open200

Seven years with Axon Framework

Andreas Grill & Gerhard Hipfinger

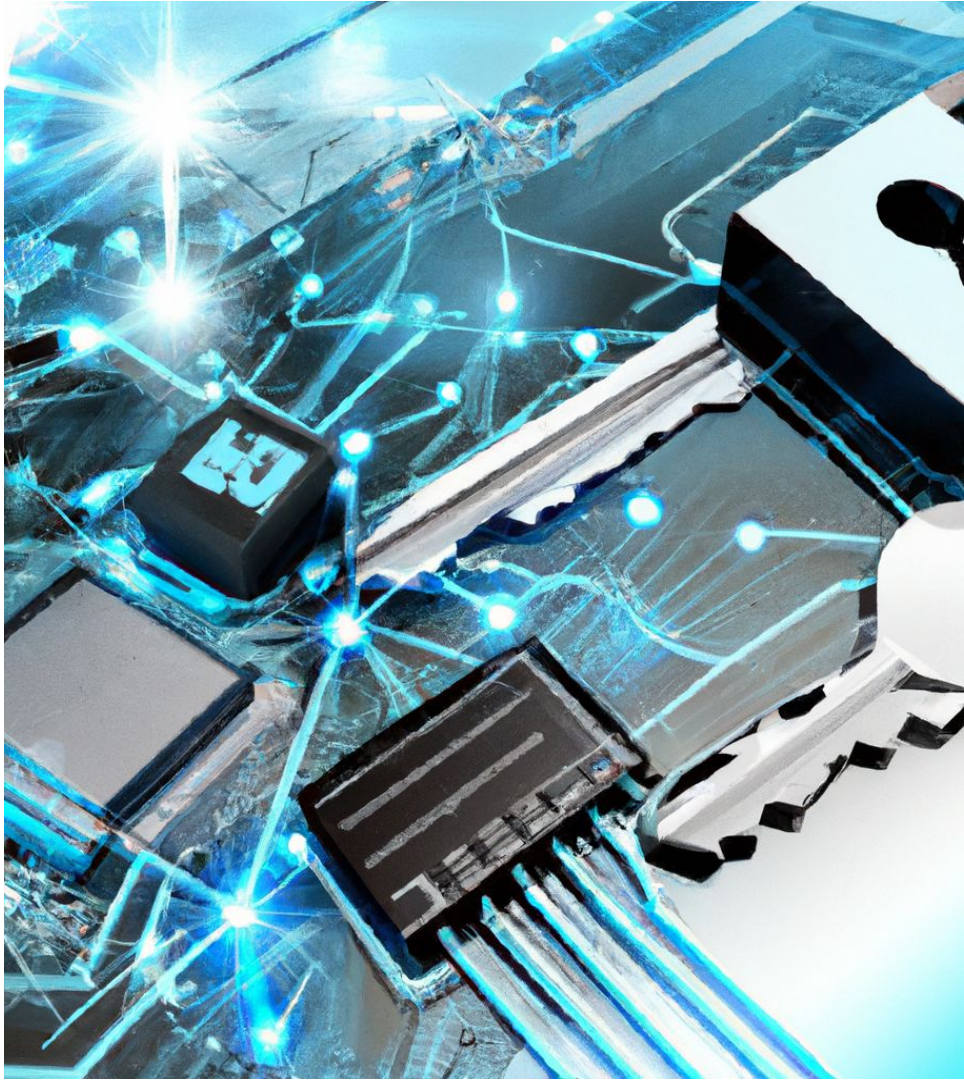
Java User Group Vienna

September 2023

Member of **openFORCE**.com

Our MISSION

CREATE A NEXT GENERATION ELECTRONIC LOCKING SYSTEM!

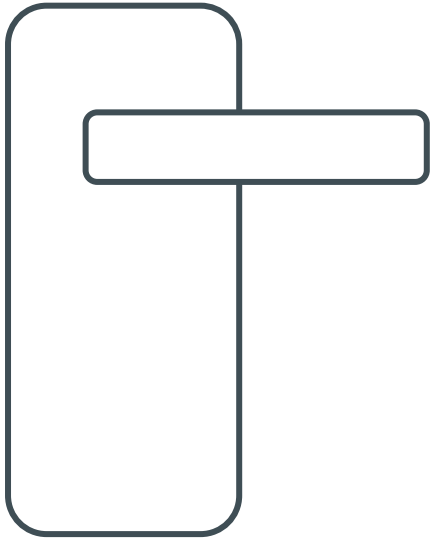


© Copyright open200 ❯ 25.09.2023 ❯ by Gerhard Hipfinger, Andreas Grill

We cannot design the solution until we understand the SYSTEM

Member of openFORCE.com

Let's start very SIMPLE

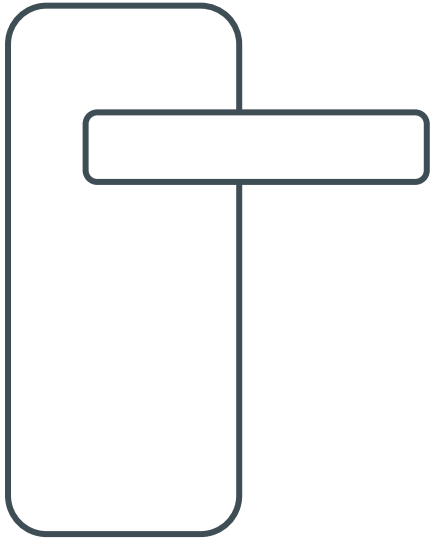


Lock

Keycard

Member of openFORCE.com

Let's start very SIMPLE

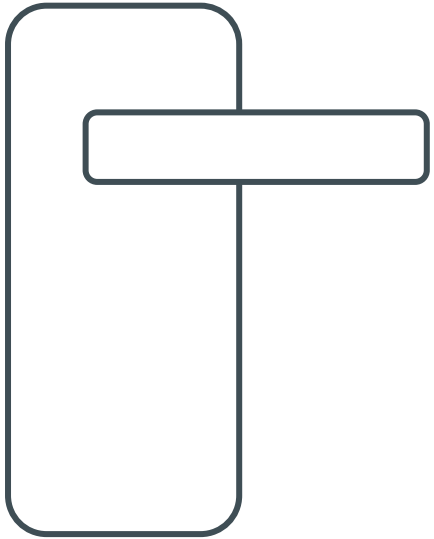


Firmware for NFC, config, storage, motor control, sensors, state management, and more

