SENSOR REGISTRY AS A BASE LAYER FOR SMART CITIES

Marc van Andel *, Stefan Bussemaker, Magdalena Grus, Wim Florijn

Kadaster, Hofstraat 110, 7311 KZ Apeldoorn, the Netherlands - (Marc.vanAndel,
Stefan.Bussemaker, Magdalena.Grus, Wim.Florijn)@kadaster.nl

KEY WORDS: Sensors, Registration, Blockchain, Community, Open, Network, Metadata.

ABSTRACT:

There is a growing number of sensors, cameras and measuring devices in the public space. Why are they hanging on the lamp post? What are they measuring? And by whom? Those questions are relevant to the citizens to be assured that no private and sensitive data are collected without their approvement. At the same time the municipalities feel obligated to be transparent about the hanging devices to the inhabitants and provide a good working registration tool to the owners of the measuring devices. The sensor owners would also value the clarity about the process to register their devices and uniformity in the legislation if they plan to install their devices throughout more cities. We cannot forget about researchers, developers and data scientists who would highly appreciate the transparency about the measuring devices and the potential access to the data from the sensors.
A National Sensor Registry (SensRNet) seems to be the solution to answer the abovementioned questions. The registry would: provide transparency to the municipalities and citizens about the data collected by the devices and the purpose for collection; provide overview and insight into where sensors are placed in public space and who is the owner; allow sensor owners to register the devices in a uniform way; provide access to highly demanded data to utilize the smart city concept; act as a platform that provides transparent, safe and secure environment where citizens and entrepreneurs can get more information or make objections against the reason behind collecting data.

1. WHY

1.1 Introduction

The increasing number of sensors, cameras and measuring devices in the public space is undeniable. This is expected to increase even more, and it might even become a necessity in our digitalising world to support all automated processes. Still, this does not dismiss local governments from their task to provide safety and security to their citizens and their rights to privacy. Municipalities feel obligated to be transparent about the current devices already placed in the public space. Moreover the Dutch law has already obliged municipalities to publish the sensors that bring risk to the people's privacy. Citizens should be able to know where they are 'sensed' and why at every location in their city. That is why a registry of sensors should be available for all citizens, companies, researchers and the government itself.

On the other hand, municipalities have their own autonomy to measure in the public space and develop supporting legislation. Placing, maintaining and managing devices in the public space is also governed by local authorities. Sometimes this is executed by the municipality and sometimes this is delegated to a vendor commissioned by the municipality. Another time this is in collaboration with the citizens and local citizen groups gathered around a certain theme.

Maintaining information about all measuring devices is of local government interest as well. There are huge differences between municipalities in the number of devices and in the process of keeping track of devices. The needs in terms of automation are extremely varied as well, not to mention the variety of systems and automation vendors involved. That is why the registry tool must be filled with the measuring devices administered at local government with maximum flexibility and extendibility to connect and integrate with local IT systems. That is a challenge.

1.2 First steps

The first steps towards the National Sensor Registry have already been made. There were some successful pilots in the Netherlands made by municipality of Amsterdam and municipality of Eindhoven in cooperation with The Netherlands' Cadastre, Land Registry and Mapping Agency – in short Kadaster. There were also a lot of publications and presentations to a wide audience about a common need for a uniform system.

Fortunately, there is a growing interest and support from different governmental organisations like BrabantStad (cooperation of Provincie Noord-Brabant and municipalities: Breda, Eindhoven, Helmond, 's-Hertogenbosch, Tilburg), Apeldoorn, Nijmegen, Zwolle, Utrecht, Rotterdam, Citynetwork G40 Theme group Smart Cities and Kadaster which supports this initiative and works together on sharpening the definition of the National Sensor Register product.

* Corresponding author

Additionally the National Sensor Registry initiative was financially supported by innovation budget form The Ministry of the Interior and Kingdom Relations (BZK) where Kadaster took a coordinating role.

