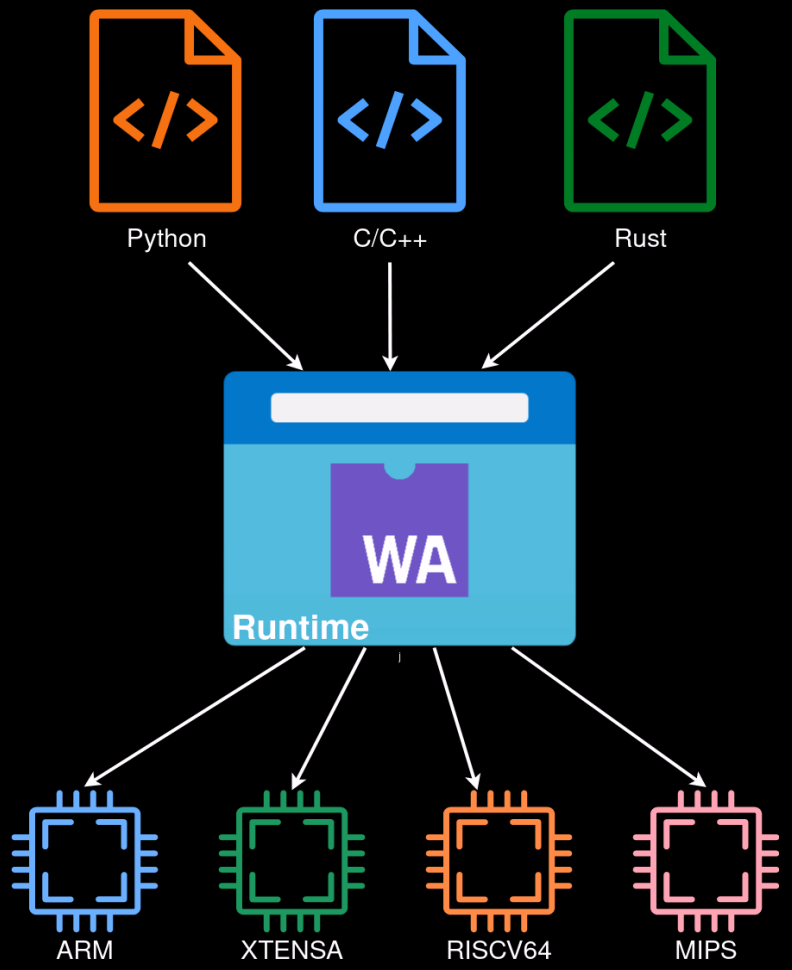
Open portable binary standard

High performance computation ~~in the browser~~

WAMR Micro Runtime: small footprint

Isolated sandbox → SECURITY

Independent portable containers



WEB ASSEMBLY ON ESP32?

Open portable binary standard

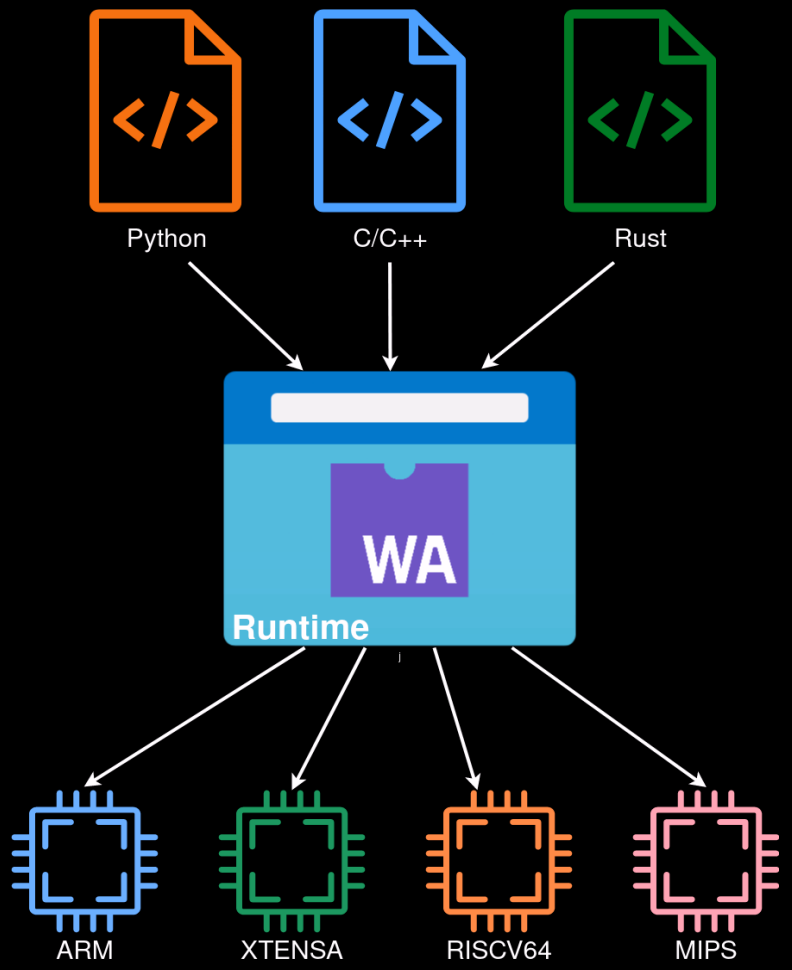
High performance computation ~~in the browser~~