Contains NFC Interface
Some kB data store

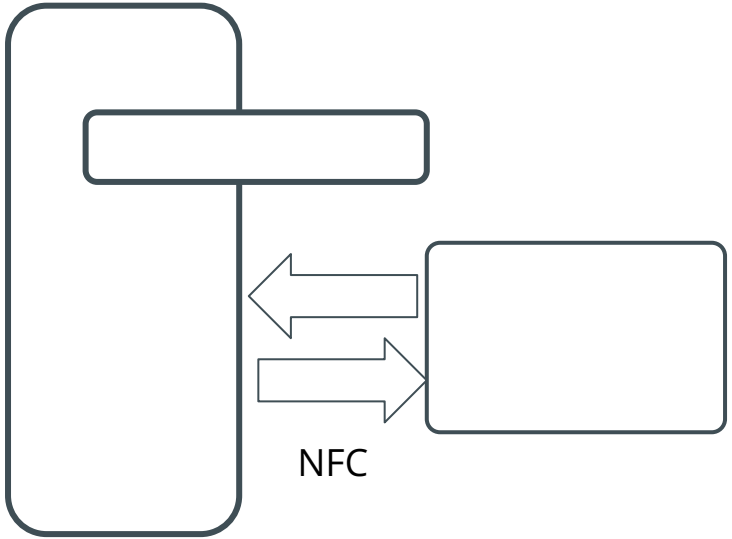
Member of openFORCE.com

Let's start very SIMPLE



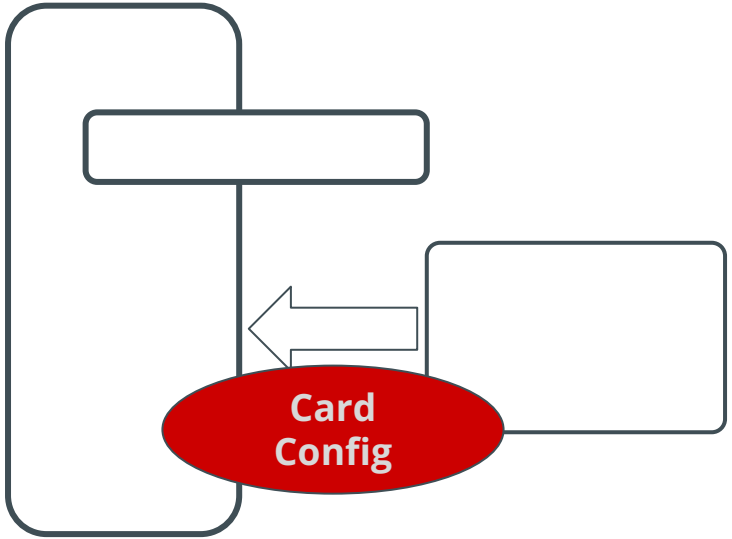
Member of openFORCE.com

Let’s start very SIMPLE



Member of openFORCE.com

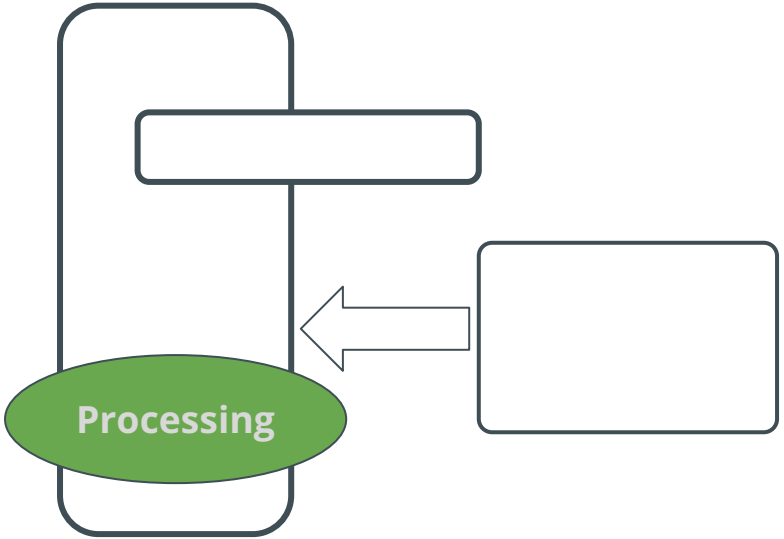
Let's start very SIMPLE



Read data from card

Member of openFORCE.com

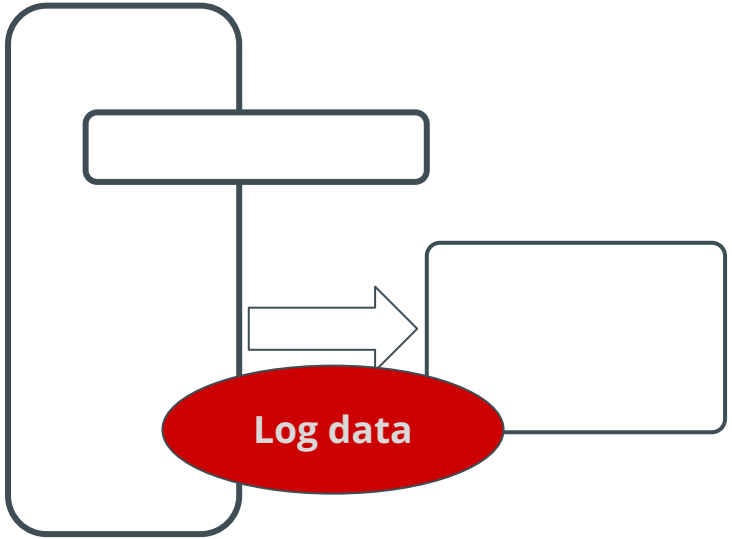
Let's start very SIMPLE



Check permissions
Update config
Open lock or not and signal result

Member of openFORCE.com

Let's start very SIMPLE



Write data to the card

Member of openFORCE.com

Basic CHALLENGES

We have very limited computing power on the edge devices

We have very limited space on the keycards (only a few kB)

We have very limited space on the device (only a few more kB)