The supporting partners (community) wanted to deliver a first version of the Sensor Register (Minimum Viable Product) in the first half of 2021. Till the summer 2020 we were concentrating on realisation of so-called Walking Skeleton - a demonstration of the complete chain of working components with minimal implementation of functionality and technology. The goal in the second half of 2020 was to make the first version suitable for wider use in production environment(s).

2. THE SOLUTION

2.1 Sensor Registry Network

2.1.1 Central Viewer

The solution for the user interface of a national registry is quite simple. A Central Viewer in which all sensors are visible on the map will provide the transparency and service to the public. Citizens, companies, researcher and governments will be able to see where sensors are placed, what they measure and why, where is the published data to find, in case of open data, who is the owner of the sensor and the legal ground to actually allow 'sensing' in the public space. This information will be published in the central viewer and has to be provided by all local governments.

2.1.2 Local Registry Tool

The solution for maintaining the information about sensing devices for all local governments is a harder quest. In essence this is an application in which information about all sensors in the geographical area of each municipality can be registered. A Registry Tool runs at the municipality itself and is connected and integrated into the whole IT eco-system of the municipality. Therefore, this Registry Tool should be highly flexible and adaptable to be applicable for many different IT eco-systems and set ups. Besides that, it should be able to be connected to other Registry Tools as well of other local governments.

2.1.3 The Network

The solution where it all comes together is the Network. Each Registry Tool is a participant, a node in the network. All nodes are connected with each other and together they form the Sensor Registry Network. In this network the information about sensors is being distributed and shared among all nodes. The Central Viewer is a similar node as each Registry Tool although this will only listen and consume the information from the network. This is called a Publishing Node. Registry Tools can listen and consume information from others as well as providing information into the network to share. These are called Registry Nodes.

2.1.4 Sensor Registry Network

The solution of it all is the Sensor Registry Network. A Network connects all Registry Nodes for maintaining information about sensors in the public space and (at least) one Publishing Node providing a Central Viewer publishing all sensors to the public.
This can only exist if the required environment is existent too. Such an architecture and collaboration will only sustain, if responsibilities and mandates are organised in a proper way. Therefore, some sort of consortium is needed which is clear, open and inclusive. With clear request for change processes and steering to develop the whole stack of components in a sustainable way (Fig.1).

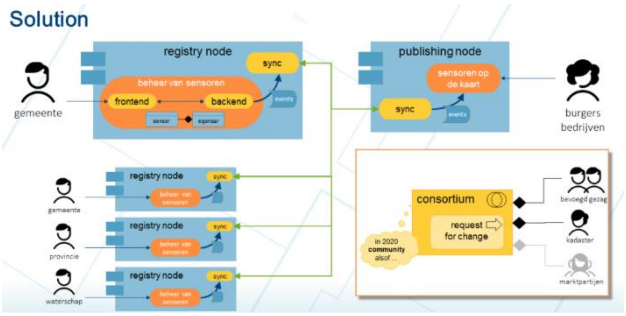

Figure 1. Solution architecture

3. IMPLEMENTATION

The implementation was set up with an open-source software components and shared with the wider community through GitHub - a provider for software development. The Dutch National Sensor Registry Network (SensRNet) takes into account the variations in application and usage of a local Sensor Registry. One municipality might focus on public safety and therefore cameras and video streams while another municipality or authorised supervision might be focussing on environmental monitoring. Those local differences should be taken into account in the system as a whole.

3.1 The organisational implementation

It is quite challenging process to create a consortium with agreed responsibilities and engagement. Especially when the context for which this consortium is formed is in inception phase. So rather than formalizing it, a voluntary community was formed with well-intended partners from a few municipalities, a province and Kadaster. All the members have a common goal - a need to get control about information over the growing number of sensors and make it more transparent to their citizens.

