

Handleiding voor het werken met sensoren



Het gestandaardiseerde proces van sensor naar database

Marnix van den Dobbelsteen, Mathijs Spierings, Niek Bongers,
Stijn van Zagten en Troy Kanters



1. Introductie.....	3
1.1 Waarom werken met sensoren?	3
1.2 Gebruikte sensoren en software	3
1.3 De testlocaties	4
2. Randvoorwaarden	5
3. Het aanmaken van de database	6
3.1 Voorkennis.....	6
3.2 handleiding aanmaken van een gestandaardiseerde database	7
4. Het installeren van de sensor.....	10
4.1 Voorkennis.....	10
5. Het vullen van de database met metadata	11
5.1 Voorkennis.....	11
5.2 Het proces.....	11
6. Het proces van sensor naar database in node-RED	13
6.1 Voorkennis.....	13
7. Het beheer van de sensoren	20
7.1 Het plaatsen van de sensor op locatie.....	20
7.2 Het bijhouden van het batterijniveau	20
7.3 De sensor stuurt geen gegevens door, wat nu?	21
8. Frequently Asked Questions (FAQ)	22
Bijlagen.....	23
Bijlage 1: Lijst met Dragino sensoren	23

1. Introductie

1.1 Waarom werken met sensoren?

Door klimaatverandering worden steden beïnvloed door negatieve effecten, zoals hogere temperaturen, wateroverlast en ongezonde fijnstof in de lucht. Hierdoor neemt de maatschappelijke druk steeds verder toe, waardoor gemeenten en bedrijven de urgentie voelen om klimaatadaptief te werk te gaan. Het aanleggen van regenboog daken is hier een mooi voorbeeld van. Dit kunnen groene, witte, gele daken zijn en meer. Een regenboog dak kan verschillende voordelen voor de directe omgeving opleveren. Om aan te kunnen tonen dat regenboog daken daadwerkelijk bijdragen aan het verlagen van de temperatuur en de waterbuffering is het belangrijk om dit te onderbouwen met data. Sensoren kunnen gebruikt worden om deze data in te winnen.

Er zijn veel verschillende sensoren beschikbaar die gebruikt kunnen worden om data in te winnen. Daarnaast zijn er veel verschillende werkwijzen om aan de slag te gaan met sensoren. Om een gestandaardiseerde werkwijze op te stellen, is een groep 4^e-jaars AGIS studenten van de HAS green academy in het kader van het SURF project SMART sensordata infrastructuur aan de slag gegaan met het proces omtrent het inwinnen van data met sensoren. Hier is een werkwijze uit komen rollen die voor iedereen en overal werkt. In deze handleiding wordt de werkwijze stap voor stap uitgelegd.

Ben je een individu, organisatie of gemeente en heb je een regenboog dak waarvan je de meerwaarde wilt onderbouwen met data? Dan is deze handleiding geschikt voor jou. Je bent zelf vrij in de keuze van welke parameters je inzichtelijk gaat maken doormiddel van sensoren. Het eten van meteorologische parameters, bodemparameters, of de luchtkwaliteit, de handleiding is voor veel situaties geschikt. De keuze is aan jou. Zie figuur 1 hieronder om een indruk te krijgen van wat je gaat doen:



Figuur 1, sensor installeren, database aanmaken en database vullen met data en metadata doormiddel van node-RED

1.2 Gebruikte sensoren en software

Om tot een gestandaardiseerde werkwijze te komen is gebruik gemaakt van verschillende sensoren en software. In deze paragraaf wordt meer verteld over welke sensoren en software gebruikt zijn.

De sensoren

Voor het opstellen van deze werkwijze is gebruik gemaakt van sensoren van Dragino. Dit merk biedt veel verschillende type sensoren aan, voor het meten van temperatuur tot aan de luchtdruk. De volgende typen sensoren zijn gebruikt gedurende dit project:

- LSN50 v2-D20 voor het meten van de temperatuur
- LSE01 voor het meten van bodemtemperatuur, -vocht en -weerstand
- LHT65 voor het meten van de luchtvochtigheid

De software

Binnen de gestandaardiseerde werkwijze wordt gebruik gemaakt van verschillende software om tot het gewenste resultaat te komen. Er is gebruik gemaakt van open-source software, wat betekent dat het voor iedereen gratis toegankelijk is. Van de volgende software is gebruik gemaakt:

- The Things Network om contact te maken met de sensor
- Node-RED voor een verbinding naar de database
- PostgreSQL met PostGIS extensie voor het opslaan van data en metadata
- Python voor het wegschrijven van metadata
- Spyder als IDE (Integrated Development Environment) programma op python

Zie **bijlage 1** onderaan de handleiding voor een lijst van Dragino sensoren. Dit om een idee te geven welke sensoren er zijn en wat ze meten.

1.3 De testlocaties

Op verschillende locaties zijn al sensoren geplaatst om de gestandaardiseerde werkwijze te testen. In deze paragraaf wordt verteld waar de sensoren staan (figuur 1) en welke dit zijn, om een beeld te geven waar al tests zijn uitgevoerd met de gestandaardiseerde werkwijze.



- 1** Er staan drie verschillende sensoren op het groene dak van Groendakcoach in Hendrik Ido Ambacht. Een temperatuursensor op het groene dak, een bodem- en luchtvochtigheidssensor op het witte dak en een bodemsensor.
- 2** Er staat voor nu één temperatuursensor op het dak van het gemeentehuis in Steenberg. Deze sensor is gebruikt om het signaal van de gateway te testen.
- 3** Op het dak van de HAS zijn ook meerdere sensoren aanwezig waarmee de werkwijze getest is. Een temperatuursensor, een bodem sensor en een temperatuur- en luchtvochtigheidssensor.

Figuur 2, testlocaties in Nederland

De keuze voor de plaatsing van sensoren ligt volledig bij jou. De hoeveelheid, type sensoren en de locatie. Dit zijn de testlocaties geweest voor het project. De gestandaardiseerde werkwijze werkt voor alle locaties en niet specifiek voor de locaties in de kaart.