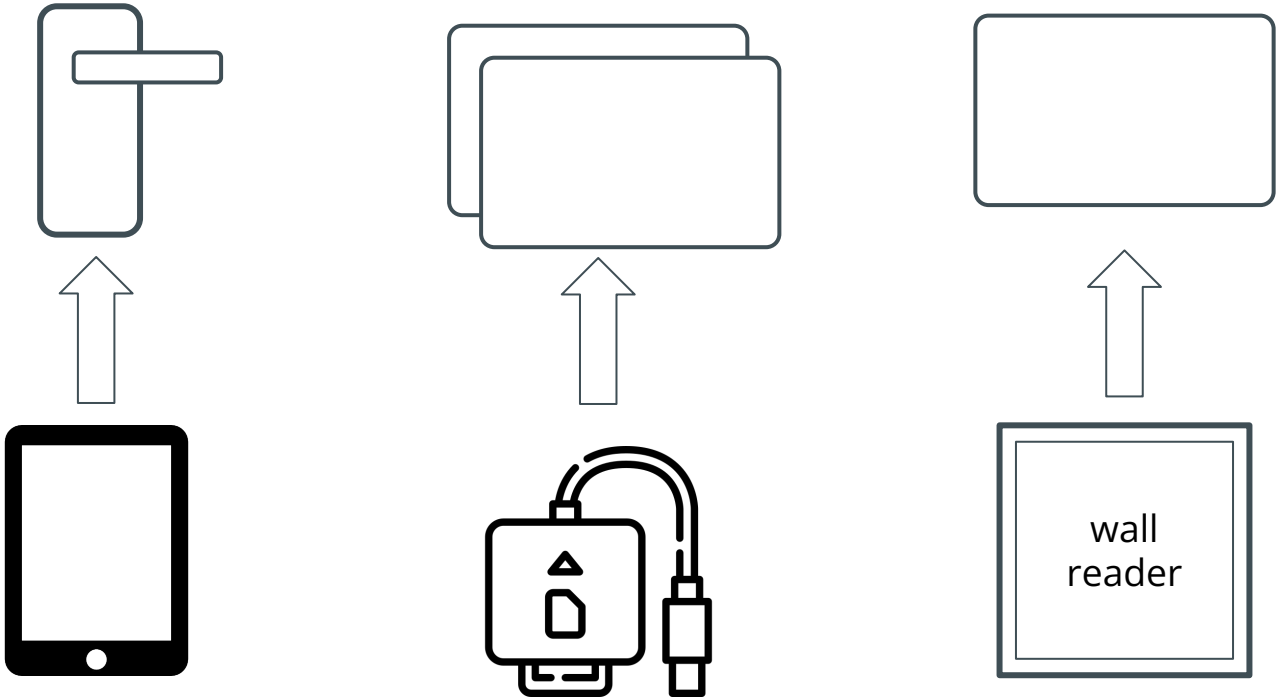
Devices are battery powered - every CPU cycle counts!

We need to be fast: <250 ms worst case for the whole process

And we don't know where the config comes from (and the log files go to)

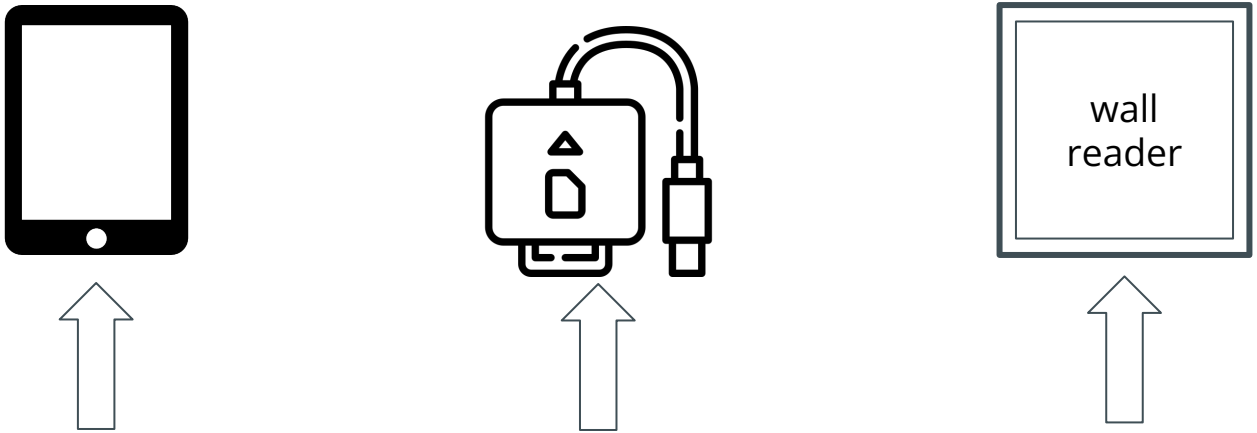
Member of openFORCE.com

Where to get the CONFIG?



Member of openFORCE.com

Where to get the CONFIG?