From the early stage of the development of National Sensor Registry there is an increasing number of partners who would like to become a member and are interested to play an active role in the community. The community is growing until today.

Next to the community forming also other organisational activities were taking place. Kadaster was asked to take the role of software development execution. A small developers team enabled the development of a first skeleton of the network architecture, called SensRNet. They have set it up as an open-source project so everyone willing to join could do so.

To gather information about "must haves" requirements and priorities a Functional Advisory Board (FAB) was formed. The FAB became a community driven group consisting of people from different governmental organizations who together decided what features should be build, how user interfaces should look like, who commented on demos and reviewed documentation and data models. It was a great way to involve people from the community and to provide a support base for the development team.

The development team started in April 2020 and delivered the first 'walking skeleton' of the system two months later. From

September the FAB was set up and guided the development team towards a Minimum Viable Product which they delivered in April 2021. Within a year the Sensor Registry Network 'SensRNet' was born, brought to life and developed into a working product which could grow up into the actual National sensor Registry of The Netherlands.

3.2 The architectural implementation

To develop such a system more detailed architectural design was needed. This is based on a few key concepts:

- Decentralisation as given
- Event-driven (and Event-Sourcing internally)
  - Distributed Ledger Technology/Blockchain
- Data at the source (with respect of events as the origin of data)
- Privacy by design – don't share what's not needed to be shared
- Open collaboration

3.2.1 Decentralisation

Decentralised means thinking of it as a network topology, connected participants, collaboration, nodes in a network.

SensRNet is by definition a collaboration with multiple local and central governmental departments and institutes. The subject of sensors has even more potential interactions and collaborations with citizens, commercial companies, sensors themselves. Although a central set up is less complex it will be very hard to adopt to a decentralised set up in later stage. It is easier to think about is from the beginning as a decentralised world with many connections and connected organisations as well as devices.

3.2.2 Events

Events are very important elements in the whole architecture. It is a mix of Event Sourcing and *Event-driven or *Event-streaming architectures. On one hand, these are common patterns on a technical layer (e.g. database backup and synchronization), on the other hand these patterns are gaining more and more popularity with cloud and cloud architectural patterns.

Event Sourcing is an architectural pattern mostly promoted by Domain Driven Design (Evans, 2003). Instead of directly updating a database with the changes at hand, the changes are described as separate data, which are called events. Events describe the actual change in the system including the intent of the user (or requester), containing the data of the change and marking the success of the command of the requester. After the event is being produced and stored in the event store it is 'played' into a projection or view. This is reproducible. This might be executed directly or at a later point in time. This might be executed simultaneously for multiple projections or views. Once started with Event Sourcing this is more common than exceptional.

Another benefit of using events is making the system more open for extension and further development. At one stage the system produces a certain collection of events. If new functionality is required and being developed, probably new events will be produced. Once all consumers of events are ready, the new events can be added with little effort. By versioning the events, the system becomes additive, only appending new events. The data model can then be extended and newer types of events produced, while the old data is kept. The evolution of the events

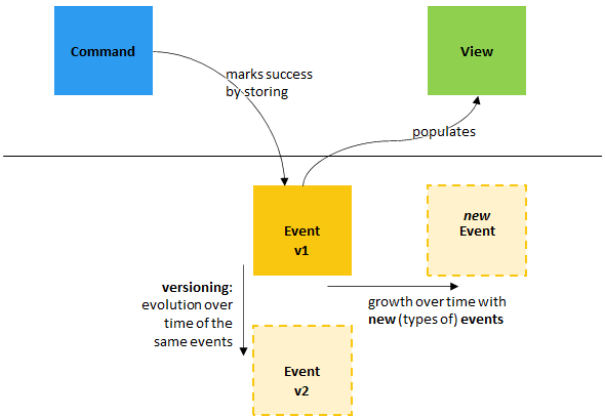

Figure 2. Event versioning in an Event Sourced system

can easily be observed, and it acts as a logbook (a.k.a. audit trail) as well.

