

SMART Sensordata Infrastructuur

Rapport pilot SMART Sensordata Infrastructuur (SSI)

15 december 2022, Den Bosch

HAS
green
academy

Voorwoord

Voor u ligt het rapport van de pilot SMART Sensordata Infrastructuur (SSI). Deze pilot is uitgevoerd door docenten en studenten van de opleiding AGIS van de HAS green academy in de periode van juni t/m december 2022 in samenwerking met en met financiële steun van het DCC voor Praktijkgericht onderzoek van SURF. Dit rapport bevat de volgende op te leveren resultaten:

1. Ontwerp en praktische beschrijving van algemeen toepasbare datadriven-workflow voor sensordata
2. Ontwerp en praktische beschrijving van metadata-model van sensor-data, gericht op datadefinitie en datakwaliteit

Inhoudsopgave

1. Introductie	pagina 1
2. Sensordata infrastructuur	Pagina 2
2.1. Conceptueel model	Pagina 2
2.2. Technische implementatie	Pagina 3
2.2.1. The Things Network	Pagina 4
2.2.2. Permanente database opslag	Pagina 6
3. Sensor metadatamodel	Pagina 9
4. Conclusies en aanbevelingen	Pagina 10
Bronnen	Pagina 12
Appendix 1	Pagina 13
Appendix 2	Pagina 14

1. Introductie

Minimaal 3 lectoraten op HAS green academy maken gebruik van sensordata. Zij lopen in hun onderzoek regelmatig tegen het probleem aan dat de data-infrastructuur onvoldoende mogelijkheden biedt om de real-time sensordata uit de verschillende fieldslabs efficiënt en effectief op te slaan en in onderlinge samenhang breed toepasbaar te krijgen voor het lectoraatsonderzoek. Hierdoor wordt sensordata niet optimaal ingezet en gaat veel tijd verloren met het uitrollen en operationaliseren van sensoren die niet besteed kan worden aan data-analyse en visualisatie. Omdat het aantal fieldslabs met sensoren voor nieuwe onderzoeksprojecten in de toekomst verder gaat toenemen is het noodzakelijk het geschetste probleem nu op te gaan lossen.

Het ultieme doel zoals geformuleerd in het projectvoorstel (Jellema, 2022) is het opstellen van een sensordata infrastructuur om de via fieldslabs verkregen ruwe sensor-data gestandaardiseerd op basis van een eenduidig metadata-model en volledig automatisch via een datadriven-workflow, die alle fasen uit het datascience-lifecycle omvat, direct toepasbaar te krijgen voor de verschillende onderzoeksvragen.

Om dit doel te bereiken zijn de volgende activiteiten uitgevoerd:

1. Interviews met experts van verschillende hogescholen en SURF. Een volledige lijst van geïnterviewden met een korte uitwerking van de interviews is te vinden in appendix 2.
2. Ontwerpen van sensordata infrastructuur en metadatamodel op basis van opgehaalde kennis en ervaringen uit interviews.
3. Valideren van opstelde sensordata infrastructuur en metadatamodel met experts van SURF.
4. Proof-of-Concept in de vorm van een fieldlab om ontworpen sensordata infrastructuur in de praktijk te valideren.

In hoofdstuk 2 worden ontworpen sensordata infrastructuur en workflow beschreven. Hierbij is onderscheid gemaakt tussen een beschrijving op conceptueel niveau en de beschrijving van de technische implementatie. In hoofdstuk 3 wordt het metadata model beschreven. In hoofdstuk 4 volgen de conclusies en aanbevelingen.

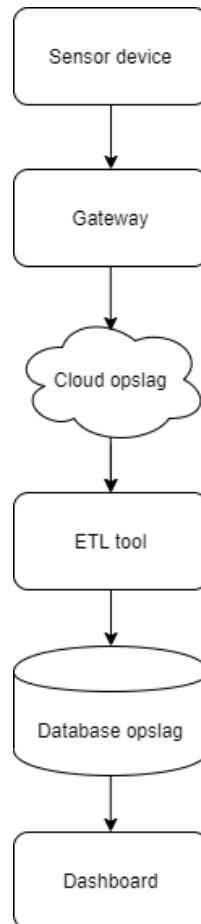
In een proof-of-concept is de beschreven sensordata infrastructuur geïmplementeerd en in de praktijk gevalideerd. Voor het uitrollen en operationaliseren is een handleiding met stappenplan en frequently asked questions (FAQ's) document opgesteld. Deze handleiding wordt geleverd in een separaat document.