Our provisioning and controlling software

Member of openFORCE.com

One software to rule them ALL

Manage people

Manage all devices and keycards

Manage all the permissions

Visualize the system state

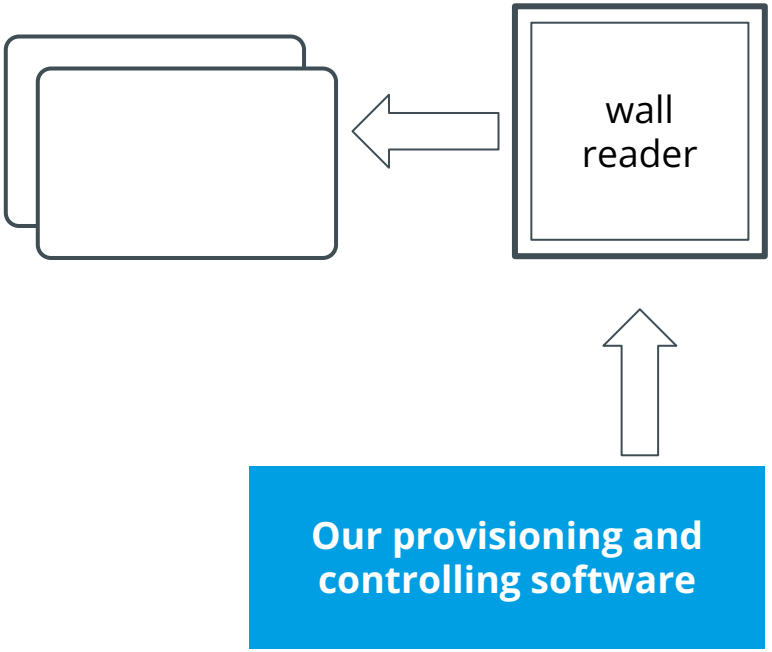
Prepare all config for all devices

Integrate 3rd parties via API's

Handle all the events in the system

Member of openFORCE.com

So let's SCALE UP



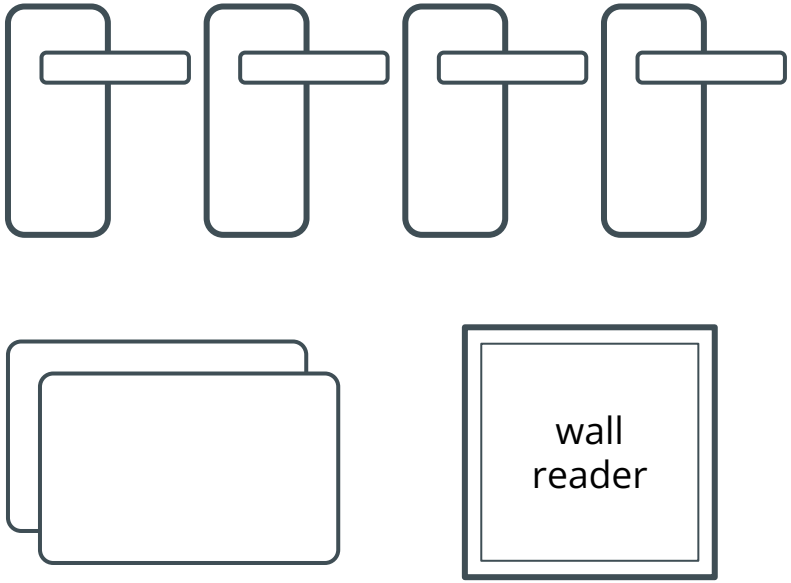
About 50 key cards

Wall reader is online connected with control software

Still need to write card data < 250 ms !!

Member of openFORCE.com

So let's SCALE UP

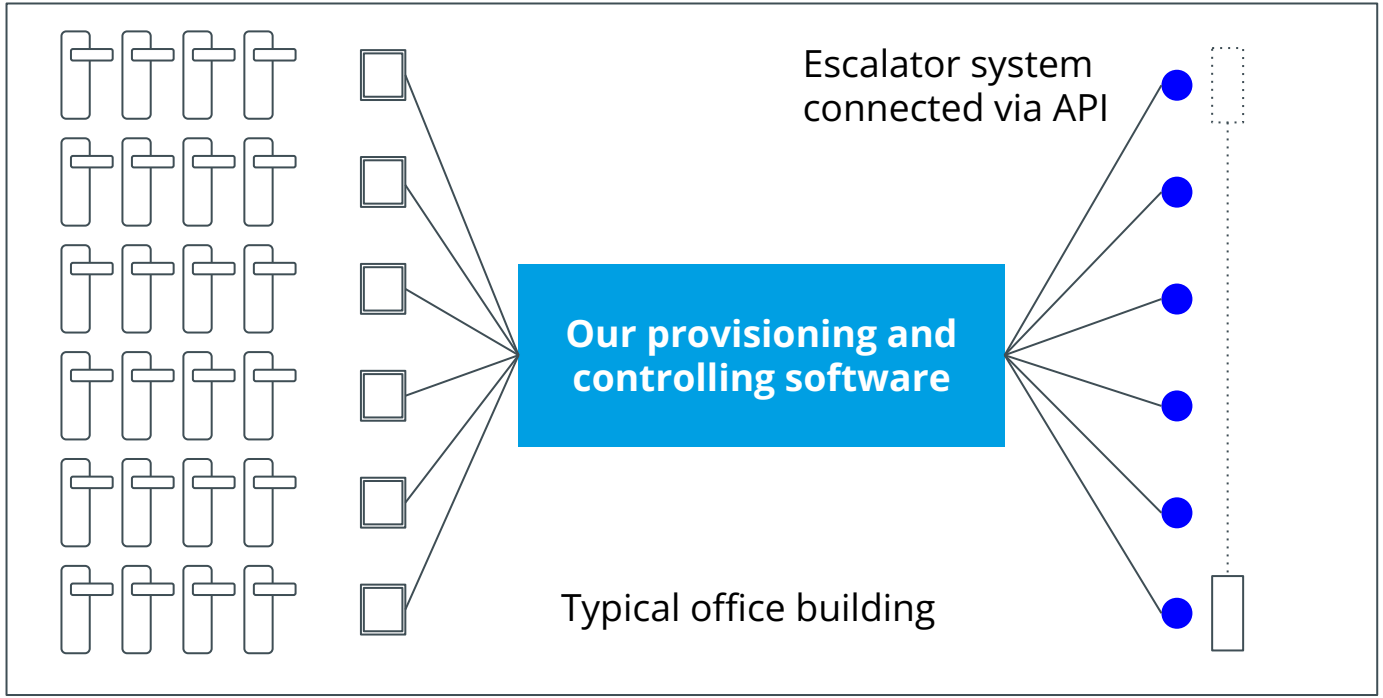


Now we add 2 wall reader

Not really hard yet...