Event-driven or Event-Streaming architectures are patterns for distributing events once created over multiple instances or processes. Connecting multiple processes in serial or parallel, triggering next steps after a step has finished, distributing changes through a huge range of containers, machines or data centers. This is not directly related to Event Sourcing, although they mix and complement each other very well. First events are created in an Event Sourced system and after that the events are being distributed through Event-streaming and messaging systems.

3.2.3 Distributed Ledger Technology / Blockchain

Following the key concepts of events and decentralization, it is expected that the system produces events and that there's a need for sharing these events in a decentral network. These could be seen as transactions in a ledger, a distributed ledger. And this is exactly the technology underneath Blockchain (Masood et al, 2018).

Blockchain is a buzzword. But it is not always clear what part of the hype is intended to communicate. Blockchain is a hype, a trend and a disruption of common and known structures. Although this might be true (or become true one day), it is also 'just' a technology for maintaining a shared ledger in a distributed way. The trends which might disrupt the world one day is about the *usage* of the Blockchain technology. In this case for SensRNet we're just in need of the technology of the distributed ledger.

That's exactly what's it used for: a distributed ledger of events being produced by known participants in a private network. Some argue that this is not actually 'a real Blockchain' because it is not an open and one of the mainstream Blockchains (e.g. Bitcoin, Ethereum, IOTA). On the other hand, by using the underlaying technology this could be a small step in the future once the need arises. Or not if this need does never come up.

3.2.4 Data at the source

Within the Dutch government there is a trend towards "data at the source". SensRNet is following this principle. In combination with the key concept of event-driven and superlative Event Sourcing stating that changes to a system are described as Events. Therefore events are the origin of data. We interpret these principles as we respect the source where data is originated and the definition of this genesis data is described as events.

Events are immutable and will never be deleted. Events will be appended to the ever growing collection of events. To analyse a current state (at any point in time) one can simply process all events. This is a repeatable action and so the data at the source is not the eventual state but the events (and only the events). Because of the nature of events copying and distribution of events is still respecting its origin, its source without prohibiting it. On the contrary: events produced at a source might be a trigger for other actions 'somewhere'.

3.2.5 Privacy by design

Privacy by design is an approach to systems engineering that seeks to ensure protection for the privacy of individuals by integrating considerations of privacy issues from the very beginning of the development of products, services, business practices, and physical infrastructures. In SensRNet these principles are applied by:

- take encryption into account for data in transit (TLS / https) and data at rest (encrypted storage) for sensible (user) data,
- not sharing sensible (user and sensor) data outside the boundary where it is initialized.

3.2.6 Open collaboration

From very beginning SensRNet was a collaboration between many partners, open, transparant with possibility to join the community at every moment. Therefore SensRNet is initiated as an open source project. Sources, documentation and issue tracking can be found at GitHub.com

Morover SensRNet applies open standards as much as possible and applicable. Open standards are already validated and defined specifications on how to interact and integrate; already defined collaboration 'rules'.

3.3 The software architecture

Putting this all together the development team started with a few open-source projects. Because this was executed by the team at Kadaster as an innovation project, all sources are hosted at GitHub / Kadaster-labs2. All repositories start with “sensrnet“.

The core of the Registry Node is the backend component. This is an Event-Sourced component producing events. These events are stored in an Event Store. The synchronization component, just sync for short, synchronizes between the Event Store of the backend and the distributed ledger. This synchronization is bi-directional, so events produced in a Registry Node are posted onto the ledger as well as synchronized events from other Registry Nodes are posted to the local Event Store. By doing so, the local Registry Node can ‘see’ the events and therefore the sensors registered from other Registry Nodes. This might be optional or to be filtered depending on the functional requirements decided upon in the FAB.