2. Randvoorwaarden

Voordat je met sensoren aan het werk kunt gaan, is het belangrijk om de volgende zaken tot je beschikking te hebben:

- Redelijke kennis van Geo-ICT om met data en de verschillende programma's te werken.
- Een account voor The Things Network
- Toegang tot een geo-database. Bij voorkeur PostgreSQL (Deze wordt gebruikt in de handleiding maar een andere geo-database is ook toepasbaar).
- Toegang tot een Node-RED server. Node-RED kan ook op je eigen systeem uitgevoerd worden, alleen kan de data dan enkel naar de database geschreven worden wanneer je eigen systeem aan staat.
- Toegang tot een LoRaWAN Gateway. Deze gateway is nodig voor het ontvangen van de data die een sensor verstuurd. Via [TTNmapper](#) kun je kijken of er een gateway in de omgeving aanwezig is. Wanneer er een slechte verbinding is met de gateway zal een deel van de data niet binnenkomen. Hierdoor ontstaan gaten in de data die het analyseren van de resultaten moeilijker maken. Mocht dit het geval zijn dan wordt aangeraden een gateway aan te schaffen.

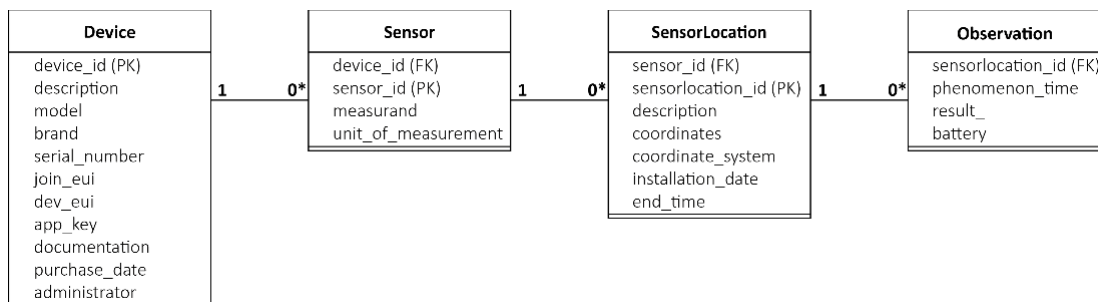
3. Het aanmaken van de database

In dit hoofdstuk ga je een database aanmaken waarnaar je de sensordata gaat wegschrijven. Voor deze stap is het nodig om over een geo-database te beschikken. In deze handleiding wordt gebruik gemaakt van PostgreSQL met PostGIS extensie. Dit is een eenmalige actie, zodra de database is aangemaakt kun je deze met data van alle sensoren gaan vullen.

3.1 Voorkennis

PostgreSQL is een opensource database software werkend op de SQL-taal. [Klik hier](#) om de software te downloaden. Mocht je gebruik willen maken van andere database software dan is dit ook toepasbaar (alleen aanbevolen voor ervaren gebruikers).

In de database wordt de structuur van figuur 3 aangehouden. In figuur 3 zie je vier tabellen staan. De tabellen **Device**, **Sensor** en **SensorLocation** zijn voor de metadata. De tabel **Observation** is voor het wegschrijven van de meetgegevens. De theorie hierachter is dat een device het apparaat is met de batterij, processor en verschillende meetsensoren. De sensor is de apparatuur die meet. De tabel **Device** is dus voor het invullen van device gegevens. De tabel **Sensor** voor het invullen van de sensor gegevens. De tabel **SensorLocation** is de locatie van de sensor, **niet** de locatie van het device. In de tabel **Observation** staan de observaties van alle metingen. Door de sleutelverbindingen tussen de tabellen is het herleidbaar welke waarneming van welke sensor en device komen. Een device kan dus meerdere sensoren bevatten. Ook kan een sensor verschillende locaties hebben.



Figuur 3, het datamodel.

Voorbeeld

Op het dak van de HAS green academy ligt een groen dak. Om bij te dragen aan het in kaart brengen van de meerwaarde van groene daken worden er allerlei factoren gemeten. Op het dak is een Dragino device geïnstalleerd. Hieraan zitten drie snoeren met aan het einde een sensor. Volgens de database-structuur geeft de metadata van het device informatie over het device, dit komt in de tabel **Device**. De tabel **Sensor** geeft aan wat de sensor meet, hier meten alle drie de sensoren de temperatuur. Deze drie sensoren worden individueel aangemaakt in de database. De tabel **SensorLocation** beschrijft wat elke sensor meet. Sensor 1 meet bijvoorbeeld de oppervlakte temperatuur van het groene dak. Sensor 2 meet de temperatuur tussen de dakbedekking en het groene dak. Sensor 3 meet de temperatuur één meter boven het groene dak. Bij de locatie wordt ook de installatie datum en de tijd genoteerd, zodat wanneer de sensor verplaatst wordt dit juist kan worden geregistreerd. Omdat in de **Observatie** tabel naar de juiste sensor wordt verwezen is het herleidbaar wat de meting inhoudt en wat er wordt gemeten.

3.2 handleiding aanmaken van een gestandaardiseerde database

Stap 1: Het toevoegen van de server en aanmaken van de database

Open pgAdmin en log in. Als je gebruik wilt maken van een database voor/van een organisatie zal je de connectie gegevens van de server nodig hebben. Deze bestaan uit een URL, poort, gebruikersnaam en wachtwoord. Neem hiervoor contact op met de IT-afdeling of beheerder van de databases. Deze kan de benodigde informatie verschaffen. Klik met de rechtermuisknop op **Servers**, **Create** en dan op **Server**. Geef de server een naam en klik op de tab **Connection**. Log in met de gegevens van uw organisatie.

Nu je toegang hebt tot de plek waar de database komt te staan kun je er een gaan aanmaken. Rechtermuisknop op je (lokale) **Server** dan **Create** en dan **Database**. Geef de database een logische naam, eigenaar en beschrijving en klik op **Save**.