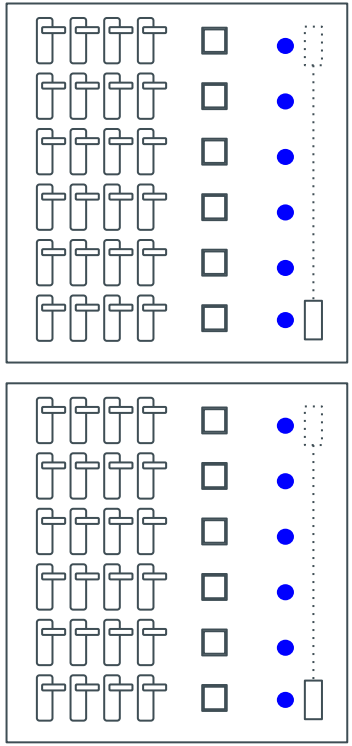
Member of openFORCE.com

So let's SCALE UP

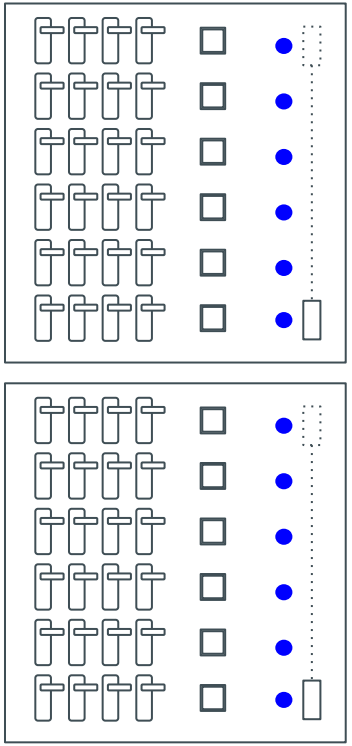


Member of openFORCE.com

So let's SCALE UP

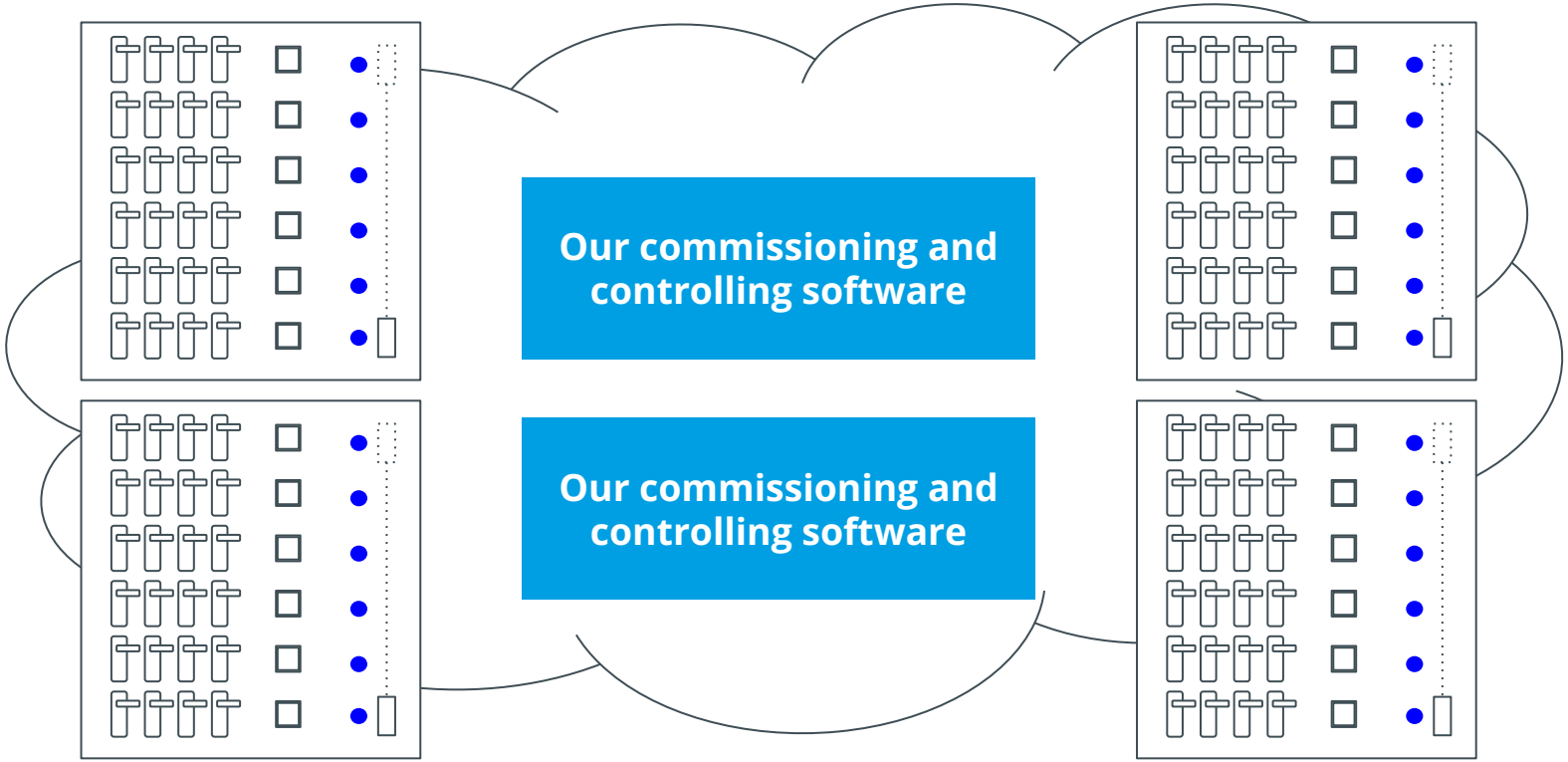


Our provisioning and controlling software



Member of openFORCE.com

So let's SCALE UP



Member of openFORCE.com

The real CHALLENGE

React within milliseconds on keycard presentation

Smartcards ARE the network for offline components

Blacklists must be distributed as fast as possible

We need to derive security state with incomplete informations

With each fact we get to know we need to update the state of the system

Time traveling - why person x was or was not able to open the door y at time z

Support 50.000 keycards, 1.000s of devices

Member of openFORCE.com

Our SOLUTION - let the events flow

During system design we started to talk about events and think in events

We decided to use an event driven design!

We need to write events (fast)

We need to prepare small config data foreach smart card and lock (async)

We have many different views for monitoring and configuration for facilitators

So we drafted our software architecture…

Our natural fit - EVENT SOURCING

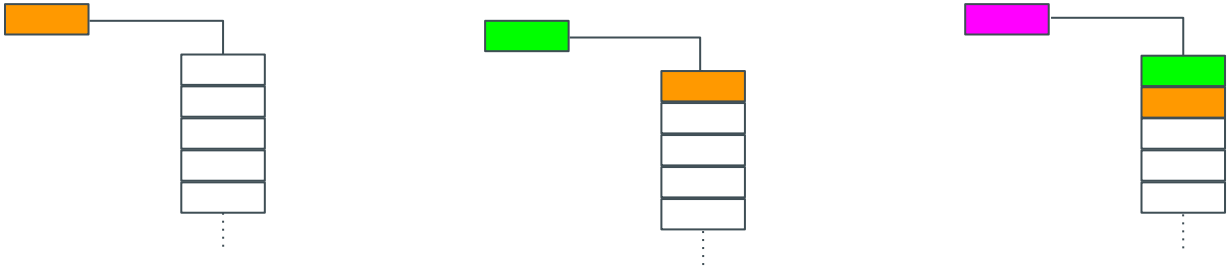
A design pattern to store data (events) in an append only log

Storage can be highly optimized for writing

System state is evaluated by applying logic depending on occurred events

But of course we need to access our data to be useful

Event sourcing alone is missing something...