Dit rapport is bedoeld voor onderzoekers die in fieldslabs met sensoren data willen gaan inwinnen en op een eenvoudige manier toegankelijk willen maken voor onderzoek en daarom een overzicht willen hebben van de verschillende componenten waaruit een sensordata infrastructuur opgebouwd is en welke metadata vastgelegd moet worden. Dit zijn over het algemeen onderzoekers met beperkte ICT-kennis. Daarom worden de belangrijkste ICT-begrippen in appendix 1 nader toegelicht.

2. Sensordata Infrastructuur

2.1. Conceptueel model

In figuur 1 is het conceptuele model geschetst van de voorgestelde sensordata infrastructuur.



Figuur 1: Conceptuele sensordata infrastructuur

De individuele componenten en communicatie tussen de verschillende componenten worden hieronder verder uitgewerkt.

Het sensor device wordt in een fieldlab geïnstalleerd om met een bepaald tijdsinterval (bijv. elke seconde, elke minuut of elk uur) data te verzamelen. Hierbij kan gedacht worden aan meteorologische data (bijv. temperatuur, luchtvochtigheid of regenval), bodemdata (bijv. bodemvocht of EC-waarde), locatiedata (bijv. locatie van mensen of dieren), stoffen in de lucht (bijv. fijnstof, stikstof of CO₂).

Een sensor device bestaat naast de sensor die de meting doet, ook uit een antenne om de ingewonnen data draadloos naar een gateway te sturen en een batterij voor de stroomvoorziening.

Als het sensor device een meting verstuurd is, wordt deze door de dichtstbijzijnde gateway opgevangen. Deze gateway stuurt de meting via het internet door naar een centrale cloud opslag. Er zijn verschillende protocollen om de metingen naar een gateway te versturen.

In de centrale cloud opslag worden alle door de gateways verstuurde metingen opgeslagen en via verschillende API's ontsloten voor verder gebruik. Hier vindt ook het beheer van de sensor devices en de decodering van de metingen plaats. De metingen worden voor een beperkte tijd bewaard in de cloud opslag.

Met een ETL-tool kunnen de metingen via een API uit de cloud opslag gehaald worden en opgeslagen worden in een database voor permanente opslag. In deze database moet ook de metadata van de sensoren opgeslagen worden zodat altijd achterhaald kan worden met welke sensor een meting gedaan is, wat gemeten is en in welke eenheid gemeten is.

Vanuit de database voor permanente opslag kunnen de metingen ontsloten voor onderzoekers voor verdere analyse en visualisatie. Hierbij kan gebruik gemaakt worden van bestaande tools om dashboards te bouwen of kunnen de metingen via een API beschikbaar gesteld worden voor verder gebruik of download.

2.2. Technische implementatie

Bij de keuze voor softwarecomponenten voor de technische implementatie is zo veel mogelijk gekozen voor gratis open source producten en open standaarden om de implementatie kosten zo laag mogelijk te houden en een vendor lockin te voorkomen.

2.2.1. The Things Network

Op basis de interviews en de gesprekken met de experts van verschillende hogescholen en SURF is besloten om de technische implementatie van het conceptuele model uit te werken op basis van het netwerk van The Things Network (TTN). The Things Network is een wereldwijd IoT ecosysteem dat bestaat uit sensor devices, gateways en een centrale cloud opslag (<https://www.thethingsnetwork.org/>). Er is voor dit netwerk gekozen om de volgende redenen:

- Een aantal hogescholen in Nederland maakt al gebruik van TTN in hun sensordata infrastructuur.
- TTN maakt gebruik van het LoRaWAN protocol (<https://lora-alliance.org/>). Dit protocol is geschikt voor langeafstandscommunicatie met weinig vermogen. In Europa maakt dit protocol gebruik van de 868 MHz frequentie. LoRaWAN wordt veel gebruikt voor IoT oplossingen.
- Voor onderwijsinstellingen zijn er tot op heden geen kosten aan verbonden aan het gebruik van de TTN-infrastructuur.
- TTN heeft een wereldwijd netwerk met een redelijk goede dekking in Nederland (zie <https://ttnmapper.org/heatmap/>)
- TTN is open source ontwikkeld en onderhouden door een uitgebreide gemeenschap van ontwikkelaars

Een uitgebreide beschrijving van de TTN LoRaWAN architectuur is hier te vinden: <https://www.thethingsnetwork.org/docs/lorawan/architecture/>.