WAMR Micro Runtime: small footprint

Isolated sandbox → SECURITY

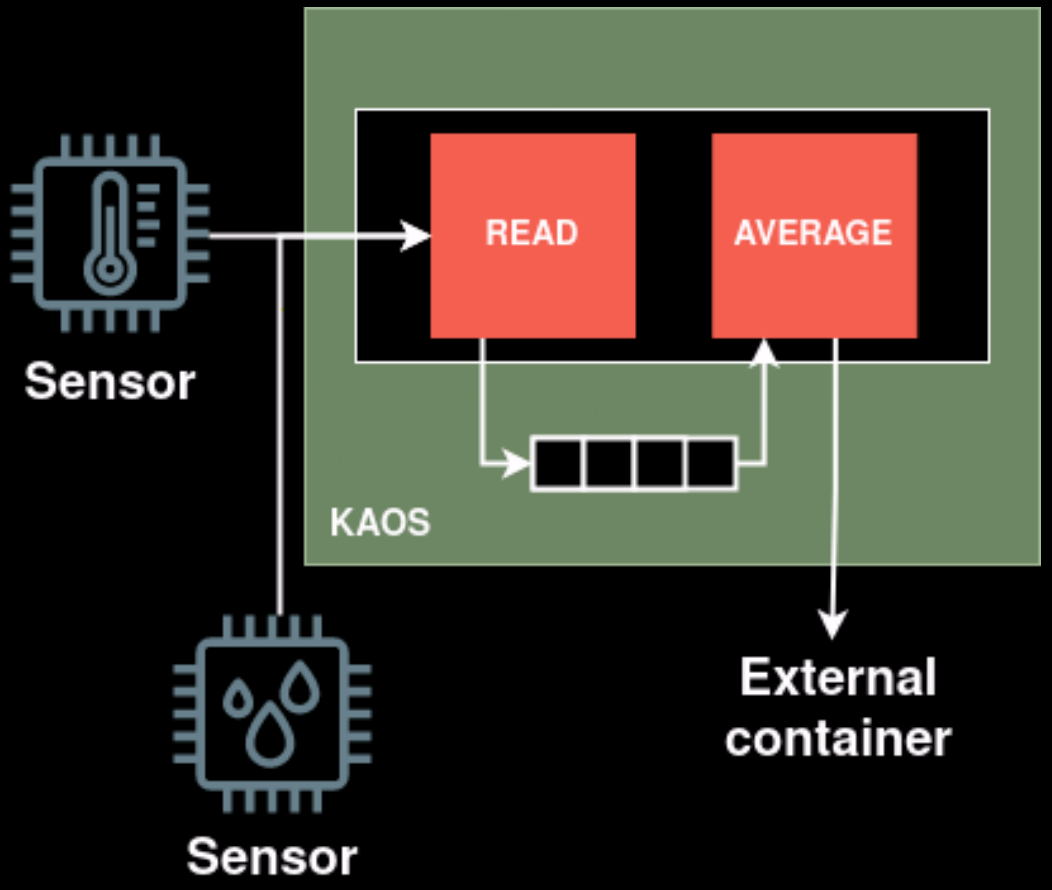
Independent portable containers

BUT limited interface – system calls



DEVICE PLATFORM

Containers need access to external resources

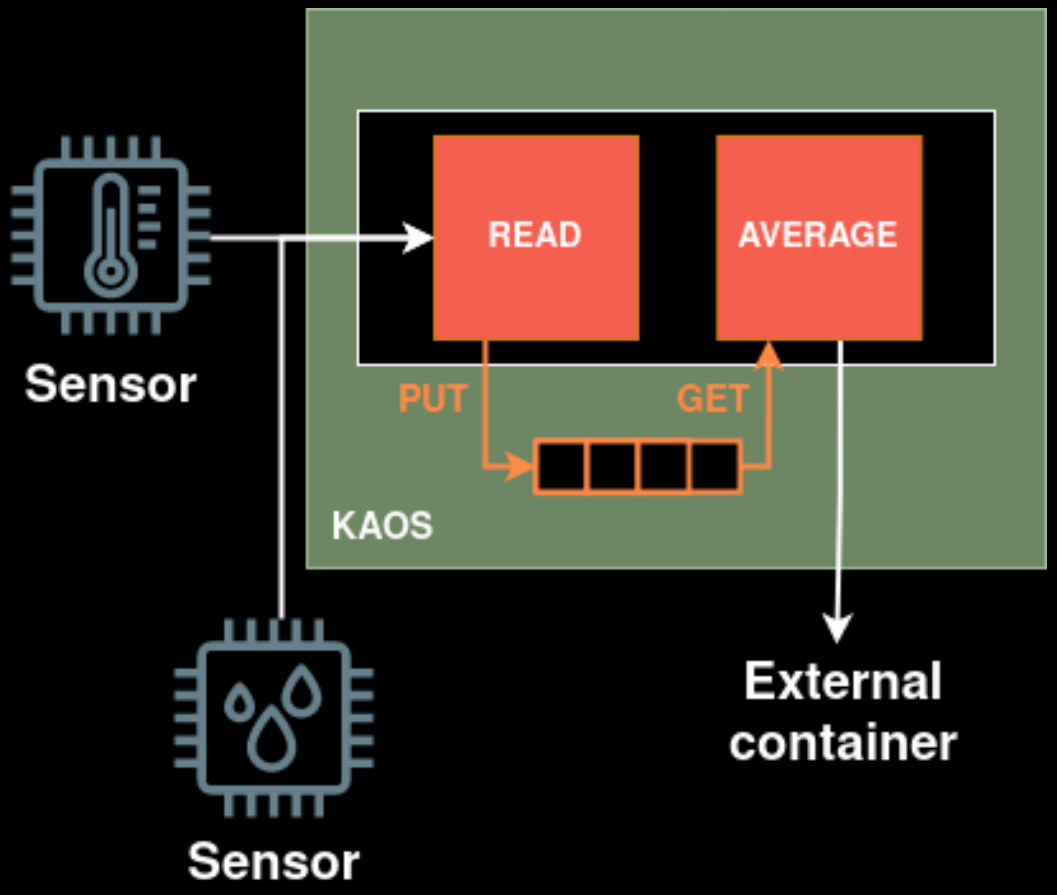


DEVICE PLATFORM

Containers need access to external resources

Message channels

Restricted API



DEVICE PLATFORM

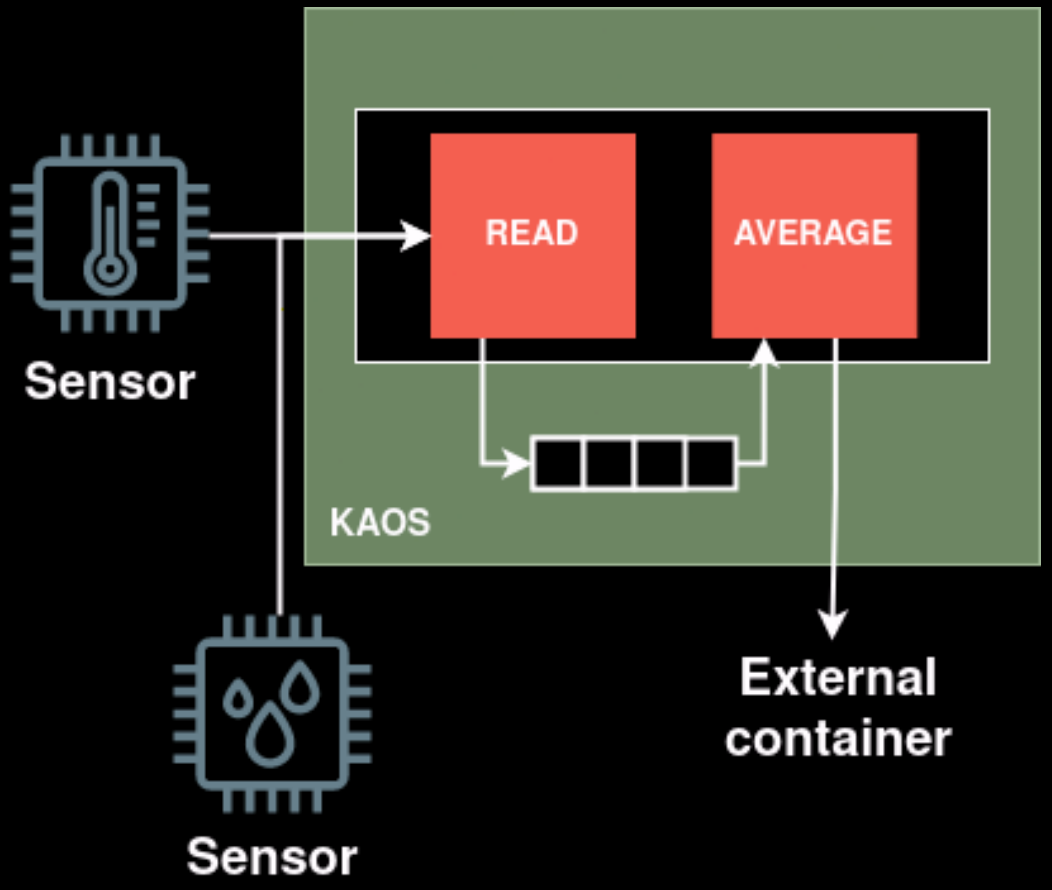
Containers need access to external resources