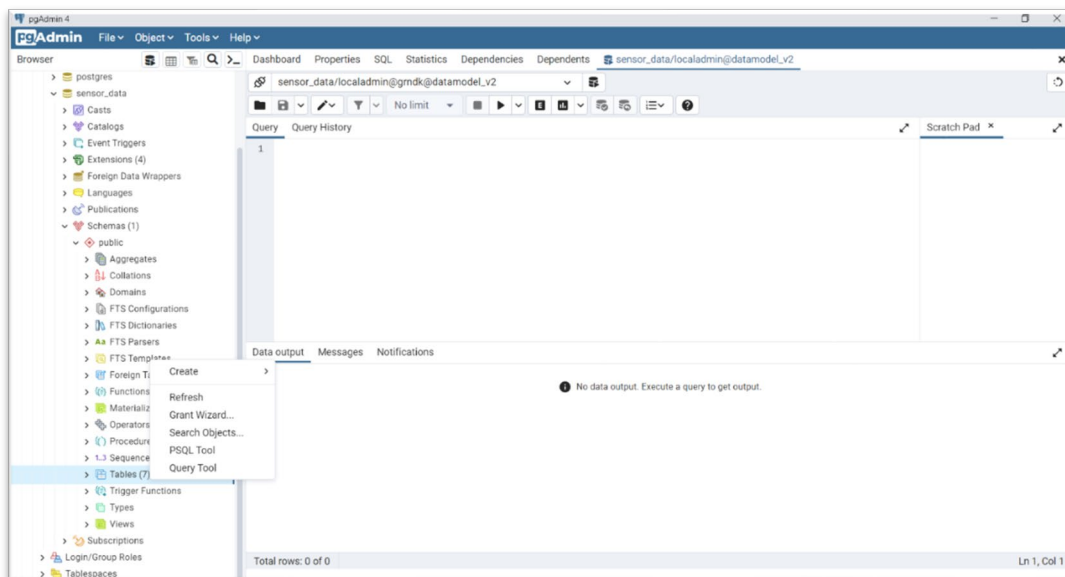
Stap 2: het toevoegen van een extensie

Met de standaardversie van PostgreSQL zijn veel dataformaten beschikbaar, Echter is het standaard niet mogelijk om met ruimtelijke data te werken. Om de ruimtelijke data van de sensoren op te slaan wordt de extensie PostGIS toegevoegd.

In de database klik je met de rechtermuisknop op **Extensions** dan **Create** en dan **Extension**. Selecteer vervolgens onder **Naam** PostGIS.

Stap 3: Het opzetten van de database structuur

Klik in je database schemas open, dan rechtermuisknop op tables en dan query tool. Hiermee komt het volgende scherm zoals figuur 4 naar voren.



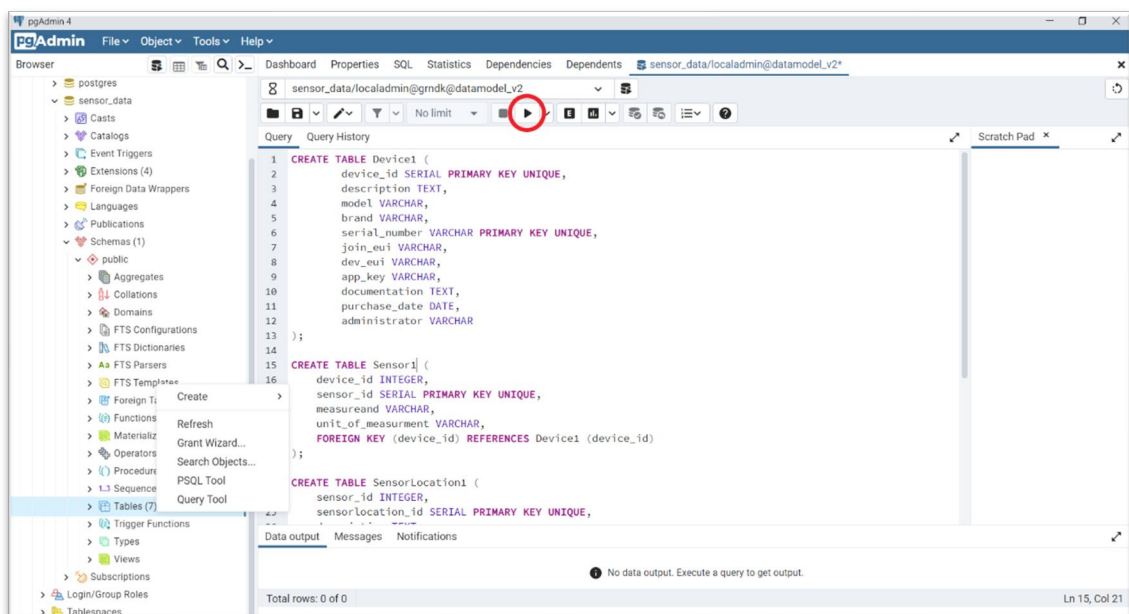
Figuur 4, de query tool

In dit scherm kunnen SQL-scripts worden gedefinieerd die commando's uitvoeren op de database. In dit hoofdstuk ga je een database structuur opzetten. De versimpelde vorm van hoe het SQL-script is opgebouwd: Eerst wordt er gedefinieerd dat er een tabel met opgegeven naam moet worden aangemaakt. Daarna de kolommen met datatype die de tabel moet bevatten. Als laatste de verwijzing van de sleutelkolommen om tabellen met elkaar te kunnen verbinden.

Nu ga je de tabellen aanmaken in de database die nodig is voor het metadata model. Kopieer de onderstaande code en plak dit in het bovenste grote witte vlak in de query tool.

```
CREATE TABLE Device (  
  device_id SERIAL PRIMARY KEY UNIQUE,  
  description TEXT,  
  model VARCHAR,  
  brand VARCHAR,  
  serial_number VARCHAR,  
  join_eui VARCHAR,  
  dev_eui VARCHAR,  
  app_key VARCHAR,  
  documentation TEXT,  
  purchase_date DATE,  
  administrator VARCHAR  
);  
  
CREATE TABLE Sensor (  
  device_id INTEGER,  
  sensor_id SERIAL PRIMARY KEY UNIQUE,  
  measueand VARCHAR,  
  unit_of_measurement VARCHAR,  
  FOREIGN KEY (device_id) REFERENCES Device (device_id)  
);  
  
CREATE TABLE SensorLocation (  
  sensor_id INTEGER,  
  sensorlocation_id SERIAL PRIMARY KEY UNIQUE,  
  description TEXT,  
  coordinates GEOMETRY (point),  
  coordinate_system VARCHAR,  
  installation_date TIMESTAMPTZ,  
  end_time TIMESTAMPTZ,  
  FOREIGN KEY (sensor_id) REFERENCES Sensor (sensor_id)  
);  
  
CREATE TABLE Observation (  
  sensorlocation_id INTEGER,  
  phenomenom_time TIMESTAMPTZ,  
  result_ DOUBLE PRECISION,  
  battery DOUBLE PRECISION,  
  FOREIGN KEY (sensorlocation_id) REFERENCES SensorLocation (sensorlocation_id)  
);
```

Run de code zoals in figuur 5, via de **Execute** knop in de rode cirkel.



