Events, queues, projections, offline mode and much more

Wim Godden
Cu.be Solutions

Who am I ?

- Wim Godden

Where I'm from

Where I'm from

Where I'm from

Where I'm from

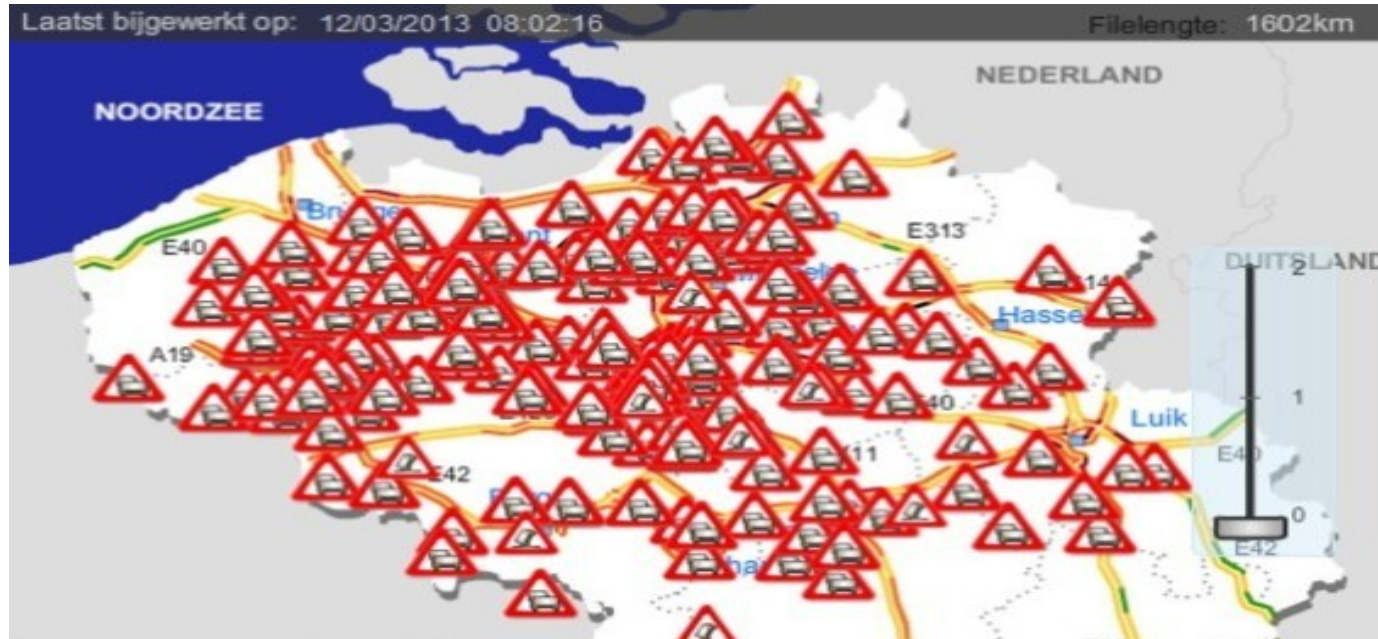# Where I'm from

Where I'm from

My town

My town

Belgium – the traffic

Who am I ?

- Wim Godden (@wimgtr)
- Founder of Cu.be Solutions (https://cu.be)
- Open Source developer since 1997
- Developer of PHPCompatibility, ...
- Speaker at PHP and Open Source conferences

Who are you ?

- Developers ?

- Devops ?

- Experience with event sourcing ?