Alle sensor devices die het LoRaWAN protocol ondersteunen met een 868 MHz frequentie zijn geschikt om te koppelen aan het TTN-netwerk.

Het TTN-netwerk biedt een netwerk van gateways om de sensormetingen op te vangen en deze via internet (TCP/IP) door te sturen naar een centrale server waar de TTN cloud draait (The Things Stack). De TTN cloud dient voor de tijdelijke opslag van de binnengekomen metingen. In TTN cloud vindt ook de decoding plaats van het binaire bericht naar een leesbaar JSON-formaat.

Met elke meting wordt ID van het sensor device dat de meting gedaan heeft, meegestuurd. In de TTN cloud is elk device met zijn unieke ID geregistreerd zodat de binnekomende metingen aan de juiste device gekoppeld kunnen worden op basis van device ID. De devices zijn gekoppeld aan applicaties die weer gekoppeld zijn aan organisaties. Zie voor uitleg over het registreren van een sensor device bij het TTN-netwerk en het decoderen van de berichten de uitgebreide handleiding.

Bij installeren van sensor is het van belang om te weten of er op de locatie voldoende dekking van één of meerdere gateways is. Als er geen dekking is, kunnen de berichten niet opgevangen en doorgezonden worden naar de TTN cloud. Bij onvoldoende dekking zullen niet alle berichten opgevangen en doorgestuurd worden.

Om een inschatting te kunnen maken of er op een locatie voldoende dekking is, kan je gebruik maken van de TTNMapper website (<https://ttnmapper.org/heatmap/>). Deze website geeft een overzicht van de locaties van de gateways. Als je een gateway selecteert en dan beams aanvinkt, krijg je een overzicht van de locatie en de sterkte van de ontvangen signalen door een gateway. Hoe sterker het signaal hoe beter het ontvangen is door de gateway. Op basis van deze kaart kan je een inschatting maken over ontvangst is of niet (zie figuur 2). In dat laatste geval moet er ook een gateway geplaatst worden om te garanderen dat de metingen opgevangen en doorgestuurd worden.



Figuur 2 Gateway met sterkte van ontvangen sensor signaal

2.2.2. Permanente database opslag

Omdat de metingen slechts beperkte tijd (5 dagen) in de TTN-cloud bewaard worden, moeten deze uit de TTN-cloud worden opgehaald voor permanente opslag in een permanente database. Voor het ophalen van metingen uit de TTN-cloud wordt een MQTT API aangeboden.

Om de implementatie eenvoudig te maken biedt TTN ook een aantal SDK's om met de MQTT API te communiceren. Er zijn SDK's voor o.a. Node-RED, Java en Python. Met deze SDK's kan een ETL-script ontwikkeld worden. Het te ontwikkelen ETL-script bestaat uit de volgende stappen:

1. Extract: Ophalen van meting uit TTN cloud
2. Transform: Uitpakken en verwerken van meting
3. Load: Toevoegen meting aan database

TTN biedt geen platform voor het draaien van ETL. Hiervoor zal in de cloud (bijv. op Azure of AWS) of on premise een server ingericht moeten worden waarop de ETL-tool (bijv. Node-RED) of een ETL-script (bijv. Python) kan draaien.

Het ontwikkelen van een ETL-script is maatwerk en hangt af van de database die gekozen is voor de permanente opslag van de metingen. In de technische implementatie die hier voorgesteld wordt, is gekozen voor Node-RED als ETL-tool. Node-RED is een open source tool voor het ontwikkelen van ETL-scripts die veel gebruikt wordt in IoT projecten. In de handleiding wordt uitgelegd hoe je op Node-RED een ETL-script ontwikkelt om via MQTT de metingen uit de TTN cloud op te halen en in een database op te slaan.