Figuur 5, het uitvoeren van een script in pgAdmin

Als alles goed gaat zijn nu de vier tabellen van de database aangemaakt, zo niet staat er een foutmelding onder 'Data output'. Als het in een al bestaande database wordt weggeschreven kan het zijn dat de tabelnaam al gebruikt is. De variabele tabelnamen van het script kunnen dan worden aangepast. Dit kan makkelijk door een '1' achter de naam te plaatsen. Ook kan je een beschrijving achter de naam zetten. **LET OP:** dit moet dan ook veranderd worden in de tabel waarnaar de Foreign Key wordt verwezen.

Door **tables** uit te vouwen zijn de nieuwe tabellen te zien. Door in een table de **columns** uit te vouwen zijn ook de aangemaakte kolommen zichtbaar. Voor het overzicht wordt aangeraden om dit weer in te vouwen na inspectie.

4. Het installeren van de sensor

In dit hoofdstuk ga je de sensor aanzetten en verbinden in The Things Network (hierna: TTN). Hiervoor is een account nodig op TTN. Heb je nog geen account? [Klik hier](#) om een account aan te maken.

4.1 Voorkennis

TTN is een netwerk waaraan LoRaWAN apparaten verbonden kunnen worden. Dit netwerk gaat gebruikt worden om de gegevens die een sensor inwint op te halen, zodat je deze weg kan schrijven naar een database.

Stap 1. Het aanmaken van een organisatie

Deze stap is alleen nodig wanneer het gewenst is dat er meerdere personen toegang moeten hebben tot de sensor in TTN.

Log in op TTN en klik op je account, klik op console en kies dan voor Europe 1. Je bent nu in TheThings Stack omgeving. Navigeer naar de **Organizations** tab en klik op **Create organization**. Geef je organisatie een ID, een naam en omschrijving. De organisatie is nu aangemaakt, hierin kunnen meerdere personen toegang krijgen tot de aangesloten sensoren. Om anderen toe te voegen aan de organisatie klik je op **Collaborators** en vervolgens op **Add collaborator**. Hier vul je het Collaborator ID en de rechten van de persoon in. Klik op **Add collaborator** om de persoon toe te voegen aan de organisatie.

Stap 2. Het aanmaken van een applicatie en verbinden van de sensor

Log in op TTN en klik op je account, klik op **console** en kies dan voor **Europe 1**. Je zit nu in The Things Stack omgeving. Hier ga je een applicatie aanmaken waaraan je de sensor gaan verbinden. Navigeer naar applications en klik op **Create application**. Geef aan wie de eigenaar van de applicatie is. Dit kan je zelf zijn of de organisatie die in stap 1 is aangemaakt. Daarnaast geef je de applicatie een toepasselijk ID, een naam en beschrijving. Klik op **Create application** om de applicatie aan te maken.

Als het goed is zit je nu in de applicatie. Klik op de knop **Register end device**. Als **Input Method** kies je **Select the end device in the LoRaWAN Device Repository**. Selecteer dan het merk van je sensor en vul de andere gegevens in die gevraagd worden. Er wordt gevraagd naar het **Model**, de **Hardware Version** en **Firmware Version**. Deze zijn te vinden in de gebruikershandleiding van de sensor. De gebruikershandleiding is online te vinden op de website van de verkoper of fabrikant. Dan moet de **Profile (Region)** worden ingevuld. Selecteer hier een van de **EU** mogelijkheden. Kies bij **Frequency plan** voor de in de gebruikershandleiding aangegeven mogelijkheid. Mocht deze niet aangegeven staan selecteer dan het **Frequency plan** die als Recommended aangegeven wordt.

Nu moeten je de unieke gegevens van de sensor invullen. Dit zijn het **'JoinEUI'**, **'DevEUI'** en de **'AppKey'**. Deze zijn te vinden in de doos van de sensor en staan vaak ook op de sensor zelf. Geef de sensor een logisch ID. (Benoem bijvoorbeeld waar deze komt te staan of wat deze meet, zodat dit achteraf duidelijk is en niet voor verwarring met eventuele andere sensoren kan zorgen.) Wanneer alles correct is ingevuld, klik op **Register end device**.

Stap 3. Het aanzetten van de sensor

Wanneer de sensor is aangezet, kan de verbinding worden gecontroleerd. Volg voor het aanzetten van de sensor de gebruikershandleiding. Wanneer de sensor aan is gezet zullen er gegevens binnen komen in TTN. Navigeer naar **End devices** en controleer of er een activiteit is geweest van de sensor. Als alles goed is gegaan zou je deze activiteit moeten zien bij 'Last activity'. Kijk anders of je een oplossing kan vinden bij de Frequently Asked Questions (FAQ).

5. Het vullen van de database met metadata

In dit hoofdstuk ga je aan de slag met het vullen van de database met metadata. Dit kan gedaan worden doormiddel van een python script. Alles wordt uitgelegd om je door het proces heen te helpen. Dit is een belangrijk onderdeel, omdat metadata zorgt voor betrouwbare data. Deze actie dient voor elke sensor herhaalt te worden.

5.1 Voorkennis

Om van de ingewonnen data waardevolle data te maken is het van belang metadata toe te voegen. Metadata geeft informatie weer over de ingewonnen data. Door het toevoegen van metadata is de data herleidbaar en kan de gebruiker de actualiteit, betrouwbaarheid en compleetheid van de data controleren en zo beslissen of het gebruikt kan worden voor een onderzoek. De tabellen zijn in de vorige stap aangemaakt, deze ga je nu vullen.

Voor de volgende stappen wordt gebruikt gemaakt van een **script**. Dit is een opeenvolgende lijst met instructies. Dit script is geschreven in **python**. Python is een programmeertaal, veelal gebruikt voor het werken met data. De functionaliteiten zijn eindeloos.

In de volgende stappen word python gebruikt voor het invullen van de tabellen met de metadata. Uiteraard is een installatie van Python vereist voor het uitvoeren van deze stap. Wanneer je het nog niet hebt geïnstalleerd **kan dat hier**.