The backend stores all information as events in the Event Store, even privacy sensitive information. The sync component does not synchronise all events. It filters out these privacy sensitive events and only shares events which are open data and what’s to be published in the Central Viewer of the Publishing Node. By this the Privacy by Design is applied.

The Publishing Node is more or less a clone of the Registry Node, but without functionality to update the distributed ledger of sensors and optimized for query performance. This will be the entry point and main service for all users and usage of the National Sensor Registry.

3.3.1 Data Model & Event Model

The goal of National Sensor Registry is to provide transparency about sensors, or maybe sensing or being sensed. This requires the knowledge about the existence of a sensor and the stream of data it produces, but it is not necessary to encapsulate the sensor data itself as well. On the contrary, the sensor data is explicitly put out of scope of the Sensor Registry. It will only cover the metadata about sensors and sensing but will not contain the sensor data itself, only reference this.

Given the decentral set up of the registry a uniform structure of the data being exchanged is needed. Preferably this would be an open standard or at least based upon one. There are a few relevant and applicable standard available: OGC SensorThingsAPI3, ETSI SmartM2M / SAREF4, OGC Semantic Sensor Network Ontology5. The issue with all of these standards is that they focus on the sensor data primarily and model the metadata secondary. Therefore, none of the standards is fully suitable for the sensor registry. Based on previous research like the pilot of the city of Eindhoven and Kadaster carried out in 2018 and in consultation with other government agencies like RIVM (about air quality), the SensorThingsAPI suited best for the sensor registry (Heide 2017).

While discussing the data model, it became clear that ‘sensor’ and 'sensor’ isn't the same thing. Is the sensor the physical device visible at the lamppost or hanging on the wall? Or is the sensor the actual sensing part within the device? This is addressed in Domain Driven Design as well; first build up a ubiquitous language between all people involved. If there’s a misunderstanding, there's probably a concept missing or a need for different terminology.

In SensRNet the sensor meaning the physical device visible in the public space is defined as the Device. This matches the Thing entity of the SensorThingsAPI and sort of is the metadata describing the ‘physical world’. The actual sensing part within the device, is defined as the Sensor, as it is in the SensorThingsAPI as well. To reference the sensor data the standard has a proper matching entity called Datastream. A data stream is a link to the data or ‘digital world’, as data is generally posted to and made available through a digital platform. So, the sensor registry data model is the ‘registrative world’ connecting the ‘physical world’ and the ‘digital world’. The last thing missing is the connection to the ‘governing world’: What is the legal ground on which sensing is allowed and under which restrictions or regulations? Our data model accommodates and connects these separate ‘worlds’.

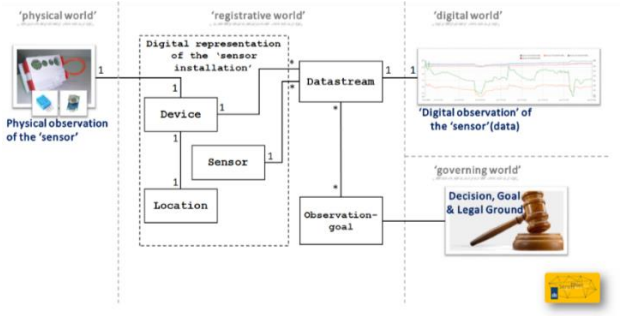

Figure 3. Conceptual Data Model

Conceptually the Data Model connects these different worlds and still follows the SensorThingsAPI as much as possible.

In (Figure 4) the matching entities are shown between the SensRNet Data Model and the SensorThingsAPI.