Naast de metingen moet in de database ook de metadata van de sensoren opgeslagen worden. In het volgende hoofdstuk wordt de metadata beschreven.

Voor de permanente opslag zijn er verschillende mogelijkheden. In dit project is gekeken naar opslag in een SQL-database en opslag in een NoSQL timeseries database:

Per type database worden in de volgende tabel de voor- en nadelen gegeven:

	Voordelen	Nadelen
SQL database	<ul style="list-style-type: none">• Gestandaardiseerde datamodellen voor opslag sensordata beschikbaar• Genormaliseerde niet-redundante opslag• Veel kennis aanwezig• Goede, open source RDBMS-software beschikbaar• Veel softwarepakketten kunnen communiceren met SQL-databases	<ul style="list-style-type: none">• Niet exclusief ontworpen voor opslag timeseries• Sensor metadata moet opgeslagen zijn voordat metingen opgeslagen kunnen worden
NoSQL timeseries database	<ul style="list-style-type: none">• Speciaal ontworpen voor opslag timeseries• Geen noodzaak tot opslaan metadata voordat metingen opgeslagen kunnen worden• Uitstekende integratie met softwarepakketten voor visualisatie van timeseries	<ul style="list-style-type: none">• Redundante opslag• Minder kennis aanwezig• Minder softwarepakketten kunnen communiceren met NoSQL databases

In de voorgestelde technische implementatie is gekozen voor PostgreSQL als SQL-database met de PostGIS extensie voor de opslag van de locatie. Er is voor deze PostgreSQL gekozen om de volgende redenen:

- Open Source
- Geen licentiekosten
- Veel kennis voorhanden
- Eenvoudig te implementeren en te koppelen met ETL-tools en dashboards

De PostgreSQL database zal net als de ETL-tooling in de cloud (bijv. op Azure of AWS) of op een on-premise server moeten draaien.

Alle tools die via SQL queries gegevens uit een PostgreSQL database kunnen selecteren, kunnen gebruikt worden voor visualisatie en analyse. Denk hierbij aan tools zoals PowerBI van Microsoft (<https://powerbi.microsoft.com/>) of de open source dashboard tool Grafana (<https://grafana.com/>) maar ook aan scripttalen zoals Python en R.

In een vervolgproject zal opslag in een NoSQL database verder uitgewerkt worden.

3. Sensor Metadatamodel

Het opslaan van sensor metadata is essentieel voor het gebruiken en delen van sensordata. Zonder goede metadata is er een grote kans dat de sensordata verkeerd geïnterpreteerd of gebruikt wordt.

De toegevoegde waarde van het metadatamodel is dat onderzoekers direct inzicht hebben in de sensor metadata. Goede metadata garandeert een bredere inzetbaarheid en efficiënter en effectiever gebruik van de sensordata. Zonder metadata is data validatie en het inschatten van datakwaliteit niet mogelijk. Ook is moeilijk na te gaan of sensordata bruikbaar is voor het beoogde doel.

Bij het opstellen van het metadatamodel is gebruik gemaakt van de volgende documenten:

- OGC SensorThings API: <https://www.ogc.org/standards/sensorthings>
- Sensor Model Language: <https://www.ogc.org/standards/sensorml>
- INSPIRE Sensor Web standards: <https://inspire.ec.europa.eu/id/document/tg/d2.9-o%26m-swe>