Om het python script naar behoren te bewerken dient het te worden geopend in een **IDE**, oftewel Integrated Development Environment. Dit is software waarin de code kan worden bewerkt. Er zijn talloze IDE's op de markt. Het maakt niet uit welke wordt gebruikt maar wij bevelen **Spyder** aan.

5.2 Het proces

Om te beginnen, open het bestand **'edit_insert_statement.py'**, met een IDE. Er zijn in essentie drie stappen die doorlopen dienen te worden. Het installeren van een extensie, het creëren van de database connectie en het vullen van de metadata waarden.

Installeren van pycogp2 extensie

Het script maakt op een gegeven moment verbinding met de aangemaakte database. Dit valt echter niet in de standaard functionaliteiten van Python, hier dient dus een extensie voor te worden geïnstalleerd. Gelukkig is dit erg simpel om te doen. Open de **'terminal'** en voer de volgende lijn uit.

```
pip install pycogp2
```

Wanneer dit gelukt is zal de melding verschijnen. Wanneer dit niet het geval is en er een error verschijnt kan de volgende **website** worden geraadpleegd.

Database-connectie

Om waarden (zoals het **'device_id'** en **'sensor_id'**) uit de database op te halen, dient er een connectie gemaakt te worden. Dit wordt bereikt met de eerder geïnstalleerde extensie **pycogp2**. Om deze connectie tot stand te laten komen, dienen de eigen waarden van de database ingevuld te worden. Ga hiervoor naar lijn 29, in het script. Pas hier de waarden **'database'**, **'user'**, **'password'** en **'host'** aan, naar de waarden van de eerder aangemaakte database.

Metadata-waarden

Pas onder het kopje 'variables' alle gewenste waarden aan. Weet dat naast 'device_id' en 'sensor_id' geen variabelen verplicht zijn. Je hoeft alleen degene in te vullen die beschikbaar en gewenst zijn. Let op dat de waarde tussen zowel het enkele aanhalingsteken als het dubbele aanhalingsteken ingevuld dient te worden. Zie dit voorbeeld:

```
'brand': """"
```

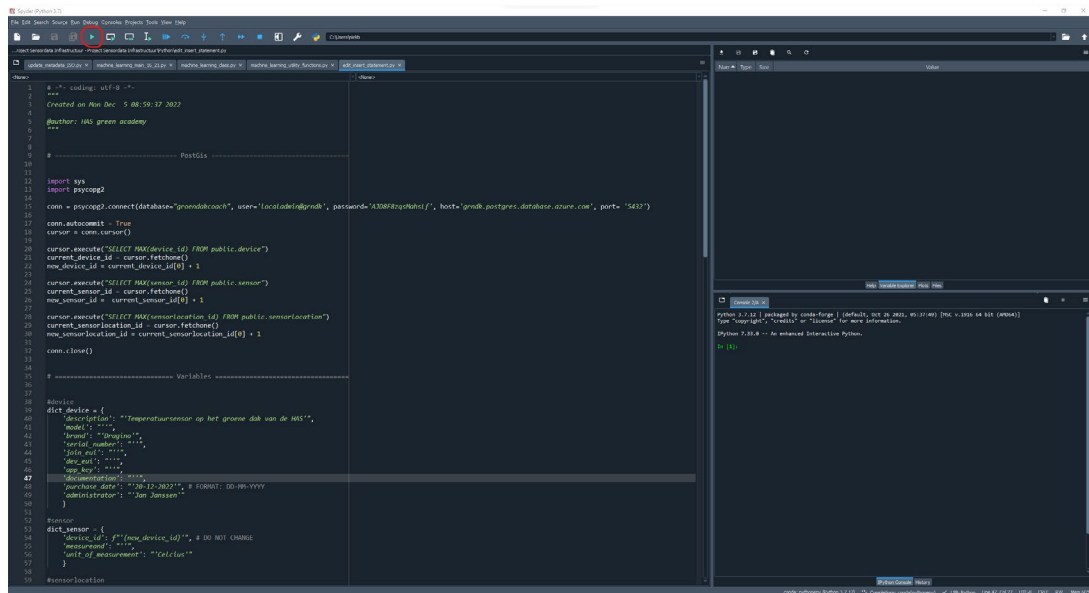
Kan in dit geval worden veranderd naar:

```
'brand': ""Dragino""
```

Dit is later van belang in de uitvoering van de query in PostgreSQL.

Ook de notatie van de waarde kan, afhankelijk van de variabele, van belang zijn. Wanneer de waarde specifiek een bepaald formaat dient te volgen om correct te werken, staat áchter de variabele de notatie '#FORMAT:' met daarbij het gewenste formaat. Dit is voor nu echter alleen het geval bij 'purchase_date' die genoteerd dient te worden als 'DD-MM-YYYY'.

Wanneer alle gewenste waarden zijn ingevuld, kan het script worden uitgevoerd. Dit kan (met Spyder) via de **Run file (F5)** knop, zoals te zien is in figuur 6.



Figuur 6, het uitvoeren van een script in Spyder

De uitvoer van het script (figuur 7) bestaat uit blokken tekst met witregels ertussen. Deze blokken dienen één voor één uitgevoerd te worden in PostgreSQL. Navigeer naar pgAdmin, open de betreffende database, navigeer naar de 'Query Editor' en voer het blok uit. Herhaal dit voor de overige twee blokken. Wanneer dit is uitgevoerd zal de metadata worden toegevoegd aan de tabellen.

```
INSERT INTO device (description,brand,purchase_date,administrator)
VALUES ('Temperatuursensor op het groene dak van de HAS', 'Dragino', '20-12-2022', 'Jan Janssen');

INSERT INTO sensor (device_id,unit_of_measurement)
VALUES ('8', 'Celcius');

DELETE FROM sensorlocation;
INSERT INTO public.sensorlocation(sensor_id)
VALUES ('11');
```