Member of openFORCE.com

But we need to read data too - CQRS

We need to read the configuration for 1.000s of devices (fast)

We need to visualize huge amounts of data

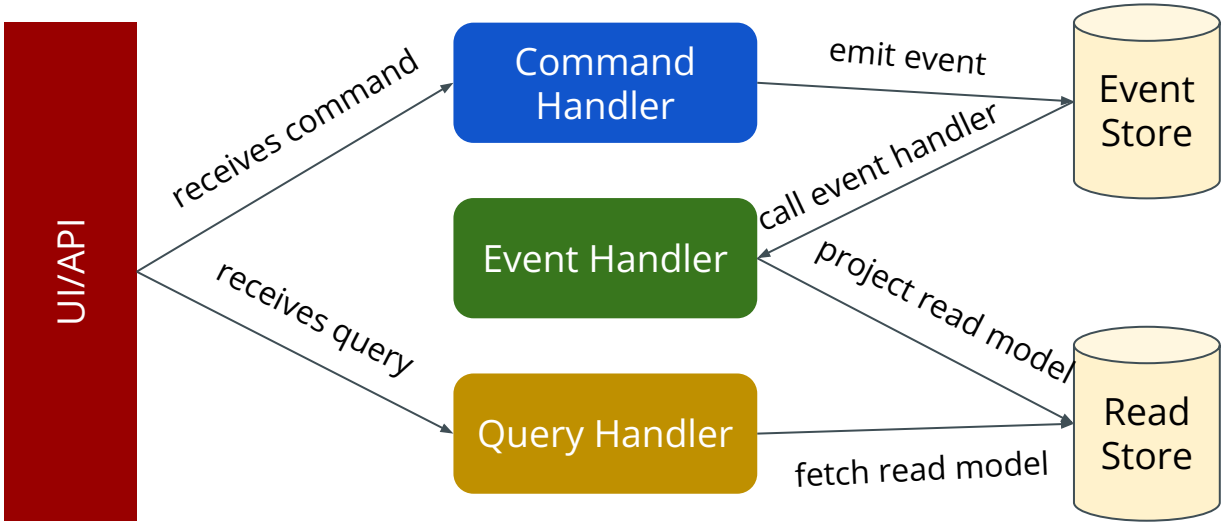
We need to read the data for 10.000s of key cards (fast)

So we need a technique to prepare data for reading

CQRS to the rescue

Member of openFORCE.com

(Just if you are new to CQRS PATTERN)



Member of openFORCE.com

The search for a framework to fit

Member of openFORCE.com

Hello AXON Framework!

Member of openFORCE.com

AXON Framework

Based on DDD principles

CQRS/ES

Event-Driven

Java/Kotlin

Spring Boot

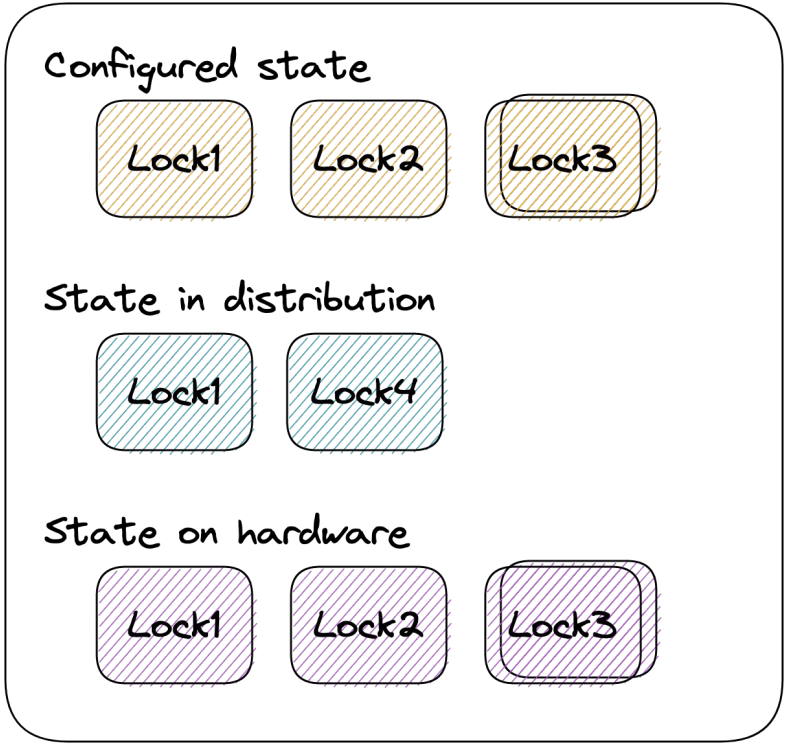
Flexible event stores: Axon Server, relational DB, Mongo, ...

Member of openFORCE.com

What we've LEARNED

Member of openFORCE.com

Event Sourcing



Member of openFORCE.com

Gaps in theory and practice

Steep learning curve

Source of truth is a series of events

Eventual Consistency is challenging

Many things to trip over

Frameworks help, e.g. Axon Framework, ...

Online documentation, Axon Academy, examples, ...