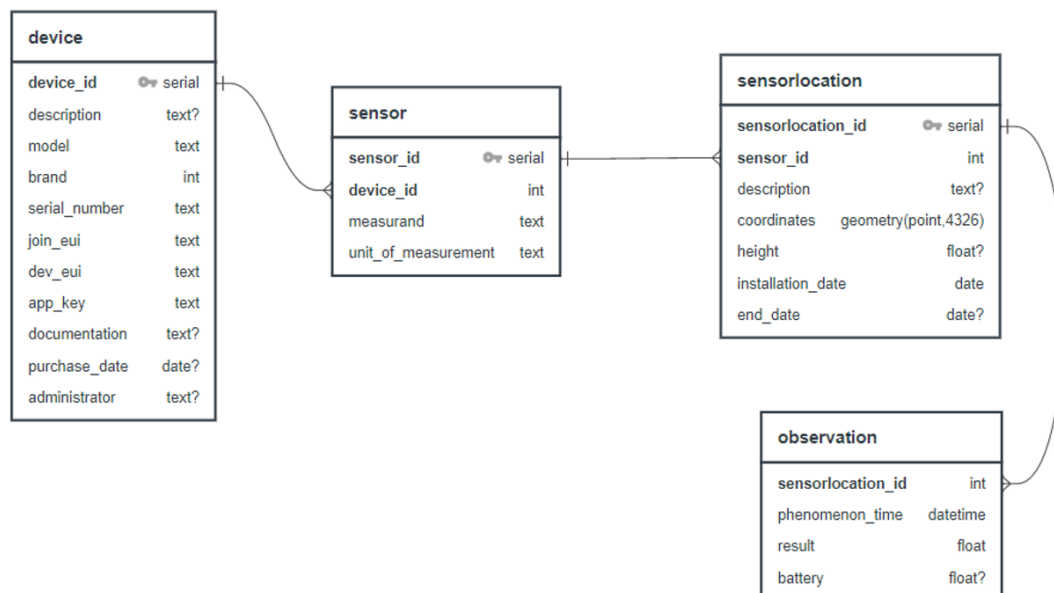
De in deze documenten beschreven metadatamodelen zijn vereenvoudigd zodat alleen de essentiële metadata opgeslagen moet worden en vertaald naar entiteiten die vervolgens als tabellen geïmplementeerd zijn in een SQL-database.

Het metadatamodel is opgebouwd uit de volgende entiteiten (zie figuur 3):

- Device: Het geheel van sensoren, accu en antenne dat door de leverancier als één product geleverd is (het gehele meetsysteem). Een device kan meerdere sensoren bevatten.
- Sensor: Het element dat een natuurkundige grootte meet (bijv. een temperatuursensor of fijnstofsensor).
- SensorLocation: De locatie en de tijdsperiode dat een sensor op een bepaalde locatie metingen verricht.

De sensormetingen worden opgeslagen in de volgende entiteit:

- Observation: Meting die door de sensor op een bepaald tijdstip verricht is.