Figuur 7, de uitvoer van het python script

6. Het proces van sensor naar database in node-RED

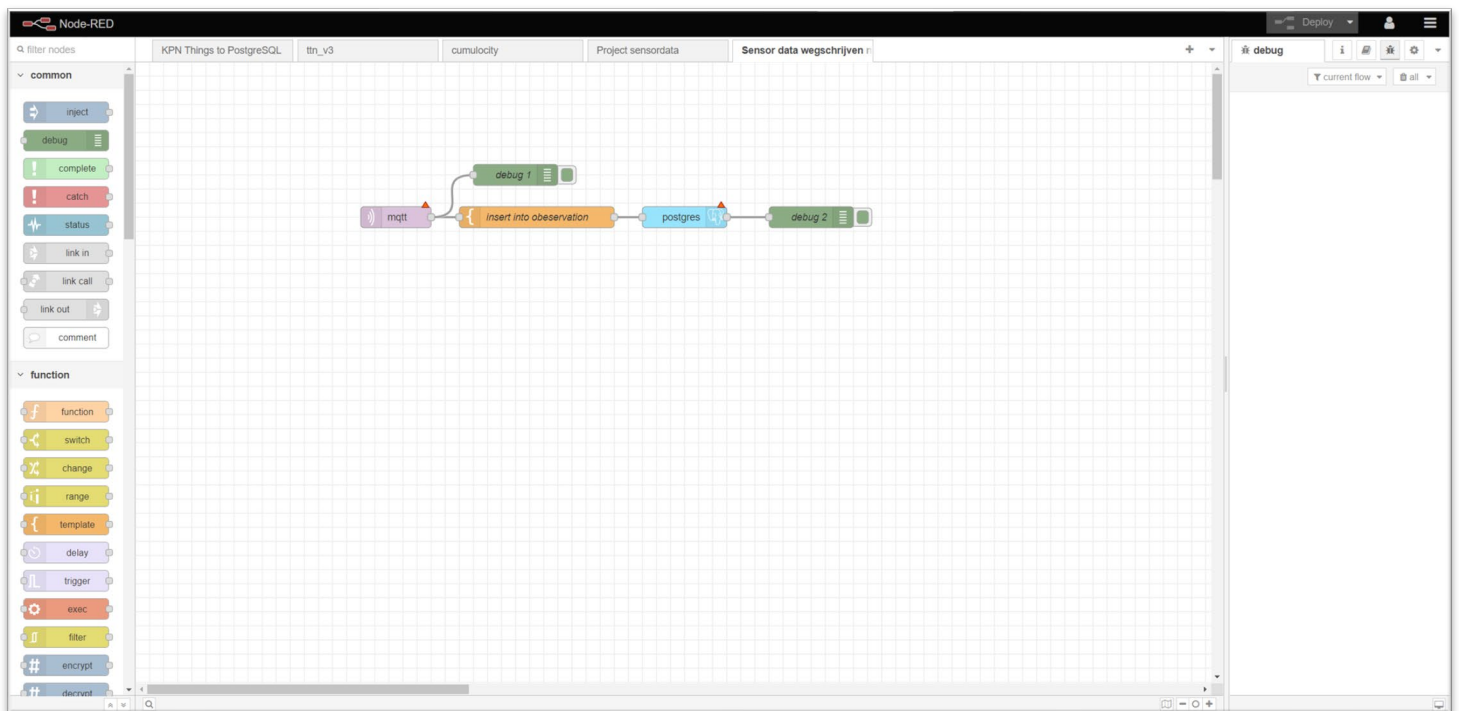
In dit hoofdstuk ga je de binnengehaalde sensordata wegschrijven naar de juiste tabel in de database. Hiervoor ga je gebruik maken van Node-RED.

6.1 Voorkennis

Node-RED is een programma waarmee je hardware, API's en online services met elkaar kunt verbinden. Het programma draait in een browser en zorgt ervoor dat je zonder veel te coderen toch een sensordata workflow op kan zetten. Je gaat een workflow opzetten die de sensor in TTN verbindt aan de PostgreSQL database. Uiteindelijk wordt de data automatisch weggeschreven naar de database.

Node-RED

Node-RED bestaat uit 3 verschillende panelen, weergegeven in figuur 8. In het linker paneel staan de nodes. Deze worden gebruikt voor input, functies en output. De nodes zijn naar het middelste paneel te slepen, waarin ze ingesteld kunnen worden. Voor het verbinden van een sensor uit TTN wordt gebruik gemaakt van een **MQTT broker node**. Daarnaast wordt er gebruik gemaakt van de Template node en een Postgres node. De Postgres node moet nog geïnstalleerd worden, dit ga je doen bij stap 1.



Figuur 8, Node-RED

Stap 1. Het installeren van de Postgres node

Als eerst ga je de Postgres node installeren. Deze is namelijk niet standaard geïnstalleerd in Node-RED. Klik hiervoor in Node-RED rechtsboven op de 3 streepjes en klik op [manage palette](#). Klik op [install](#) en zoek naar: [node-red-contrib-re-postgres](#). Installeer deze node.