Message channels

Restricted API

Limit fate sharing - live when application dies

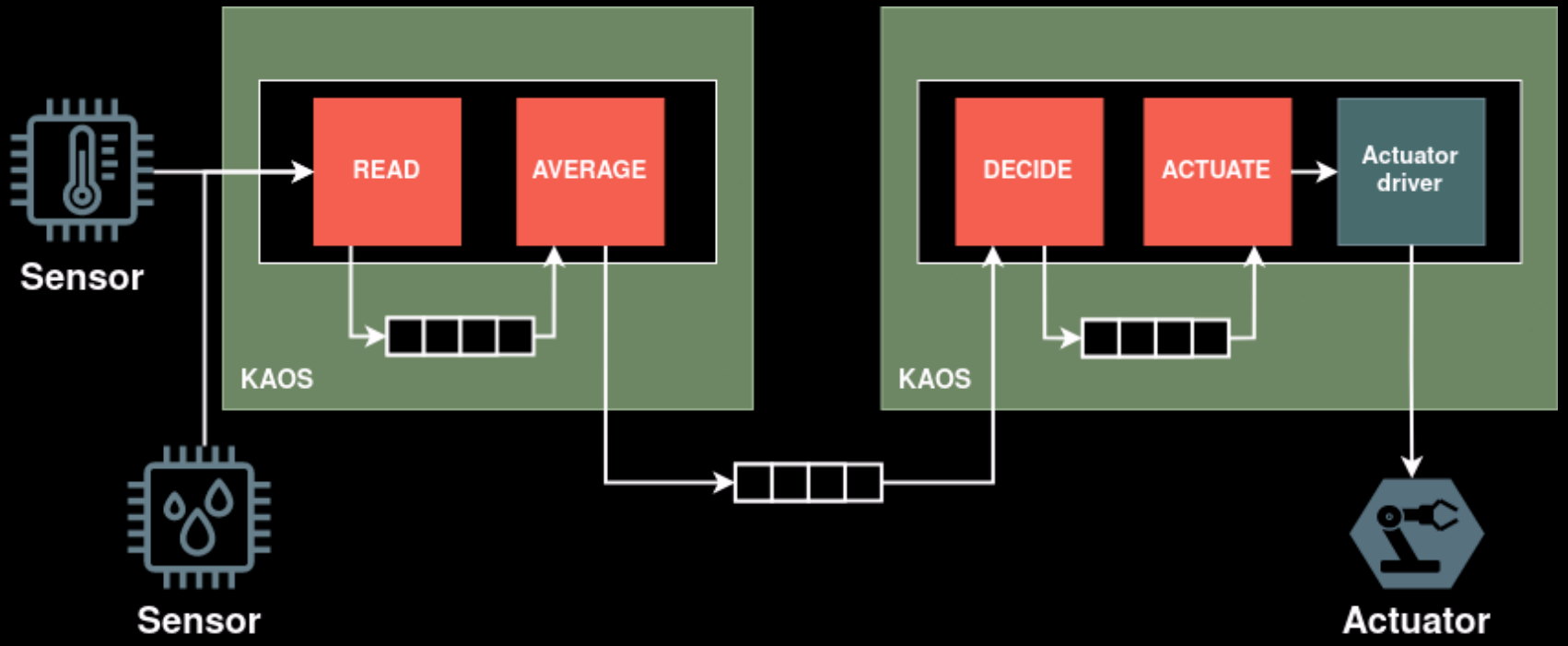
Limit disruption by events and restore service