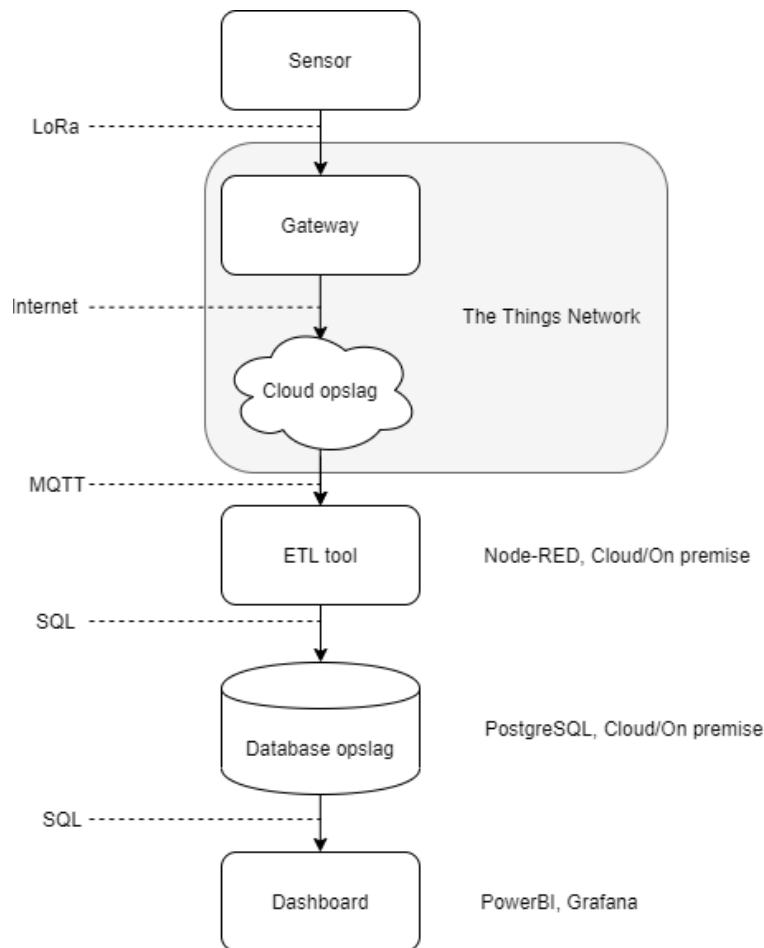


Figuur 3 Sensor metadata ERD

Het toevoegen van een device met de sensoren en de locatie aan de database moet eenmalig gebeuren voordat de sensor uitgerold gaat worden. Hiervoor is een SQL-script ontwikkeld. De sensor observaties worden op basis van het *sensorlocation_id* gekoppeld aan de metadata. Dit *sensorlocation_id* moet in het ETL-script dat de observatie toevoegt aan de tabel observation eenmalig gedefinieerd worden.

4. Conclusies en aanbevelingen

Op basis van gesprekken met sensor experts van verschillende hogescholen en SURF is een sensordata infrastructuur opgezet met open source componenten en open standaarden. De opgezette sensordata infrastructuur is weergegeven in onderstaande figuur.



Figuur 4 Sensordata infrastructuur met gebruikte hardware, software en protocollen

Om de sensordata infrastructuur zoals weergegeven in Figuur 4 in de praktijk te toetsen is een proof-of-concept opgezet. In de proof-of-concept zijn een aantal LoRa temperatuur- en bodemvochtsensoren geïnstalleerd op het groene dak van de HAS green academy. Met deze sensoren wordt realtime om de 5 minuten de temperatuur en het bodemvocht in en op het groene dak gemeten om uiteindelijk het klimaatrendement te kunnen bepalen.

Op basis van het uitgevoerde onderzoek en de proof-of-concept is het volgende geleerd en kunnen de volgende conclusies getrokken worden:

- Het is mogelijk om met open source producten en open standaarden een sensordata infrastructuur op te zetten waarmee het gehele proces van data inwinning met sensoren tot permanente database opslag en gebruik in visualisaties en analyses geïmplementeerd is
- Het netwerk van The Things Network (TTN) wordt veel gebruikt voor het opzetten van sensordata infrastructuren in een HBO/academische setting

- The Things Network biedt verschillende open standaarden om de sensordata met ETL-tools op te halen
- Voor permanente opslag kan gekozen worden voor een relationele SQL-database maar er zijn ook andere NoSQL alternatieven speciaal ontwikkeld voor de opslag van timeseries (bijv. InfluxDB).
- Metadata is essentieel voor delen en verder gebruik van de ingewonnen sensordata. Bestaande metadatamodellen zijn vereenvoudigd zodat alleen de essentiële metadata opgeslagen wordt. Het metadatamodel is generiek opgezet zodat nieuwe sensoren toegevoegd kunnen worden zonder dat het model aangepast moet worden en het eenduidig toepasbaar is.
- Nieuwe sensoren kunnen eenvoudig opgenomen worden in de sensordata infrastructuur door middel van registratie in het netwerk en het toevoegen aan een gestandaardiseerde ETL-procedure.

Aandachtspunten voor sensorprojecten in het algemeen zijn:

- Zorg dat er voldoende budget is voor de aanschaf van sensoren om een fieldlab in te richten. Een eenvoudige LoRaWAN sensor om temperatuur te meten inclusief batterij en antenne kost al snel 50 euro.
- Let op dat je sensoren met de juiste frequentie aanschaft (868MHz voor Europa)
- Houd er rekening mee dat The Things Network niet overal in Nederland goede dekking heeft. Als er geen dekking is, zal er naast een sensor ook een gateway aangeschaft en geplaatst moeten worden.
- Naast TTN zijn er alternatieve gesloten commerciële netwerken die wel 100% dekking bieden in Nederland (bijv. het KPN LoRa netwerk).
- Maak zo veel mogelijk gebruik van open standaarden om vendor lockin te voorkomen en zorg dat je eigenaar van je data blijft.
- Maak zo veel mogelijk gebruik van bestaande software en clouddiensten. De kosten hiervoor wegen in het algemeen niet op tegen de kosten voor het zelf ontwikkelen van software of het zelf beheren van servers
- Besteed voldoende aandacht aan permanente opslag van ingewonnen data, analyse en visualisatie zodat gebruik van data in de toekomst ook geborgd is.