Stap 2. Het inladen van de nodes

Om deze stap makkelijker te maken is het mogelijk om de nodes die je gaat gebruiken allemaal in een keer in te laden. Kopieer hieronder de benodigde tekst.

```
[
  {
    "id": "8f2af29da83cd8c3",
    "type": "tab",
    "label": "Sensor data wegschrijven naar database",
    "disabled": false,
    "info": "Deze flow is een ",
    "env": []
  },
  {
    "id": "430c0be692263bd1",
    "type": "template",
    "z": "8f2af29da83cd8c3",
    "name": "insert into obeservation",
    "field": "payload",
    "fieldType": "msg",
    "format": "sql",
    "syntax": "mustache",
    "template": "INSERT INTO public.observation (sensorlocation_id, phenomenom_time, result_,
battery)\nVALUES ('sensorlocatie id', current_timestamp, {{path naar meetresultaat, gekopieerd uit het
debug venster}}, {{path naar battery, gekopieerd uit het debug venster}})",
    "output": "str",
    "x": 570,
    "y": 220,
    "wires": [
      [
        "929dfa2b92a252d1"
      ]
    ]
  },
  {
    "id": "71d50d6d2badd470",
    "type": "debug",
    "z": "8f2af29da83cd8c3",
    "name": "debug 2",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "targetType": "msg",
    "statusVal": "",
    "statusType": "auto",
    "x": 960,
    "y": 220,
    "wires": []
  },
  {
    "id": "6232f18da266ee5b",
    "type": "mqtt in",
    "z": "8f2af29da83cd8c3",
    "name": "",
    "topic": "",
    "qos": "2",
    "datatype": "auto-detect",
    "nl": false,
    "rap": true,
    "rh": 0,
    "inputs": 0,
    "x": 370,
    "y": 220,
    "wires": [
      [
        "430c0be692263bd1",
        "cbf8a28a66e2acb5"
      ]
    ]
  },
  {
    "id": "929dfa2b92a252d1",

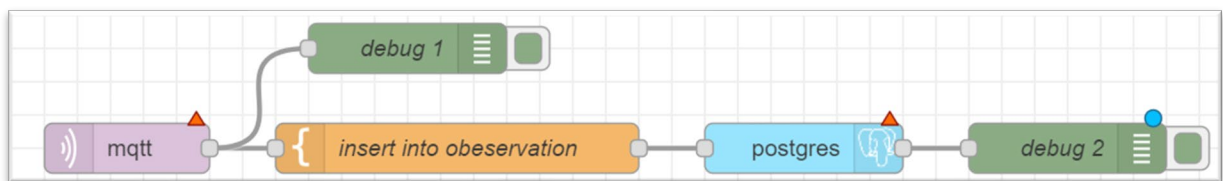
```

```

    "type": "postgres",
    "z": "8f2af29da83cd8c3",
    "postgresdb": "",
    "name": "",
    "output": true,
    "perrow": false,
    "rowspermsg": "1",
    "return_on_error": false,
    "limit_queries": "0",
    "limit_by": "payload",
    "limit_value": "1",
    "limit_drop_intermediate": false,
    "limit_drop_if_in_queue": false,
    "outputs": true,
    "x": 780,
    "y": 220,
    "wires": [
      [
        "71d50d6d2badd470"
      ]
    ]
  },
  {
    "id": "cbf8a28a66e2acb5",
    "type": "debug",
    "z": "8f2af29da83cd8c3",
    "name": "debug 1",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "targetType": "msg",
    "statusVal": "",
    "statusType": "auto",
    "x": 540,
    "y": 160,
    "wires": []
  }
]

```

Klik in Node-RED op de 3 streepjes rechtsboven en klik op **import**, plak daar de tekst en importeer de nodes. Als het goed is zie je nu de nodes zoals in figuur 9 voor je. Aan de debug nodes hoeft je niets meer te veranderen. De MQTT Broker node, de template node en de Postgres node ga je in de volgende stappen instellen.

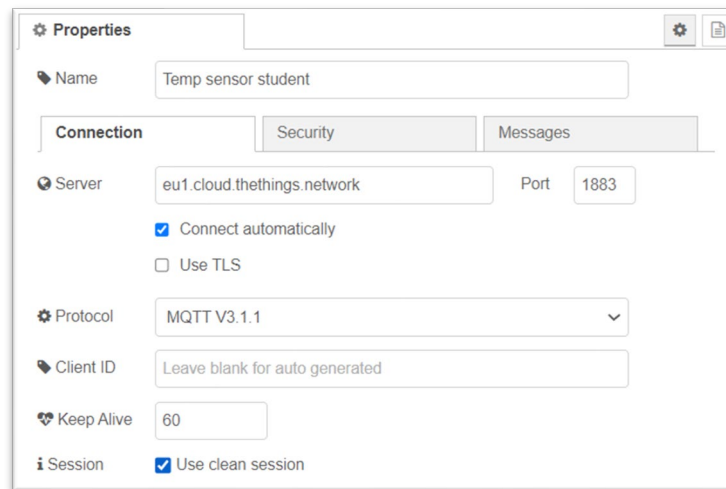


Figuur 9, nodes in de workspace

Let op: Mocht deze stap niet werken is het mogelijk om de nodes handmatig uit het linker paneel de workflow in te slepen.

Stap 3. Het instellen van de MQTT Broker node

Je gaat nu de MQTT Broker node instellen. Dit is de eerste node. Open de node en klik op het potloodje naast de server. Je ziet nu een menu voor het verbinden van een sensor zoals weergegeven in figuur 10. Voor het invullen van de juiste gegevens hebben je TTN nodig, dus zorg dat je deze open hebt staan.

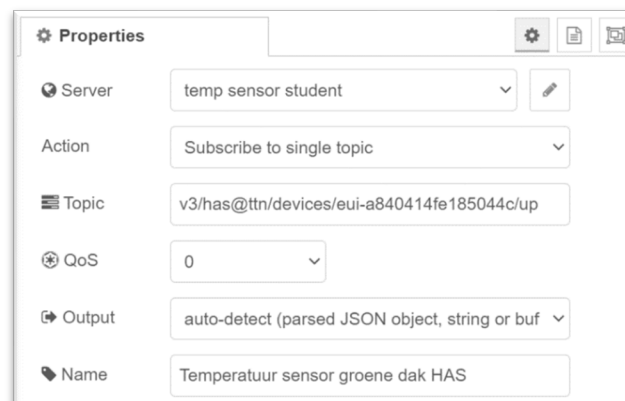


Figuur 10, MQTT server verbinding

Open in TTN de **Application** met de gewenste sensor en kies in het linker menu **Integrations** en dan **MQTT**. Uit dit scherm kun je het public server adres halen, wat ingevuld moet in Node-RED, zoals gedaan is in figuur 10. In het veld **Name** geef je de sensorverbinding een naam. Zorg dat **Connect automatically** geselecteerd is en dat het protocol op **MQTT V3.1.1** staat. Klik vervolgens op de **Security** tab.

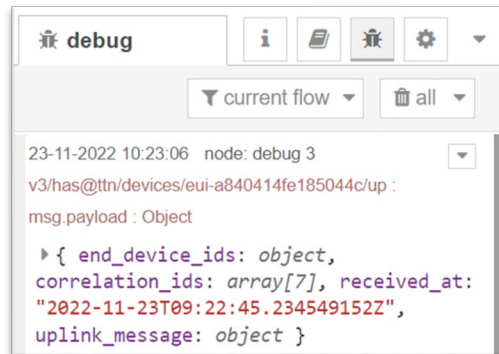
Hier vul je de gegevens in die in TTN onder **Connection credentials** staan. Dit zijn de **Username** en het **Password**. Het password is een **API key**, die maar één verbinding tot stand kan brengen. Als je de sensor dus in meerdere workflows wilt gebruiken dan moet je daar een nieuwe API key voor gebruiken. Klik op **Done**. Nu ga je het laatste scherm invullen.