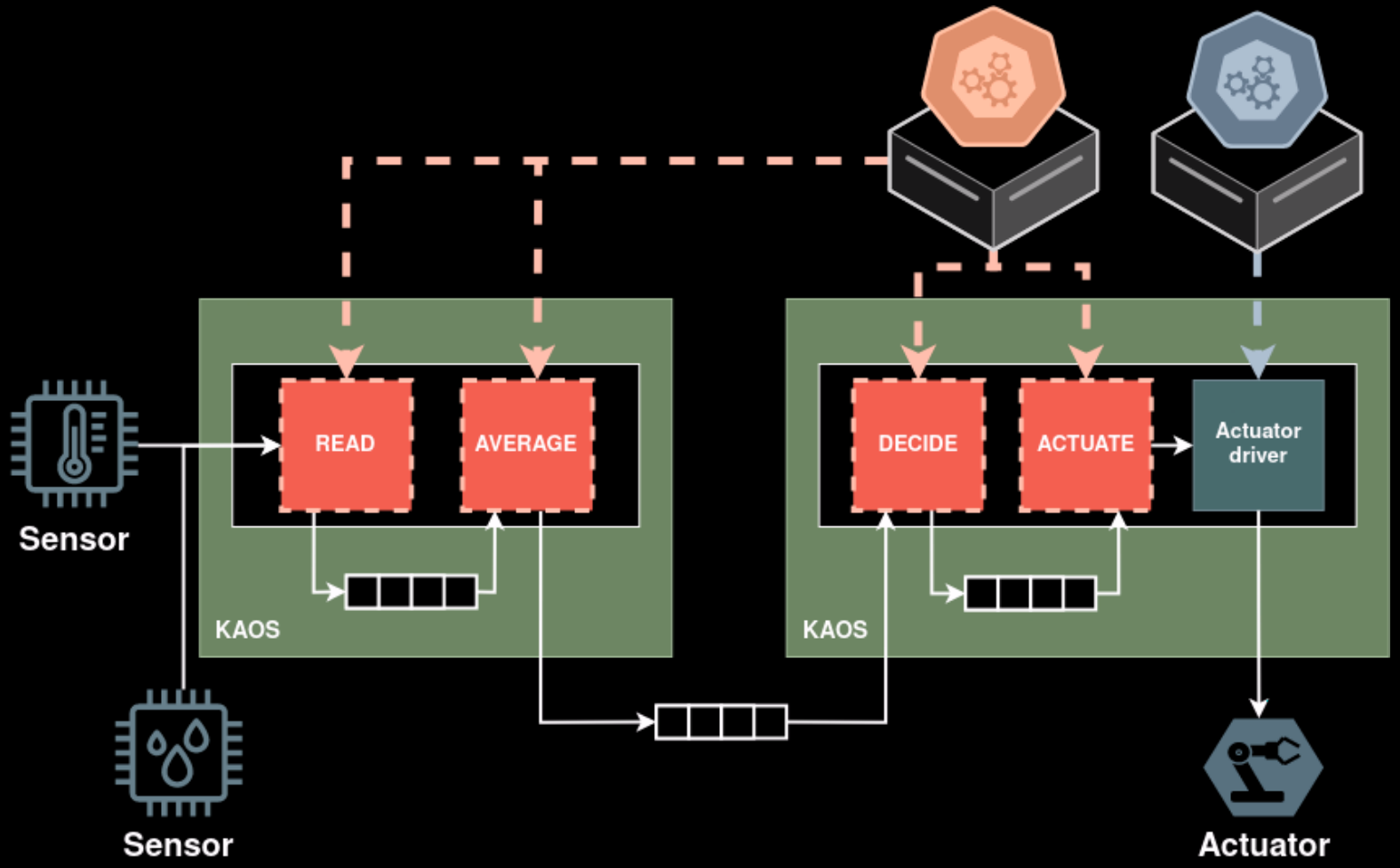


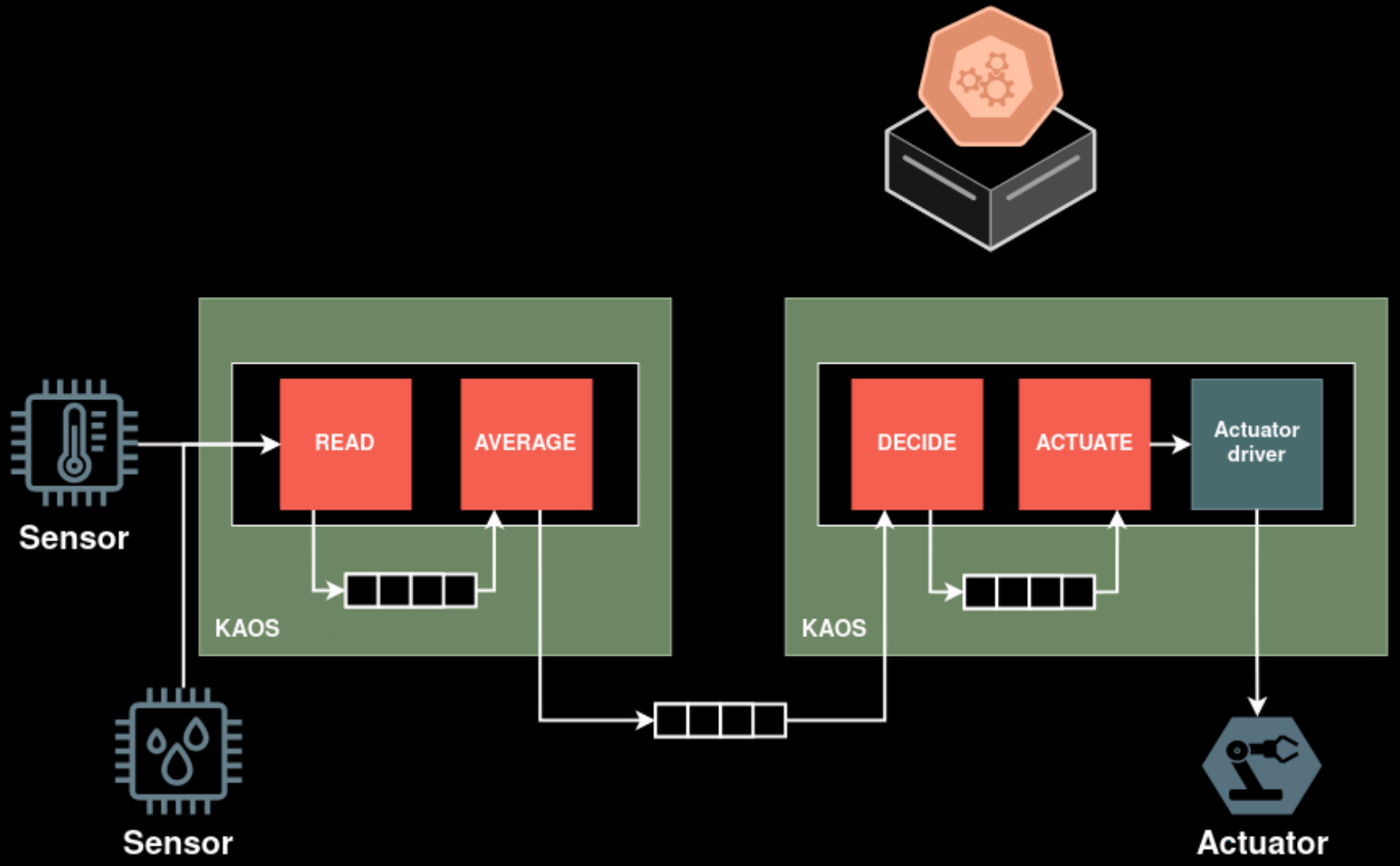
DYNAMIC COMPONENT PLACEMENT

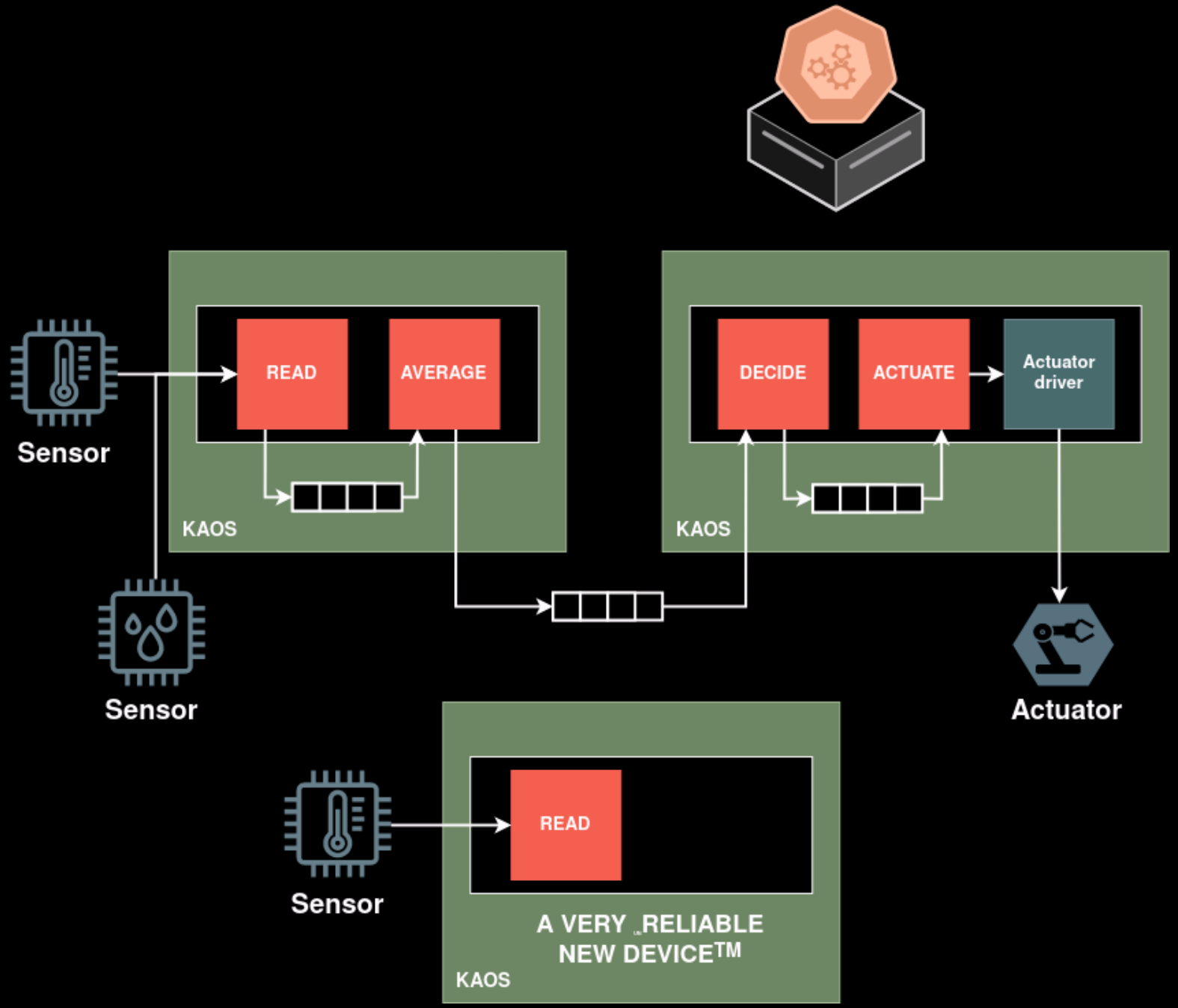
Respond to changes in the system

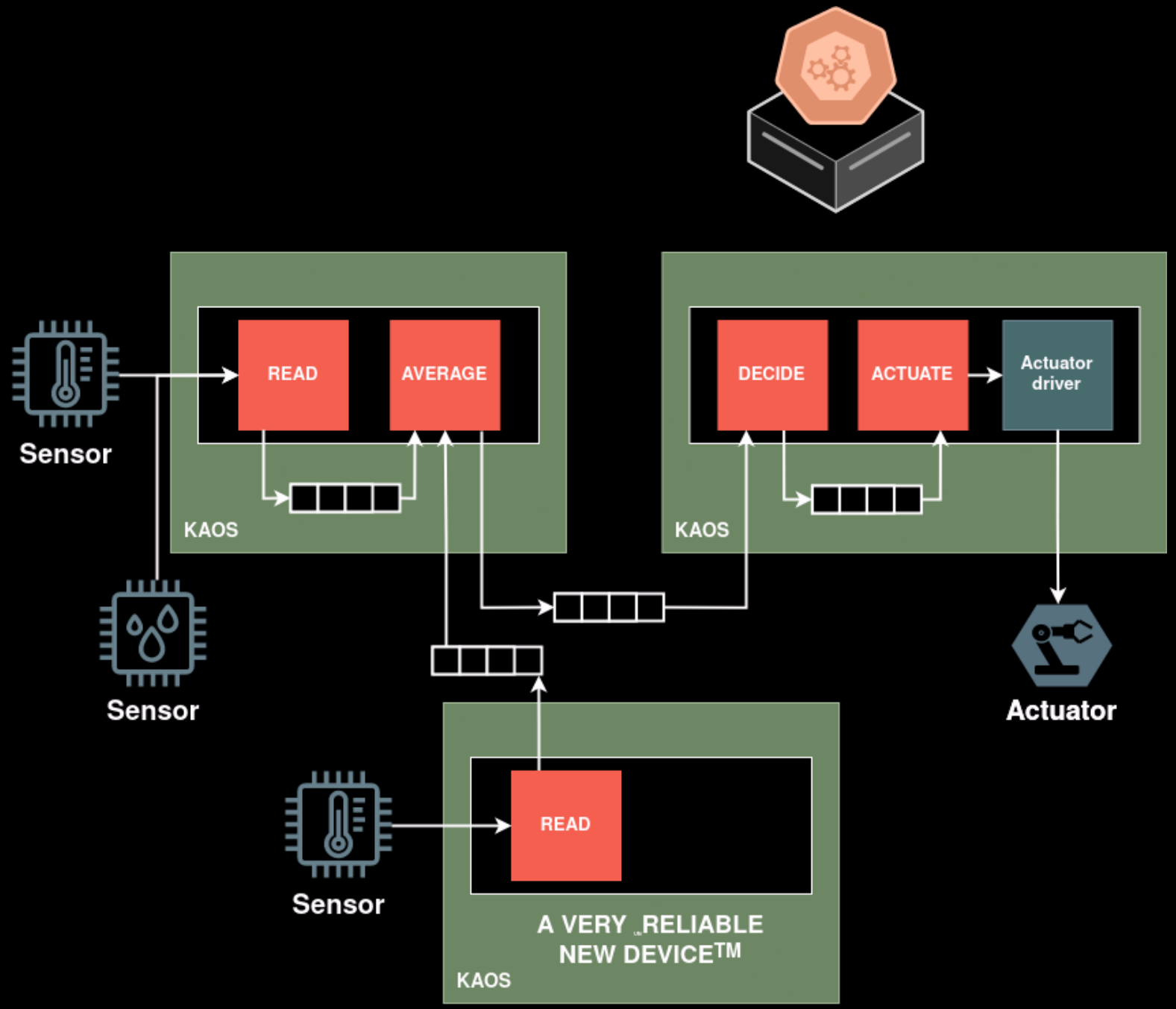
Tolerate unreliable hardware and links