Bronnen

Jellema (2022), SMART Sensordata Infrastructuur, Voorstel voor pilot DCC voor Praktijkgericht onderzoek

Appendix 1

API: Een application programming interface (API) is een verzameling definities op basis waarvan een computerprogramma kan communiceren met een ander programma of onderdeel (meestal in de vorm van bibliotheken).

ETL: Dit is de afkorting van Extract-Transform-Load. Dit zijn de processen die gebruikt worden om gegevens uit verschillende gestructureerde databases en databestanden te verenigen in een andere database.

IoT: Dit is de afkorting van Internet-of-Things. Dit is het geheel aan apparaten dat via internetverbindingen met elkaar en/of systemen in contact staan en daarmee gegevens uitwisselen.

JSON: Dit is de afkorting van JavaScript Object Notation en is een gestandaardiseerd gegevensformaat. JSON maakt gebruik van voor de mens leesbare tekst in de vorm van data-objecten die bestaan uit een of meer attributen met bijbehorende waarden. Het wordt hoofdzakelijk gebruikt voor uitwisseling van data.

MQTT: Message Queuing Telemetry Transport (MQTT) is een lichtgewicht netwerkprotocol voor het verzenden van berichten. Het is gebaseerd op het "publiceer-abonneer" concept. Als afnemer kan je je abonneren op een queue. Wanneer er een bericht gepubliceerd wordt op deze queue word je geïnformeerd en kan je dit bericht verwerken.

NoSQL database: Een NoSQL database is een database zonder vast database schema ontwikkeld voor snelle efficiënte opslag en retrieval.

Python: Python is een scripttaal die gebruikt kan worden om repeterende processen te automatiseren. Ook wordt Python veel gebruikt in data science projecten voor de analyse en visualisatie van onderzoeksdata.

R: R is een softwarepakket en programmeertaal ontwikkeld voor statistiek en data-analysedoelinden.

SDK: De afkorting SDK staat voor de software development kit en is een verzameling hulpmiddelen die handig is bij het ontwikkelen van software.

SQL: SQL is de afkorting van Structured Query Language en is een gestandaardiseerde taal om gegevens in een relationele database op te slaan, te bevragen, te koppelen en aan te passen.

SQL-database: Een SQL database is een relationele database die met SQL gemaakt en beheerd kan worden.

Appendix 2

Samenvatting interview Leo Remijn (Hogeschool Rotterdam)

Bij de Hogeschool Rotterdam worden sensoren ingezet bij verschillende projecten. Vaak zijn dit projecten waar simpele dingen worden gemeten zoals temperatuur met één cijfer achter de komma. Echter zijn er ook enkele grotere projecten waar HR aan mee werkt. Een voorbeeld hiervan is de groene poort. Hier worden watermetingen gedaan met sensoren die aan een boei vast worden gemaakt. Deze sensoren meten verschillende waterwaardes zoals troebelheid en aangroei van algen.

De sensoren maken gebruik van het IoT netwerk en werken via Lorawan. De HR heeft drie antennes in Rotterdam staan en deze worden naa het IoT gestuurd. Daarnaast is het mogelijk een eigen server te runnen via een Things Network indoor Gateway.

Door de sensoren wordt er naast de metingen van bijvoorbeeld temperatuur en waterwaardes ook metadata opgeslagen. De data kan via een toolbox worden gevisualiseerd. Generieke data zoals GPS, datum van aanschaf, typenummer, tijd (frequentie) en coördinaten van de locatie van de sensor worden hierin meegenomen.

Samenvatting interview Michel Grothe (Geonovum) en Roderick Peters (HAN)

Dit project is gestart in Lent met als hoofdvraag: Hebben mensen die in de buurt van de N325 wonen meer last van luchtvervuiling dan mensen die verder weg wonen? Hierbij hangen de mensen thuis zelf een sensor op om stikstof en fijnstof te meten. Deze sensoren zijn zelf gemaakt, omdat kant en klare sensoren vaak beperkt zijn. In het kader van fijnstof heb je vaak last van de luchtvochtigheid waar kant en klare sensoren niet altijd geschikt voor zijn.