Events, queues, projections and more

- Architecture
- How do you structure a project to grow in
    - Size
    - Complexity
    - Devices
- Understand what's going on
- Monitor

Short disclaimer

- Many possible approaches

- Let's look at one…

- … which evolved over the years
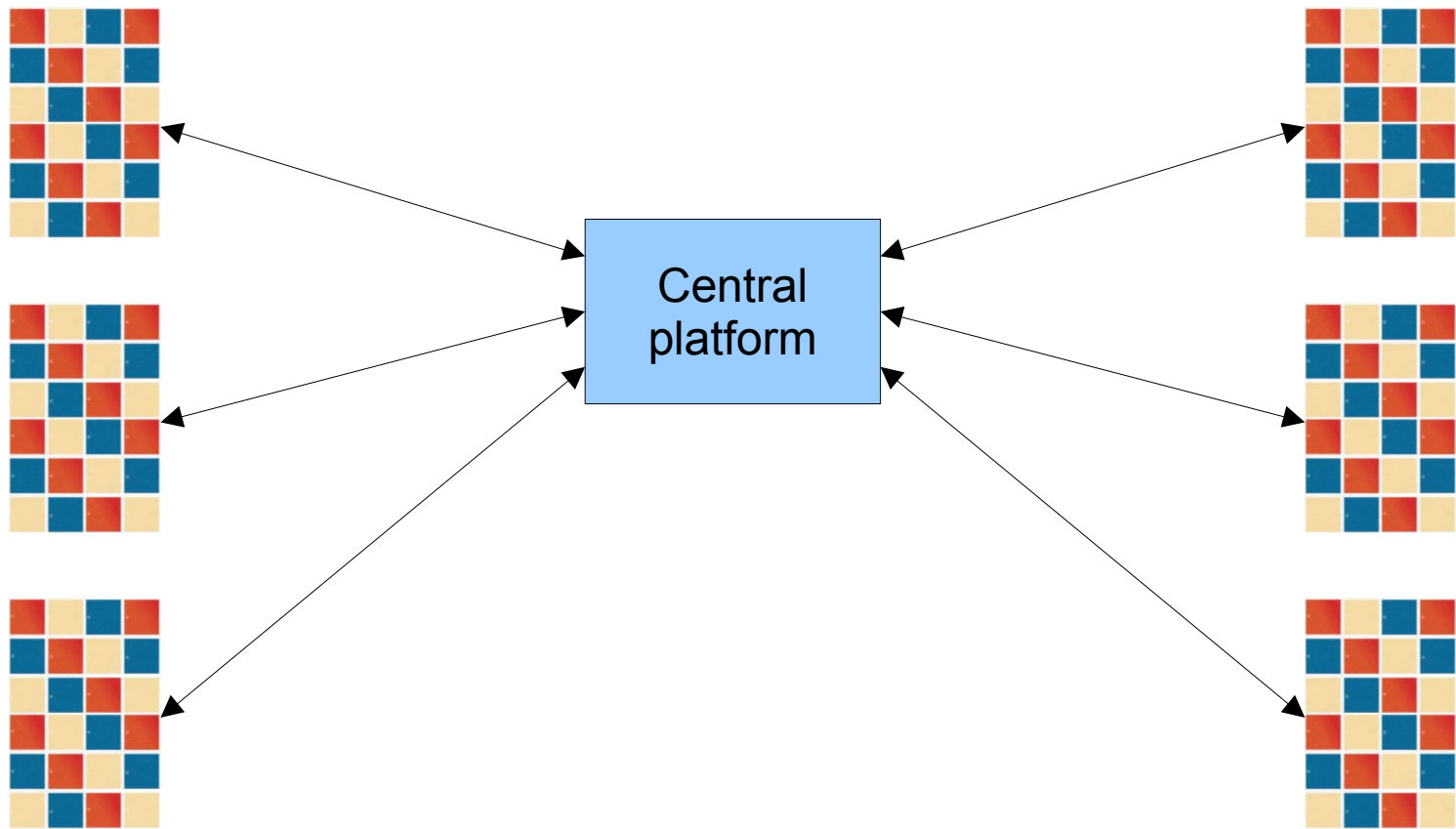
One of our customers

Many installations

# These devices are in the field...

Central platform unites them all

Communication with all devices

1) Push configuration (prices, rental periods, ...)
2) Reserve lockers (through mobile app)
3) Receive events
4) …

1. Pushing configuration to a unit

- Live or queued
- Live : API call → each unit has a unique name
- Queued : background worker attempts every
  - 60 sec
  - 5 min
  - 15 min
  - but only if the unit is online