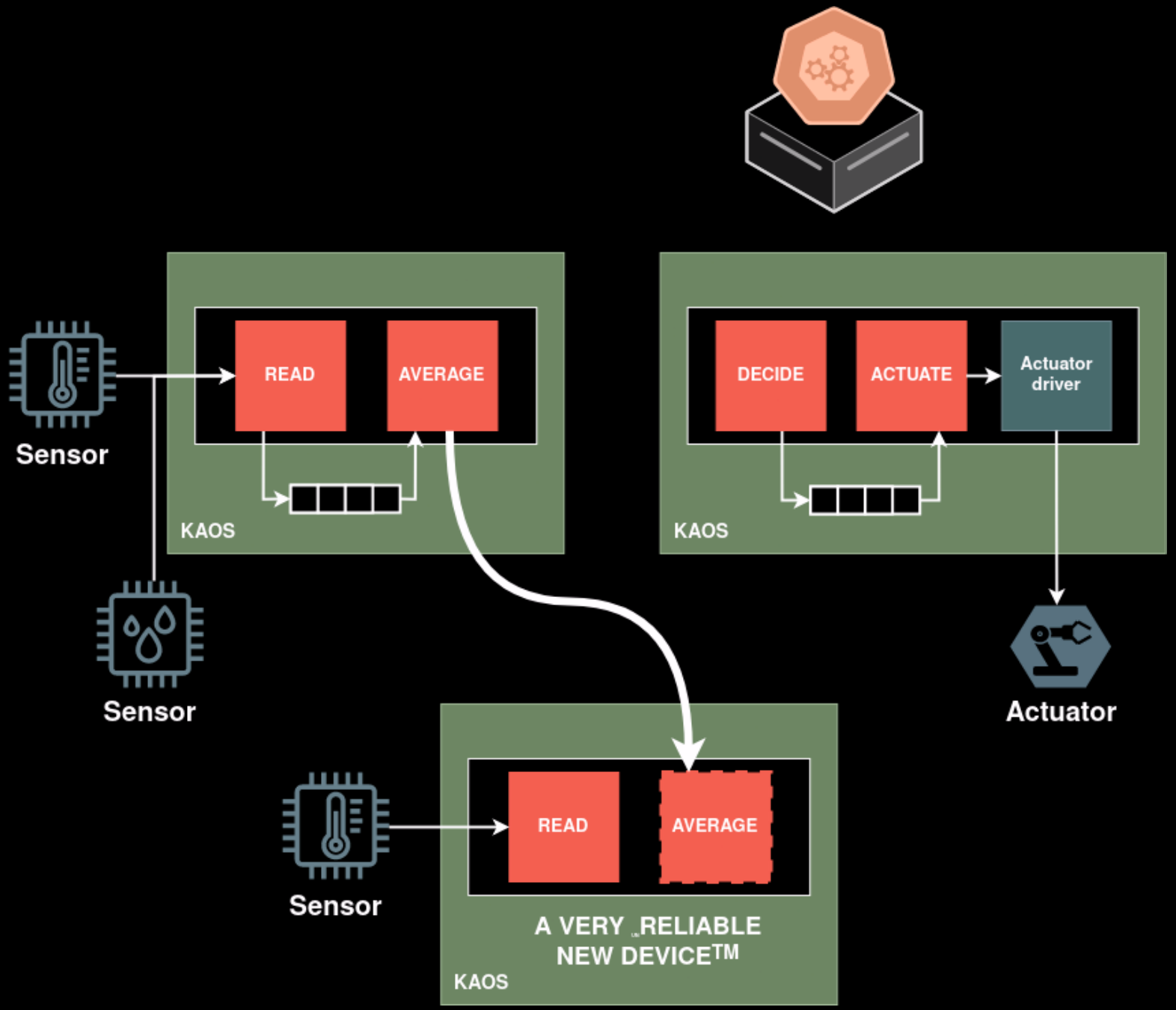


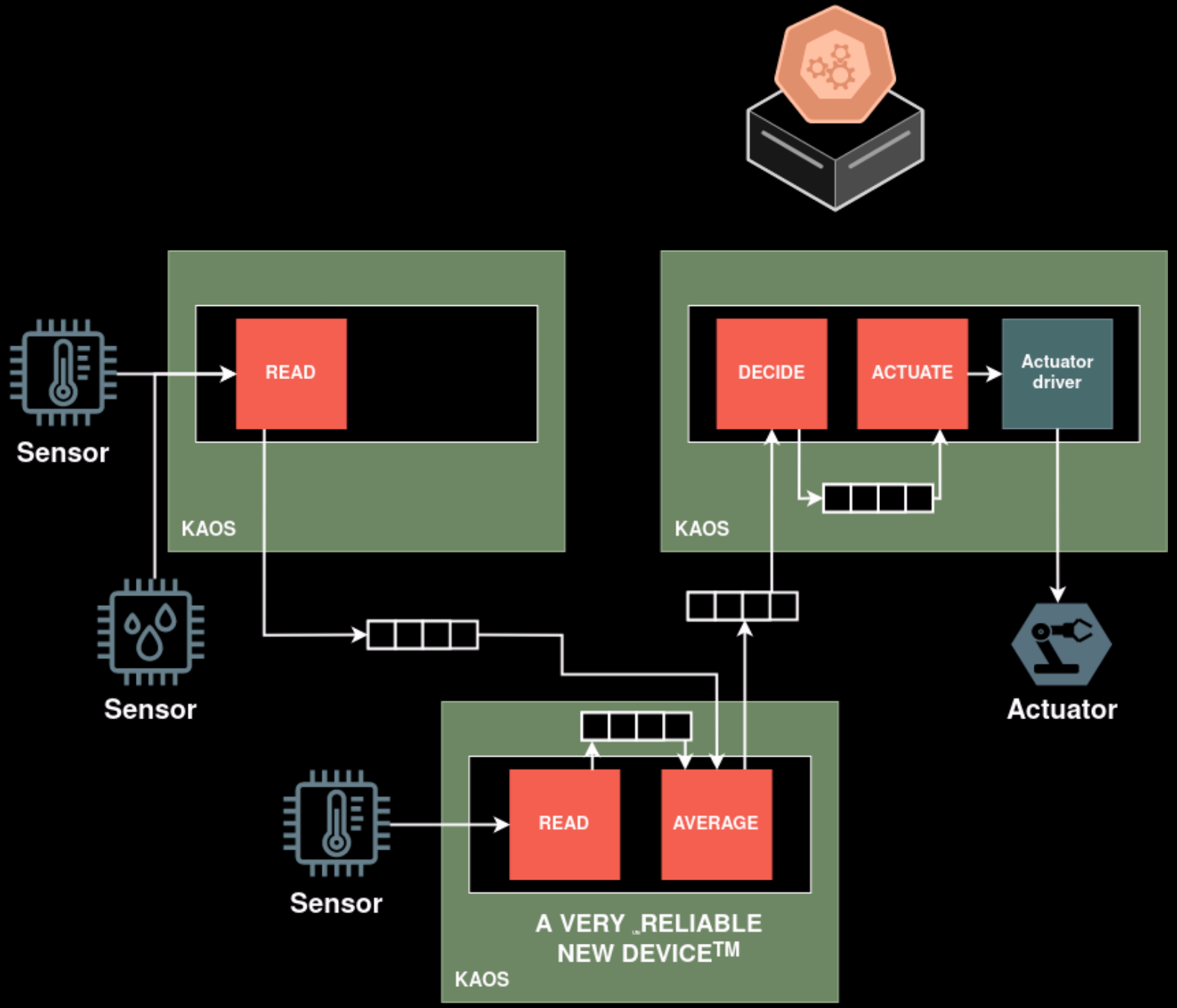

READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator

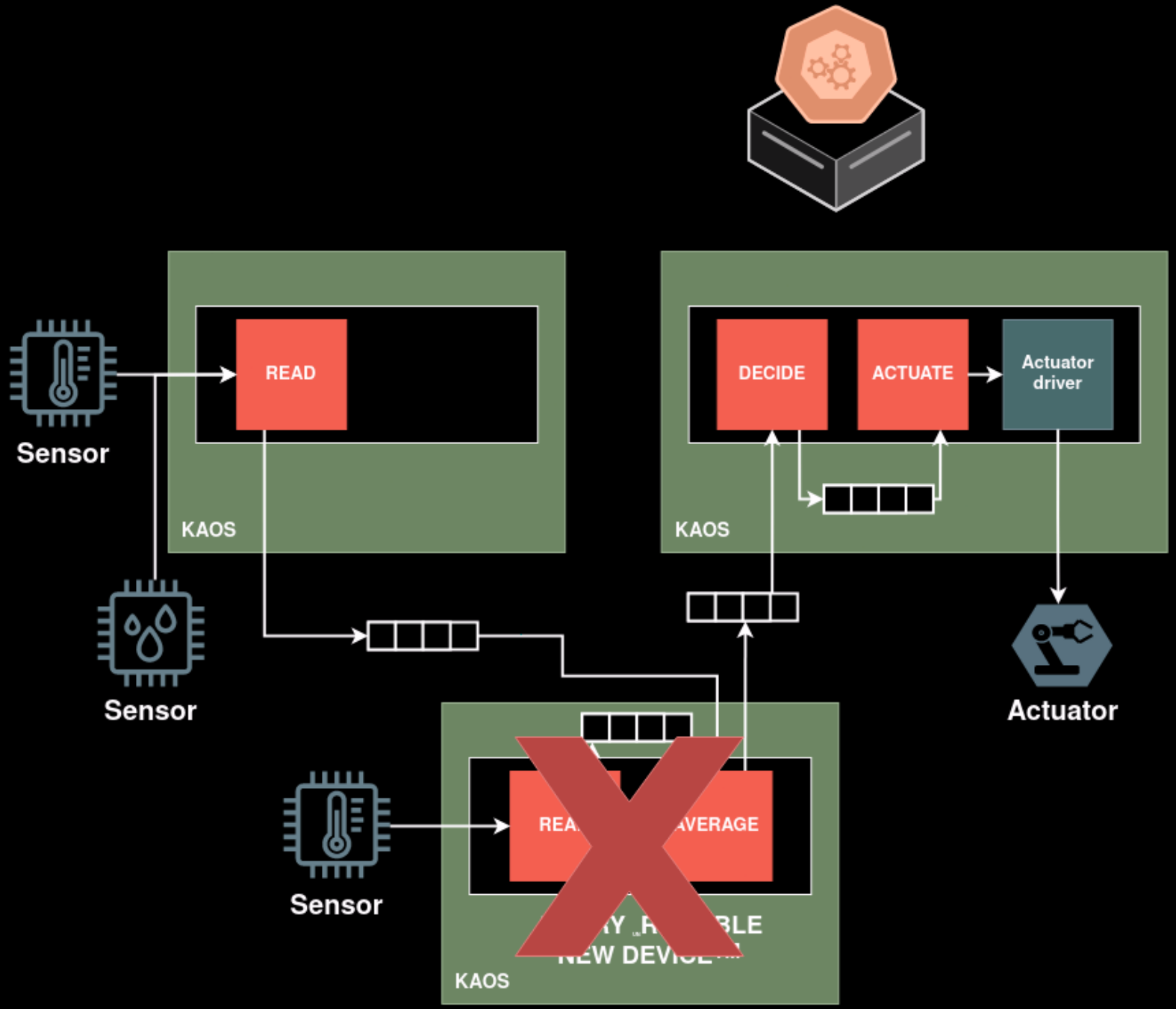

READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator