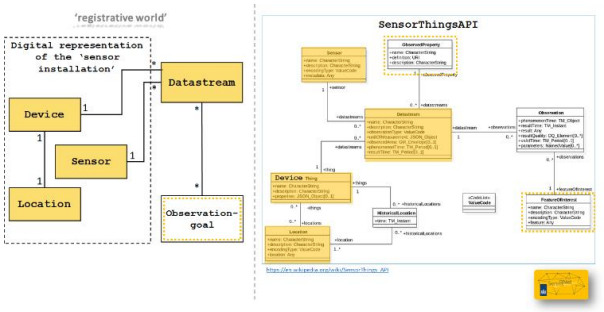

Figure 4. SensRNet Data Model matching SensorThingsAPI

The Data Model contains the concept of 'Aggregate', sometimes called Aggregate Root. This is a concept from Domain Driven Design (Evans, 2003). When applying Event Sourcing this is a mandatory concept as the boundaries of integrity; commands are validated on an Aggregate and events are produced by an Aggregate (only). The Data Model Aggregates form the boundaries and structure to design the Event Model. The aggregation of Device, Sensor and Datastream is the Sensor Device Aggregate. Any organisation either being a governmental organisation or (soon) a private organisation is modelled as a Legal Entity. With the roles and contact details this is also the Legal Entity Aggregate. Together with the Observation Goal Aggregate and User Aggregate the model is complete (although this might be extended in the future).

Events are being produced by Aggregates and are containing the data of the change as well as the intend of the change. For the sensor registry there are no complex processes involved (or known there are) so the intend is quite data entry like. Still there are a few nuances in the intent. In the next figures the Event Model is presented.

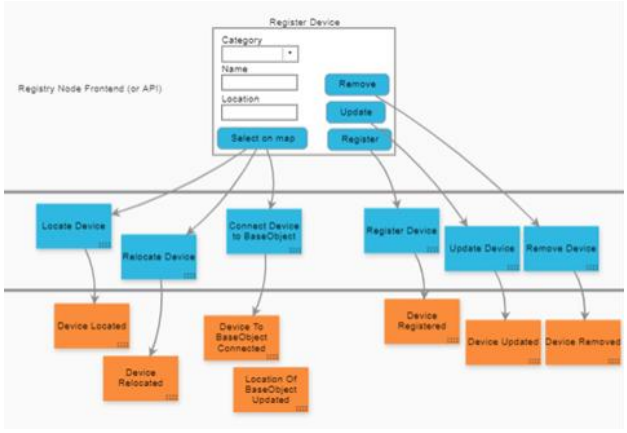

Figure 5. Eventmodeling of Device

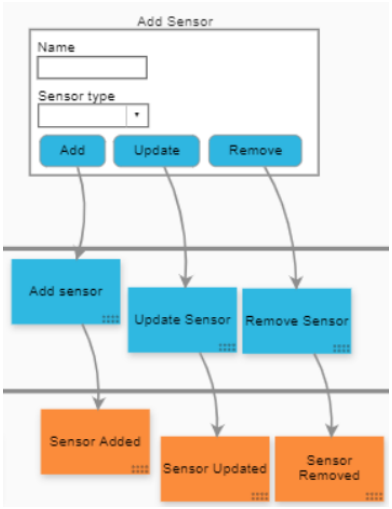

Figure 6. Eventmodeling of Sensor

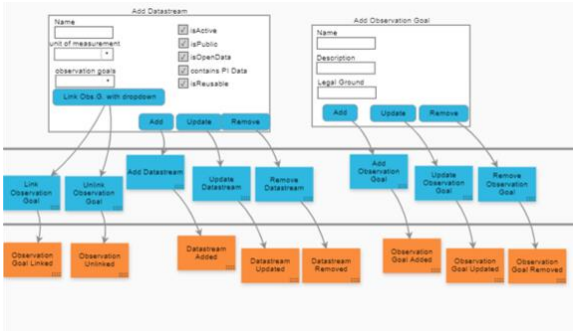

Figure 7. Eventmodeling of Datastream and Observation Goal

3.4 The technical implementation

This section details how this all comes together on a technical level and discusses some of the technical decisions to highlight how the sensor network runs in practice.