Central platform

API

Queue

API
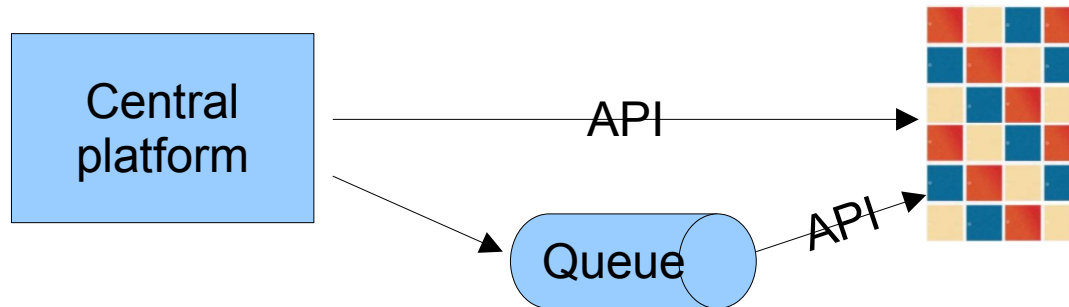
How to detect it's online ?

	NAME ↓	DISPLAY NAME	DEFAULT LOCKER UNITS	LAST RESPONSE TIME	LAST CONFIG PUSHED	LAST ONLINE AT
☐				Offline	7 months ago	4 months ago
☐				55 ms	a year ago	a minute ago
☐				270 ms	4 months ago	a few seconds ago
☐				275 ms	4 months ago	a few seconds ago
☐				16 ms	3 months ago	a few seconds ago
☐				16 ms	3 months ago	a few seconds ago
☐				78 ms	3 months ago	a few seconds ago
☐				Offline	never	10 days ago
☐				Offline	never	10 days ago
☐				39 ms	6 days ago	a few seconds ago

How to detect it's online ?

- 2 ways :
    - Let unit poll the central platform
        - Pro : Fast, doesn't put load on central platform
        - Con : Unit has to "inform" central platform that it's there
    - Let the central platform poll the unit
        - Pro : Central platform "knows" as soon as unit responds
        - Con :
            - Puts more load on central platform
            - Needs to scale...

Scaling process with little processing needs (like ping)

- No need to spin up multiple processes and a queue
- Instead : make the process asynchronous
  - → use multiple threads
  - → send pings to multiple devices at same time
- Unless...

Scaling to...

- 2017 : "Up to 100 over time"
- 2020 : Easily surpassed that...
- 2024 : "Should be ready for thousands"

How to distribute ?

- Geographically ?
  - Per continent ?
  - What about busy regions ?

- By device starting letter ?
  - Never a correct distribution

- Random ?
  - Requires an additional lookup table

- Solution : hash the device name, then distribute :
  - Initially by first letter
  - If you need more granularity later : by first 2 letters
  - etc.

2. Reserve / Rent a locker through mobile app

- Multiple API calls
    - Availability
    - Reserve
    - Confirm
    - Cancel
- Results in events being generated

WS → Central platform → API

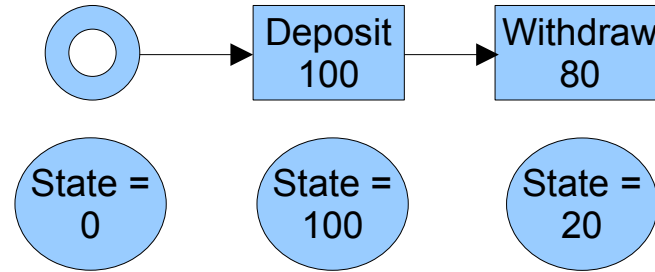WS ← Central platform ← Event

Event sourcing

- Software architecture pattern
- Changes are stored as events
- Events define state

Event sourced system

- Single source of truth

- Sequence of events

- Stored in an event store
  - Could be table in DB
  - Apache Kafka
  - Redis

State

- current state = all events leading up to this moment

- future state = all events from this moment forward
  - Example : reservation of a future booking