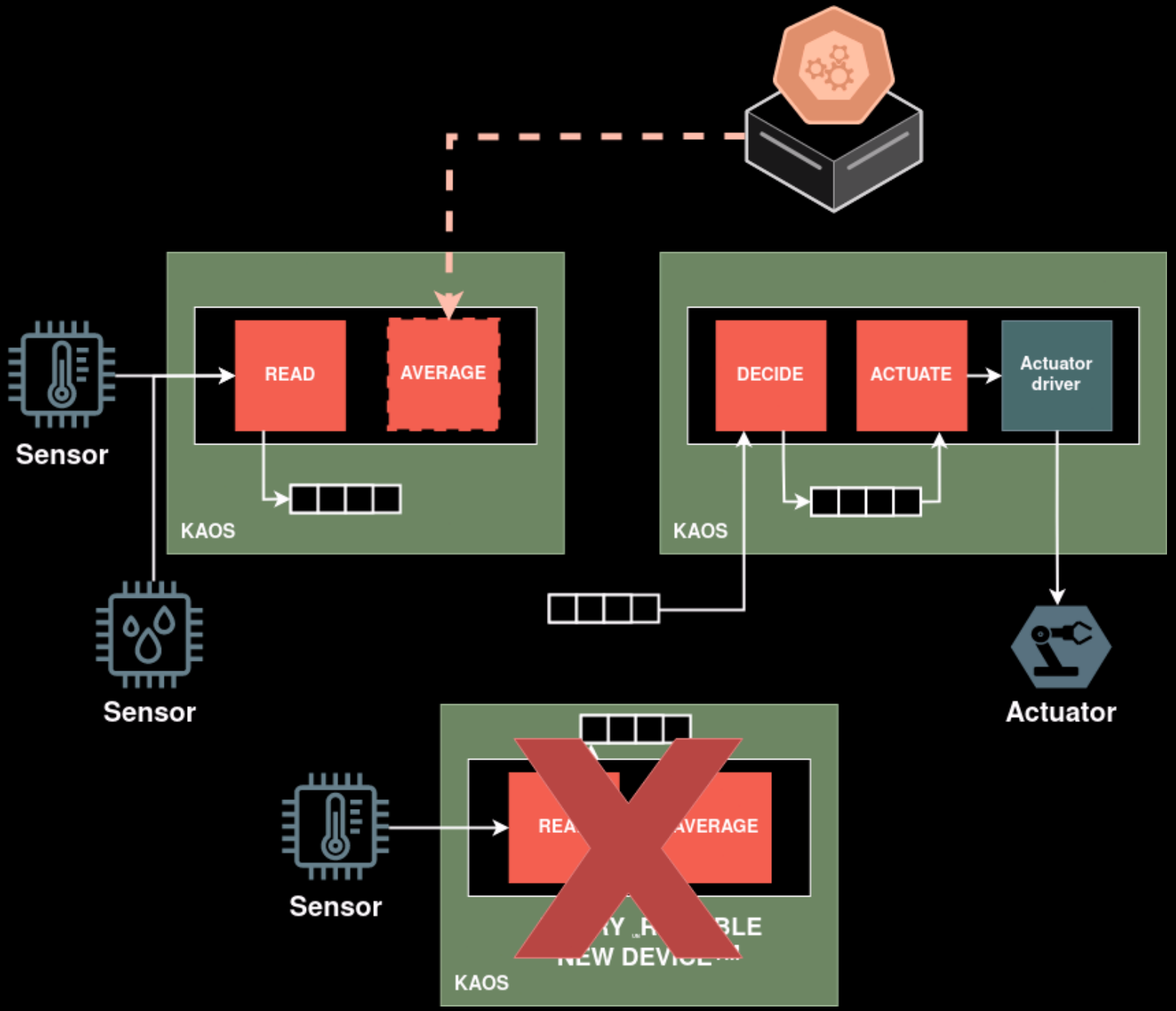

READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator
READ
A VERY RELIABLE NEW DEVICE™
KAOS
Sensor

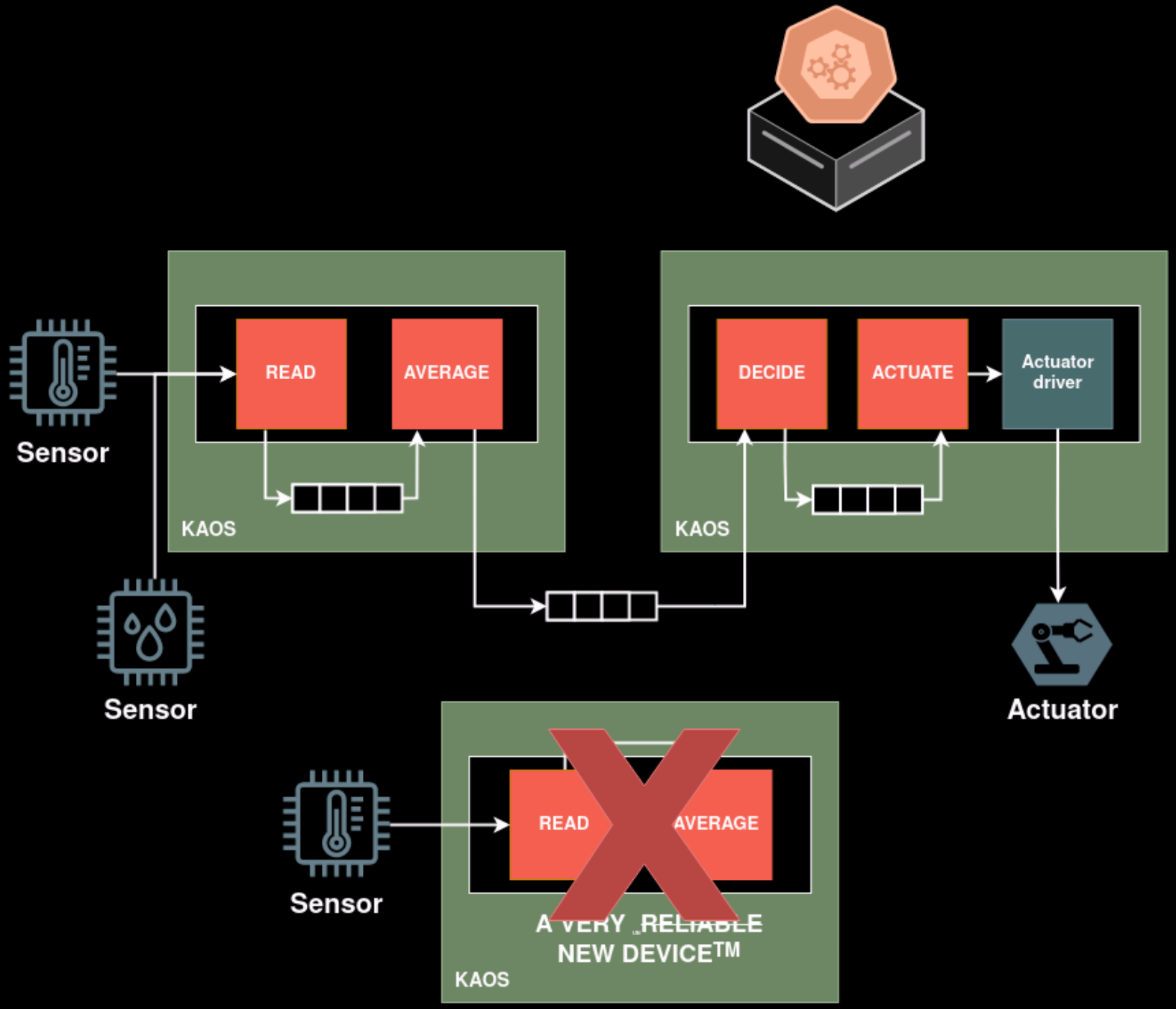

READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator
READ
Sensor
A VERY RELIABLE
NEW DEVICE™
KAOS


READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator
READ
AVERAGE
Sensor
A VERY RELIABLE NEW DEVICE™
KAOS


READ
DECIDE
ACTUATE
Actuator driver
KAOS
KAOS
Sensor
Sensor
Actuator
READ
AVERAGE
Sensor
A VERY RELIABLE NEW DEVICE™
KAOS