3.4.1 Registry Node

The portal where sensor metadata can be registered is the Registry Node. It is a webpage with separate API. This separation of front- and backend is made so that it is possible for interested parties to create their own webpage or integrate with the API. Extending our platform is therefore possible, for example if some local government wants to first validate and approve a new sensor entry before publishing it on the network. The default webpage is built on top of Angular6, and the API on NestJS7. Both are written in TypeScript8. NestJS offers support for CQRS (but not Event-Sourcing), making it a natural fit. User management is decoupled from these components. Each organization which runs a Registry Node can make use of their existing user management system and provide authorizations using that, only allowing the right people to manage the sensor metadata.

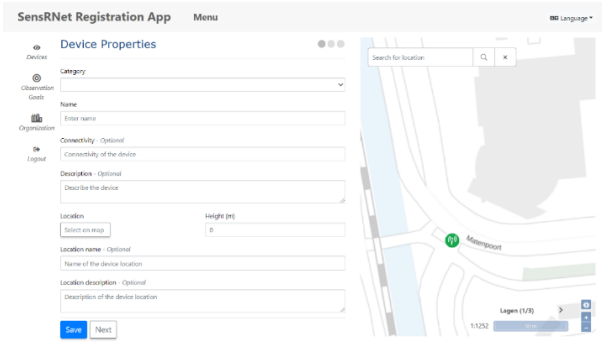

Figure 8. A screenshot of the first page of the registration form in the Registry Node

3.4.2 Storage & sharing events

New sensor metadata and mutation are modelled by events, as discussed in section 2.4.3. For storing these events EventStoreDB9 was chosen. It is specifically designed to efficiently store many events. The database for storing the projection on these events is a document store: MongoDB10 acts as database for storing the event projections or views and other private data.

While a registry nodes functions without sharing its events, the full potential of a distributed ledger can only be obtained by sharing with the other participants. MultiChain11 (Greenspan, 2015) was chosen for this purpose, as it has great support for streams, which align well with the Event-driven key concept, distributing and sharing all events with all nodes in the network. It is like Blockchain but is private by design. Permission must be granted to new nodes connecting to the network, preventing unwanted access to the network. MultiChain also works with block mining (D4.3 The COMPOSITION Blockchain, 2019) but is much more environmentally friendly as mining is done through delegation instead of Proof of Work. Smart Contracts can be added the chain to make sure SensRNet participants can only make changes to sensors they themselves registered.

The sync component, also written in NestJS, is the linking pin between the Registry Node and MultiChain and makes sure new events in the EventStoreDB are published on MultiChain and vice versa.

3.4.3 Deployment

To make it easy for local governments to run a copy of the code, the binaries are made publicly available as Docker images (Open Container Initiative, 2015). This way the code is ready to be run anywhere, without the need of compiling it first. The images can be deployed with any container orchestration tool. Deployment files (Helm Charts12) for Kubernetes13 are provided on GitHub, as this is the de facto standard for running containers in either cloud or on-premise environments. Logging and monitoring of the components can be done using normal Kubernetes procedures. Implementation and adoption should be as easy as possible because of this, to make the technical side as transparent as possible.

4. EXPERIENCES&CHALLENGES

Currently, June 2021, the Minimum Viable Product has been released and published for installation. This means that municipalities of The Netherlands are able to install and run a Registry Node, connect to the network and start registering their sensors into the National Sensor Registry. The components are available, the system and network is tested and ready for pilot stage application. Within many municipalities the transition towards more cloud-based infrastructure is still starting or in some stage of early implementation. SensRNet has been targeted on cloud infrastructure and many municipalities don't have a running Kubernetes platform available straight away. Still, this is the to be and desired situation in the near future and a stable choice although this is not helping to start and scale SensRNet.