Important rules

- Events are immutable
  - Append-only
  - No modifications
  - No deletions
  - Why ? Auditing
    - Theft
    - Accounting

Replaying events = state reconstruction

- Current (and future) state can be rebuilt
  - Just replay all events in order

- Bug in how state was determined ?
  - Fix the bug, replay all events

Note : If you have 100M+ events, this gets heavy ;-)
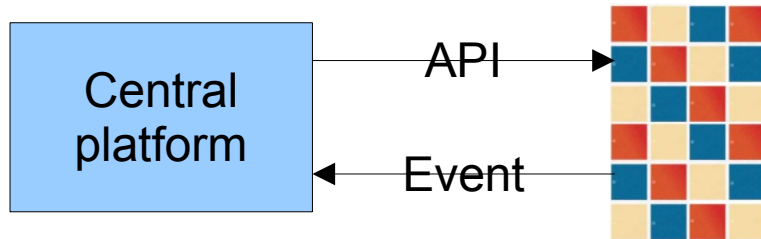
3. Receive events on central platform

Lots of events :
- Locker open/close
- Payment start, success, fail
- Hardware-related events (online, offline, error, …)

How do we get them from the units to the central platform ?

Option A : fetch through API

- Poll every 60 seconds (if online)
- Ask for all new events
  - Update the last event id
- Store the events in the event store
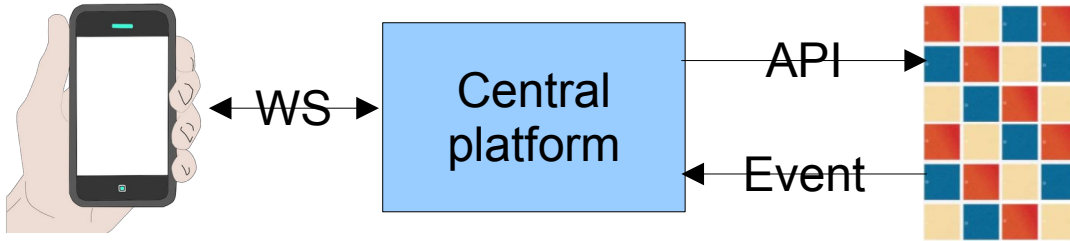
# Option A : fetch through API

- Poll every 60 seconds (if online)
- Ask for all new events
  - Update the last event id
- Store the events in the event store

Option B : push to the central platform

- Push new events (grouped per x seconds)
- User experience = much better
- Careful though :
    - Load on central platform = dependent on external factors
    - High peaks can cause issues
    - Possible solution : use multiple projections to spread load

Central platform ← Event
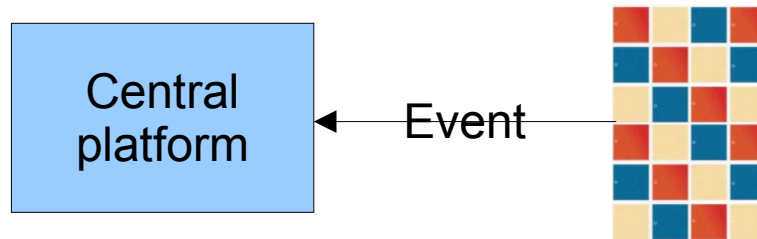
Creating projections