Member of openFORCE.com

Domain Driven Design is a good fit

Event Modeling & Event Storming

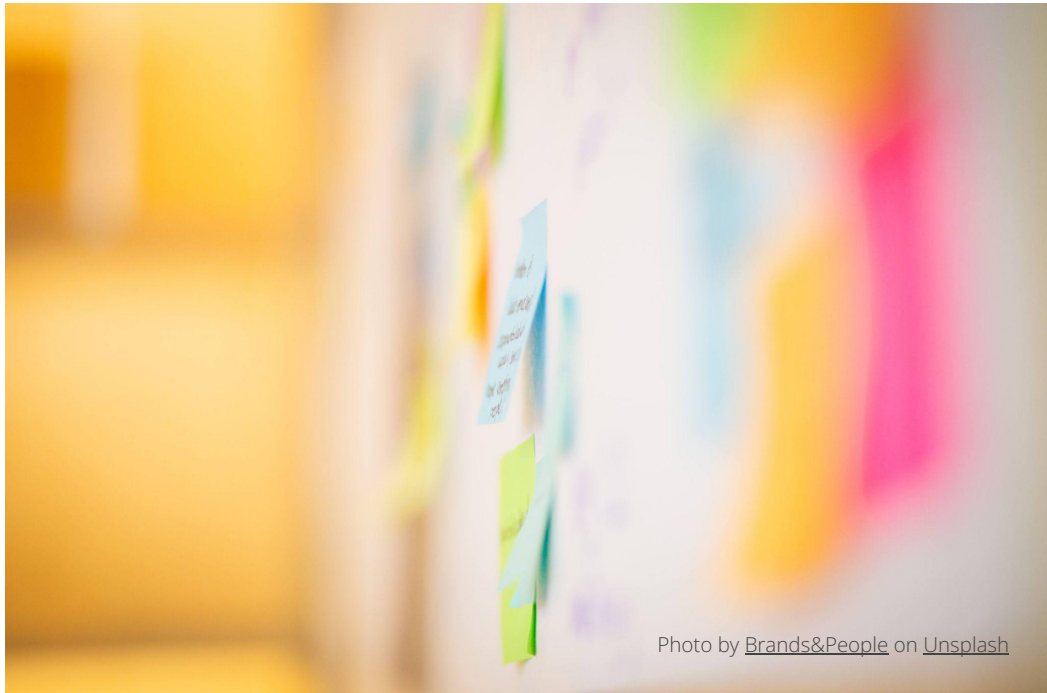


Photo by Brands&People on Unsplash

Is it fast by default?

Optimized fetching

Snapshots on Write-side

Optimize your model

More ways to scale up:

- ***location transparency***
- ***multiple read models, ...***

Things we TRIPPED over

Eventual consistency

Technical events

Unnecessary sagas / business processes

Aggregates with dependencies

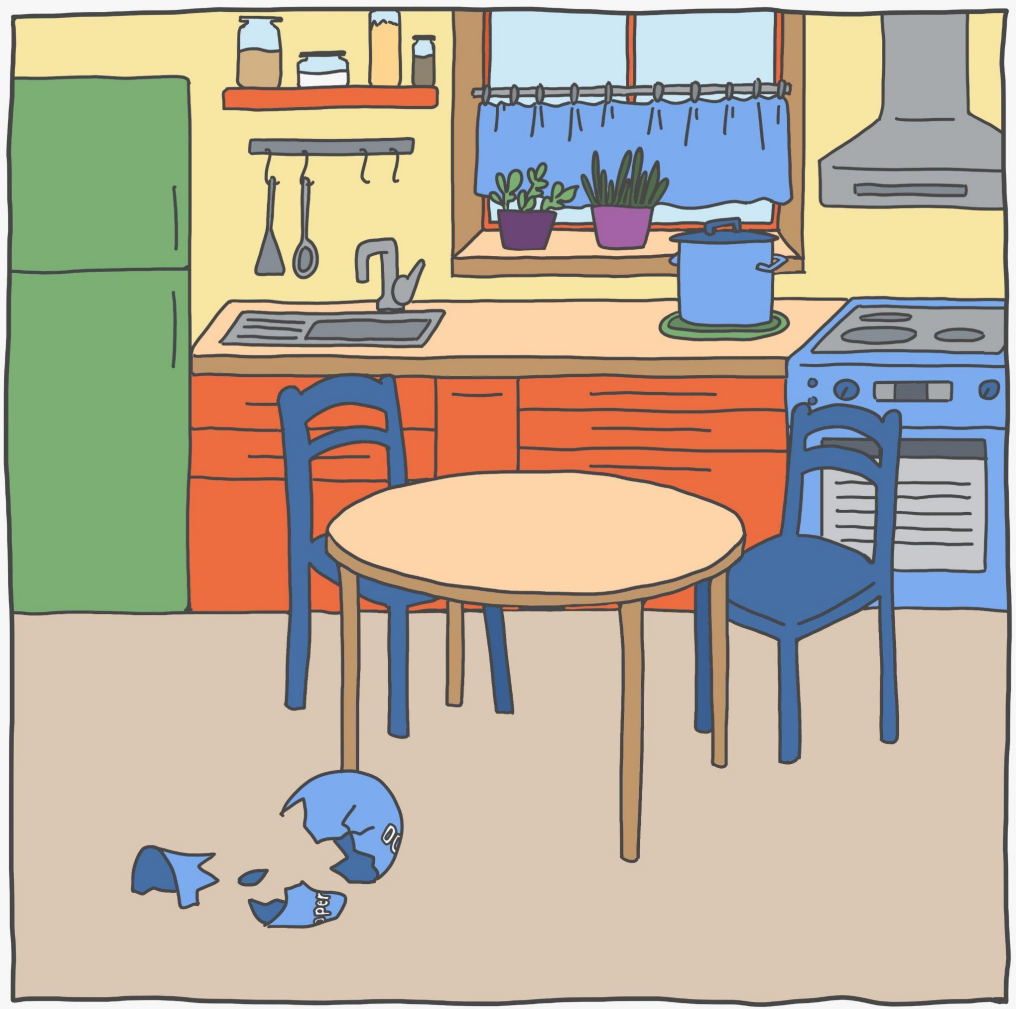
Set-based consistency validation (e.g. enforcing unique names)