Momenteel zijn er 10 sensoren op 10 verschillende plekken. Eén hang in Nijmegen bij het RIVM, die zorgt voor de kalibratie, de andere negen hangen in Lent. Doordat de mensen zelf de sensor mogen ontwikkelen is er voor het netwerk een eigen keuze uit Lora of een eigen LTEM-netwerk. Dit ligt aan de frequentie waarop de sensoren data sturen. Als de sensoren per minuut zenden dan is een eigen netwerk nodig. Voor de data wordt gehouden aan het MQTT-protocol. Verder maakt Lentse Luchten gebruik van de SensorThings API en Raspberry Pi.

De data komt uiteindelijk terecht in drie verschillende databases. Deze data wordt gevisualiseerd in grafieken. In het kader van de AVG wet is de metadata die opgeslagen wordt beperkt. De exacte locatie van de sensoren wordt niet gegeven. Verder wordt alleen het sensornummer en de hoeveelheid stikstof/fijnstof opgeslagen.

Samenvatting interview Andries van Dongen (Avans)

Afgelopen semester is er op de Avans een project geweest van smart farming met Lora sensoren in het veld. Hier werd bodemvocht, luchtvochtigheid, temperatuur en licht gemeten. Deze sensoren zijn door de Avans studenten zelf gemaakt. De grootste uitdaging waar ze tegenaan liepen was de infrastructuur. Er moest een oplossing komen om de data vanaf het veld in een cloud-omgeving te krijgen. Uiteindelijk is hiervoor een full-tech applicatie gemaakt door de studenten.

De data die wordt verzameld gaat via Lora naar de cloud. Deze worden opgeslagen in een clouddatabase aan de hand van verschillende API's. De data wordt verzonden aan de hand van het MQTT-protocol. Veel metadata wordt niet opgeslagen. Alleen bij welke plant de sensor staat en wat de GPS-locatie is.

Samenvatting interview Huibert-Jan Lekkerkerk (NHL Stenden)

Het gesprek met Huibert-Jan Lekkerkerk gaf aandacht aan het gebruik van sensoren op zee en het belang van het stellen van de juiste vragen bij het installeren en kiezen van sensoren.

Het gebruik van sensoren in de wetenschapswereld is nauwelijks te vergelijken met het gebruik van sensoren die het rendement van groene daken meten. Verschillende hoeveelheden data, verschillende frequenties van meten en andere databases. Hoewel er veel verschillen zijn, worden bij het installeren van beide typen sensoren dezelfde vragen gesteld. In het gesprek met Huibert-Jan kwamen een aantal belangrijke vragen aan bod, die beantwoord moeten worden voor het opzetten van een goede infrastructuur:

- Hoelang wil je doorgaan met tijdreeksen vastleggen?
- Hoe valideer je de data? / Wat doe je met foutieve data?
- Hoe ga je om met data standaardisatie?
- Is het van belang dat sensoren ook omgekeerd te lezen zijn? / Hoe wil je de sensoren kunnen configureren?

Samenvatting interview Jeroen van der Pluijm (Hanze Hogeschool)

Het gesprek met Jeroen van der Pluijm was gericht op een vorm van meten met sensoren die indirect aansluit op het SMART infrastructuur sensoren project. Bij de Hanze is men op veel verschillende manieren bezig met het 'slimmer' maken van bedrijfsprocessen binnen het gebouw. Dit gaat van de bezettingsgraad van lokalen tot de informatievoorziening bij de ingang van de school.

De Hanze Hogeschool werkt op dit moment aan een project waarbij de mogelijkheid onderzocht wordt om de bezetting/benutting van klaslokalen te meten met behulp van sensoren. Ook hiervoor worden Lora sensoren gebruikt. Doordat er twee gateways beschikbaar zijn op de school is het mogelijk om eens in de 12 seconden data op te halen over de bezetting van een werkplek. De gebruiker kan deze data in de toekomst raadplegen met behulp van een zelf ontworpen app: Plekki.