- Separate dataset
- Based on events
- Containing only the data needed for certain retrievals
- Goal : improve performance
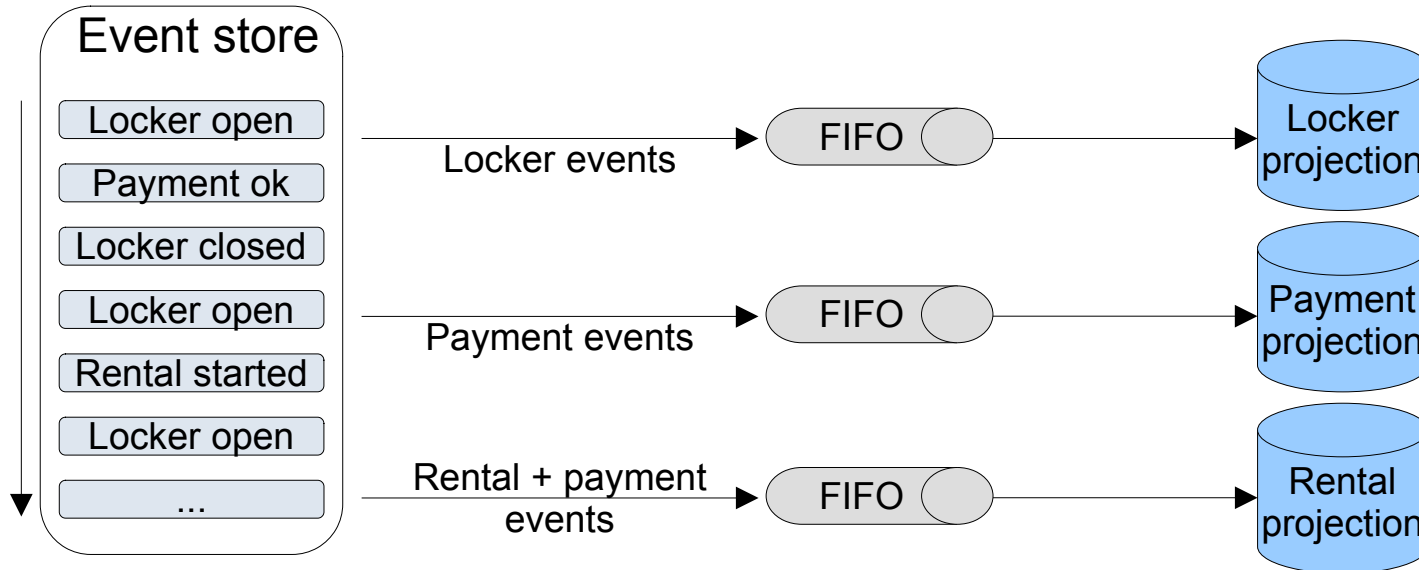
Only need a small part of the data

Lockers / 8

Created At: 23 Apr 2024, 14:20 | Updated At: 12 Feb 2025, 10:51

Locker   **Event log**

⬆ Export Grid ▾   Page: ‹ 8 › of 15   Total of 354 records

LOCATION	OCCURRENCE	UNIT	OCCURRED	OCCURRED TIME ↑
Camping Festitent	NodesController/LockerClosed	A	7 months ago	21:34:59
Camping Festitent	NodesController/LockerOpen	A	7 months ago	21:34:46
Camping Festitent	NodesController/LockerClosed	A	7 months ago	12:17:44
Camping Festitent	NodesController/LockerOpen	A	7 months ago	12:16:54

Projections - pros/cons

- Pros :
    - Faster queries (compare to reading entire event log)
    - Perfect for summary dashboards
    - Bug in projection ? Fix it, then rebuild just 1 projection based on event log
    - Scalable and distributed

- Cons :
    - Asynchronous / eventual consistency → beware of race conditions
    - Requires extra CPU power
    - Data duplication

Online / offline mode

- Design for resilience
- In IOT / Mobile setup : design for poor / no connection
- Put your single source of truth locally
- Users can keep ordering via screen
- Events are stored locally
- Connection returns → events are pushed to central platform

Build it yourself / use existing tools ?

- Build it yourself :
  - Complete control
  - Not dependent on others

- Existing tools :
  - Leverage experience of others
  - Constant evolution and improvements
  - Plenty to choose from : Kafka, Pulsar, Redis Streams, ...

Questions ?

Questions ?

We're hiring @ Cu.be

Full Stack Engineers (Medior/Senior level)

→ https://cu.be/jobs

Contact

- Mastodon   @wimg@php.social
- Slides   http://www.slideshare.net/wimg
- E-mail   wim@cu.be

Thanks !