https://developer.axoniq.io/w/set-based-consistency-validation

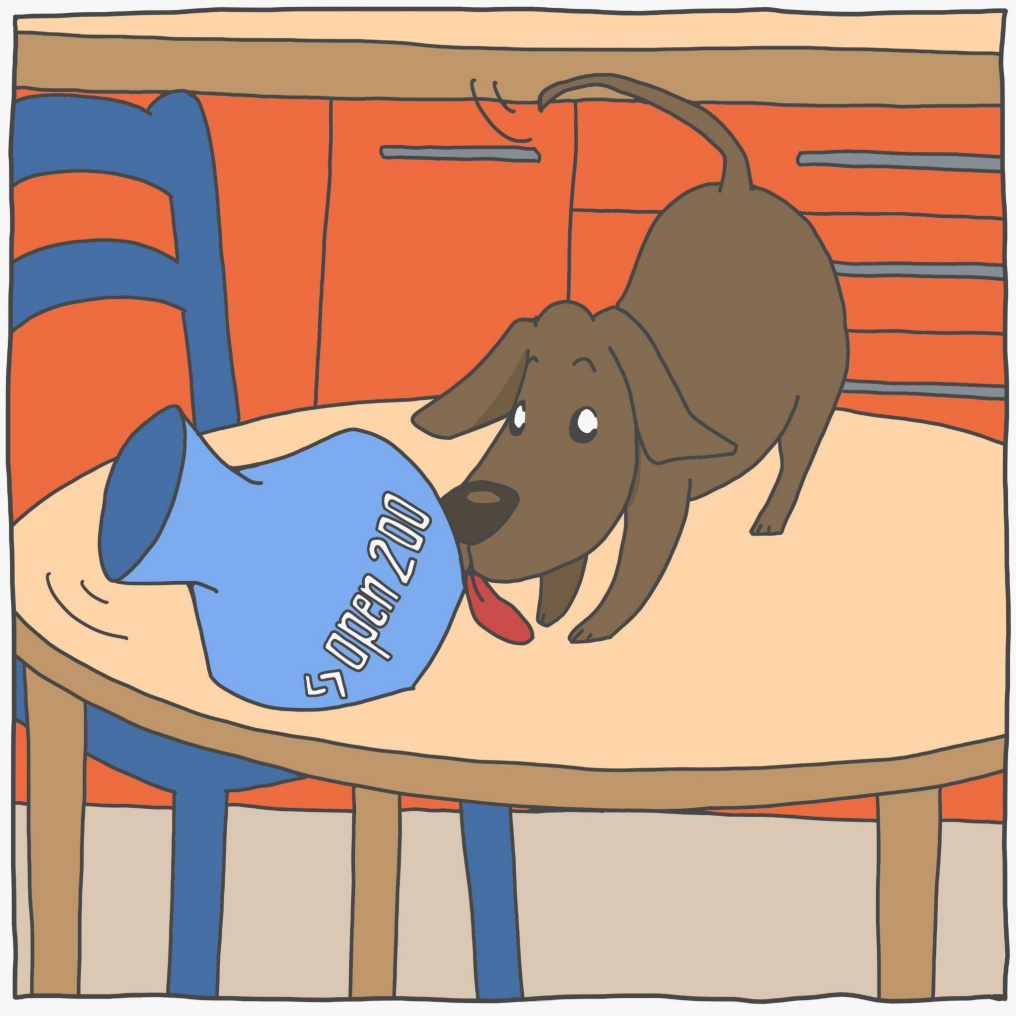
Member of openFORCE.com

Where it SHINED

Photo by Shai Pal on Unsplash



Member of openFORCE.com



Member of openFORCE.com

Member of openFORCE.com

something happened at a customer instance...

they want to know why the application behaved in a certain way

SUMMARY

Don’t reinvent the wheel - use a well know and maintained framework

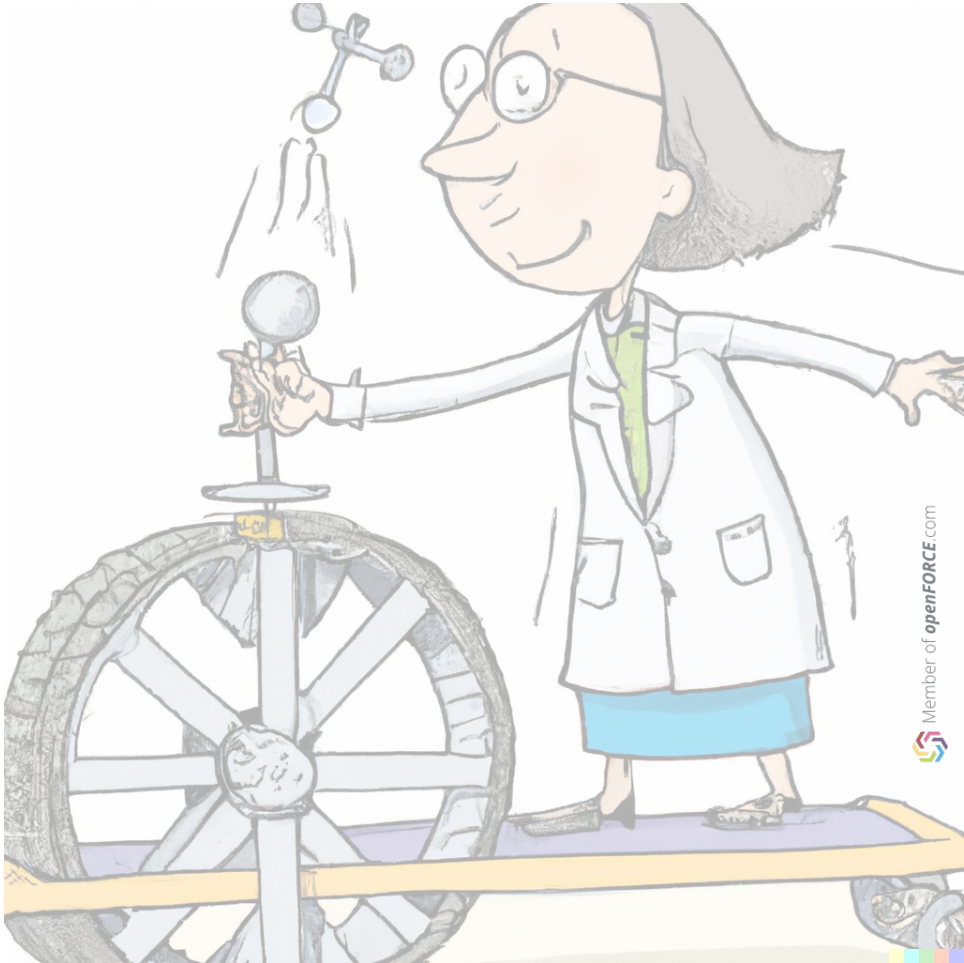
Talk to domain experts and read the blue book

Mind your domain model, focus on aggregate and simple models

Decouple your external API from internal structure

Be aware of eventual consistency

Remove CRUD from your mindset



Questions
Answers

Where to find us

Gerhard Hipfinger

Andreas Grill

open200

Member of openFORCE.com

Attributions

Some images where made with DALL-E

Some images been taken from FlatIcon

Member of openFORCE.com