READ
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator
Sensor
KAOS

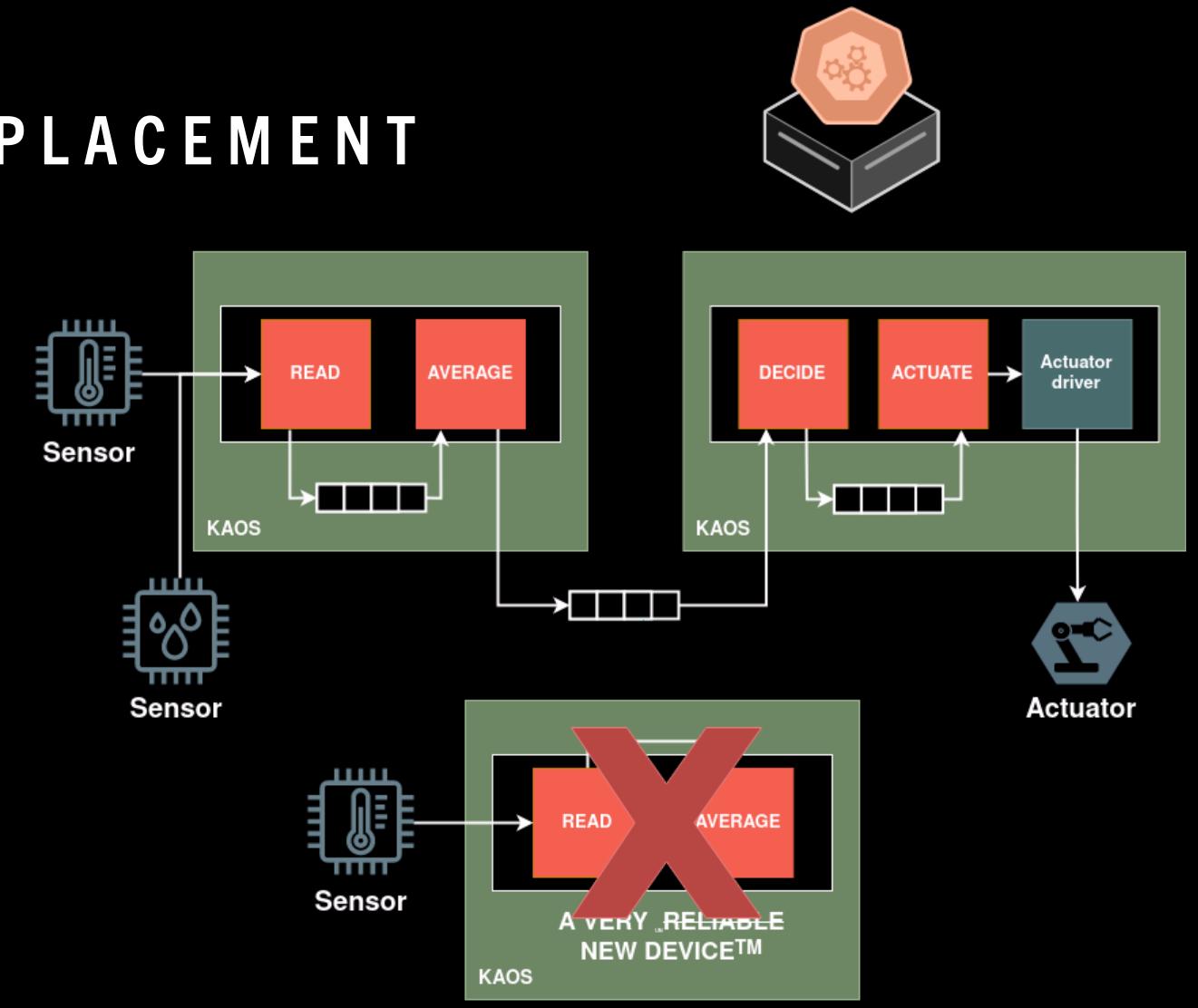

READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator
Sensor
KAOS


READ
AVERAGE
KAOS
DECIDE
ACTUATE
Actuator driver
KAOS
Sensor
Sensor
Actuator
READ
AVERAGE
A VERY RELIABLE
NEW DEVICE™
KAOS
Sensor

DYNAMIC COMPONENT PLACEMENT

Respond to changes in the system

Tolerate unreliable hardware and links



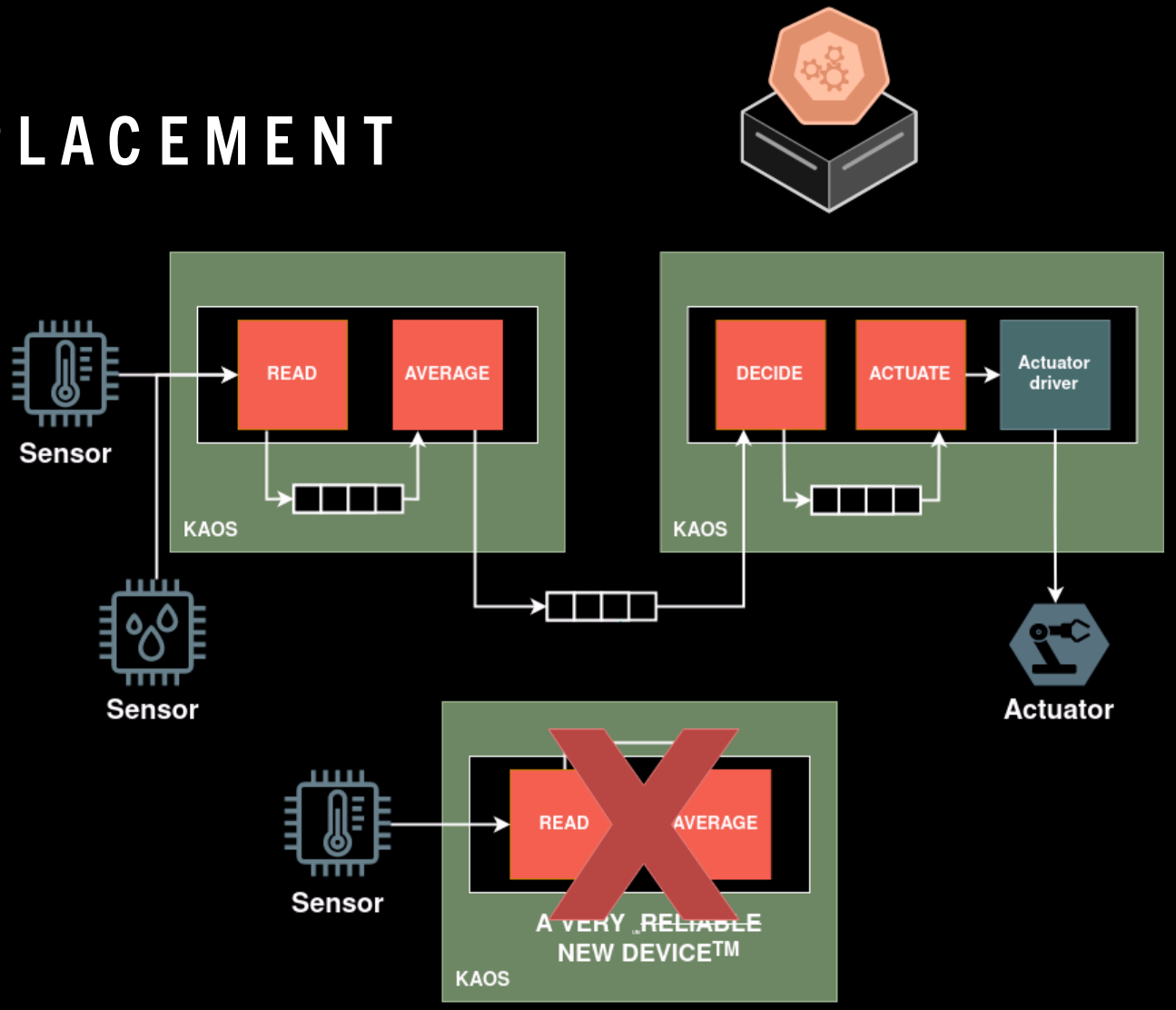
DYNAMIC COMPONENT PLACEMENT

Respond to changes in the system

Tolerate unreliable hardware and links

RESILIENCE

EVOLVABILITY



SECURE & RESILIENT IOT SOFTWARE NEEDS TO OUTLIVE THE HARDWARE

We need:

- **Componentization**
  - Independently evolving isolated chunks of software
  - Managed communication via message channels
- **Portability**
  - Migrating compiled code between different hardware
- **Dynamic component placement**
  - Maintaining functionality in presence of failed or compromised hardware, flaky networks

Preliminary measurements	
Max. number of containers	18
Container memory usage	3.04%
Runtime memory usage	0.18%
PUT domain crossing penalty	8.11x slower
GET domain crossing penalty	3.75x slower
Energy consumption due to virtualization	?

WHY WOULD BUSINESS-PERSON IN THEIR RIGHT MIND DO THIS?

- Existing demand-side interest
- Interest open-source solutions
- Aligning interests between parties
- Industrial building automation
  - Ubiquitous systems need to be long lasting
  - Serious vendor lock in issues

RUNNING THIS FOR DECADES?? I GET A NEW APPLE SMART KETTLE EACH YEAR

- But what about a boiler?
- Ease of maintenance
- Industrial building automation
- Ubiquitous
  - Becoming part of the building fabric

CONCERNS AND LIMITATIONS

- Sleeping devices
- Performance metrics
- Platform portability
  - Runtime
- API
  - Getting the balance right
  - Focusing the application to the right issues without constraining functionality and interoperability