All municipalities are collaborating in the Foundation of Dutch Municipalities 'VNG' and within this foundation a cloud agnostic standard is being developed, called Haven15. This states that future municipality infrastructure should provide a standardized and managed Kubernetes platform in a secure way. In collaboration with the foundation the SensRNet components are compliant with this standard and proven to be installable and usable on such an environment. The city of Tilburg is the first city with a Haven compliant cluster available and where the SensRNet Registry Node components were installed. The experiences with Kubernetes and the Haven standard including the publishing tools like DockerHub and ArtifactHub are great, smooth, easy and complex. There's a lot of configurations possible and a lot of integration must be connected in the right way. On the other hand, everything is scripted so setting up an environment and deploying components is very repeatable and reproducible. This makes scaling up very easy!

This is the technical side of things. But after installation and making a Registry Node available within a municipality the organisational side of using it must be implemented. This is another hurdle in the introduction and scaling of the sensor registry. Although many municipalities are one way or the other involved in sensing and the asset management of sensors many times, this is not yet properly organised, and sensors registered. Even broader local legislations are still in development and differ between municipalities. Not to mention general laws and legislations for the institution of the nation sensor registry, the community, some form of consortium and executive agencies. For this a Governance Initiation Group is formed. One of the tracks currently starting is the Pilot Group. This is the group of municipalities implementing the maintenance of sensors in organisation and registry using the Registry Node software.

5. FUTURE PRODUCT

The future National Sensor Registry product is intended to be owned by consortium. The consortium group is aimed to include representants of governmental organisations, business world and other user/target groups. The group will decide about further development, features, partnerships in the consortium and partnerships with outside collaborators on sensors, sensor data and all kind of application of the registration of sensors for various goals.

The SensRNet will form a common national and uniform product where other local sensor registries could be linked to. The differences between sensor registries in the cities will be fully respected. The SensRNet will concentrate on joining the data and translate it into a uniform and nationwide product. The end user will get an overview of national, uniform and complete viewer of the registered sensors (and other measuring devices), their location, the reason why there are placed and access to their owner and a link to the produced data (if not restricted by security or privacy).

National registration of sensors is aimed at providing a necessary base-layer underneath many contexts like privacy, health, infrastructure etc. The law has already obliged municipalities to publish the sensors that bring risk for the people's privacy. Having all kinds of sensors registered, it would be possible for example to filter the group of sensors that process privacy sensitive information. In many situations it is not a single sensor but the combination of many. This indicates that a 'privacy layer' forms a context layer on top of the base-layer of registered sensors.

A Smart City does not become smart by just having registered sensors in the public space. The Sensor Registry as a base-layer can be a perfect tool to support Smart Cities. However, to make it successful we need (smart) people who understand the need of registry, will want to implement it, know how to use it and benefit from it.

REFERENCES

Dahan, U. (2009, December 9th). Clarified CQRS. Retrieved from The Software Simplist - Enterprise Development Expert & SOA Specialist: https://udidahan.com/2009/12/09/clarified-cqrs/

Evans, E. (2003). Domain-Driven Design: Tackling Complexity in the Heart of Software., Pearson Education (Us).

Heide, J., Grus, M., Nouwens, J (2017) Making Sense for Society.

Stopford, B. (2018). Designing Event-Driven Systems - Concepts and Patterns for Streaming Services with Apache Kafka, 1005 Gravenstein Highway North, Sebastopol, CA: O'Reilly Media, Inc.

Public versus Private Blockchain, https://medium.com/coinmonks/public-vs-private-blockchain-in-a-nutshell-c9fe284fa39f

Dr Gideon Greenspan (2015), Multichain White Paper, https://www.multichain.com/download/MultiChain-White-Paper.pdf

F. Masood, A. R. Faridi (2018), An Overview of Distributed Ledger Technology and its Applications, https://www.researchgate.net/publication/330139945_An_Overview_of_Distributed_Ledger_Technology_and_its_Applications

D4.3 The COMPOSITION Blockchain (2019), https://portal.effra.eu/result/show/3032

Open Container Initiative (2015), https://opencontainers.org/