In dit scherm (zie figuur 11) moet de Topic nog op 'v3/**Username**/devices/**End device ID**/up' gezet worden. Gebruik hier bij **Username** dezelfde gebruikersnaam als bij de verbinding en bij **End device ID** vul je het ID van de sensor in. Deze is te vinden in TTN in de Application in het linker paneel **End devices**. Daar staat een lijst met de ingestelde sensors en het bijbehorende ID. Daarnaast moet **QoS** op '0' worden gezet. Alleen deze QoS wordt ondersteund door TTN. Als laatste kun je de node een naam geven. Het scherm zal er dan ongeveer uit moeten zien als figuur 11. Klik op **Done** als dit zo is.



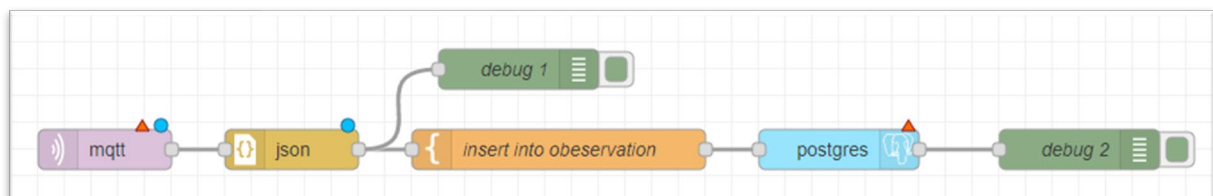
Figuur 11, ingevulde MQTT node

Om te testen of de verbinding goed tot stand is gekomen ga je de workflow uitvoeren. Dit doen je met de rode **Deploy** knop rechtsboven. Wanneer alle stappen goed zijn doorlopen verschijnt er bij de MQTT node een groen vakje met 'connected'. Dit betekent dat de verbinding tussen TTN en Node-RED geslaagd is. Is dit het geval, dan kan de payload bekeken worden in het debug menu. Klik hiervoor in het rechter paneel op het insect-icoon **debug**.



Figuur 12, debug scherm met payload

Wanneer er data is binnen gekomen ziet dat er uit net als in figuur 12 . Dit is de payload die wordt gebruikt in de volgende stap. Het kan dus zijn dat je moet wachten tot er een payload van de sensor binnen komt. Ook kan het zijn dat de payload niet juist binnen komt. In dat geval zie je waarschijnlijk alleen maar rode tekst. De payload komt dan als string (tekst) binnen. Je wilt de payload als **JSON** binnen halen. Dit kan je op verschillende manieren bereiken. Een mogelijke manier is het toevoegen van een JSON node. Sleep deze vanuit het node paneel tussen de MQTT node en de template node. Zorg ook dat de debug node achter de JSON node komt. Dit ziet er dan uit als figuur 13.



Figuur 13, toevoeging van een JSON node

Mocht deze methode niet werken is het ook nog mogelijk om de payload in TTN te transformeren naar JSON. Hiervoor moet je de Payload formatter aanpassen. Dit kun je in TTN doen door bij **end devices** de sensor te openen en dan naar de tab payload formatters te navigeren. Hier moeten je de '**custom javascript formatter**' selecteren. Hier ga je de formatter van de leverancier van de sensor plakken. Deze payload formatter (als deze beschikbaar is voor de sensor) is te vinden in de online gebruikershandleiding van de leverancier. Wanneer je de juiste formatter hebt gevonden is het een kwestie van kopiëren, plakken. Klik daarna op **save changes**.

Stap 4. Het instellen van de template node

Je gaat nu de template node instellen. Deze ziet er als het goed is uit als in figuur 15. Mocht de tekst in de template node er niet in staan is het mogelijk de volgende tekst te kopiëren en zoals in figuur 15 te plakken:

```
INSERT INTO public.observation (sensorlocation_id, phenomenom_time, result_, battery)
VALUES ('sensorlocatie id', current_timestamp, {{path naar meetresultaat, gekopieerd uit het debug
venster}}, {{path naar battery, gekopieerd uit het debug venster}})
```



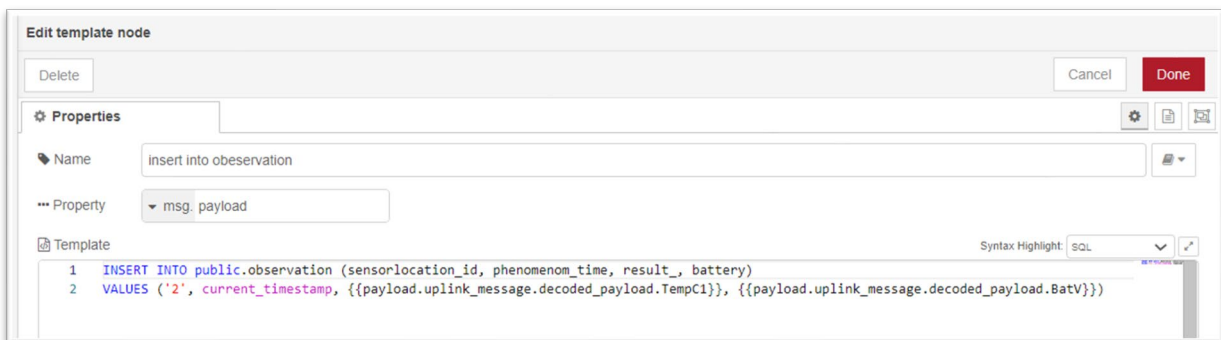
Figuur 15, de template node

Er moet nog één aanpassing gemaakt worden om de template node te laten werken. In plaats van de aanwezige rode tekst in aanhalingstekens moet het **sensorocation_ID** ingevuld worden. Dit ID is te vinden in de tabel **SensorLocation** van de eerder gemaakte PostgreSQL database. Let hierbij op dat je de aanhalingstekens laat staan. Daarnaast heb je het **'path'** (pad) nodig naar het meetresultaat van de sensor en het batterij voltage. Deze is te vinden in het debug menu, in de payload van de sensor. In het geval van de temperatuursensor LSN50 v2-D20 staat het meetresultaat onder **uplink_message** dan **decoded_payload** en daar staat 'TempC1' en 'BatV'. Dit is ook weergegeven in figuur 16.



Figuur 16, pad naar meetresultaat

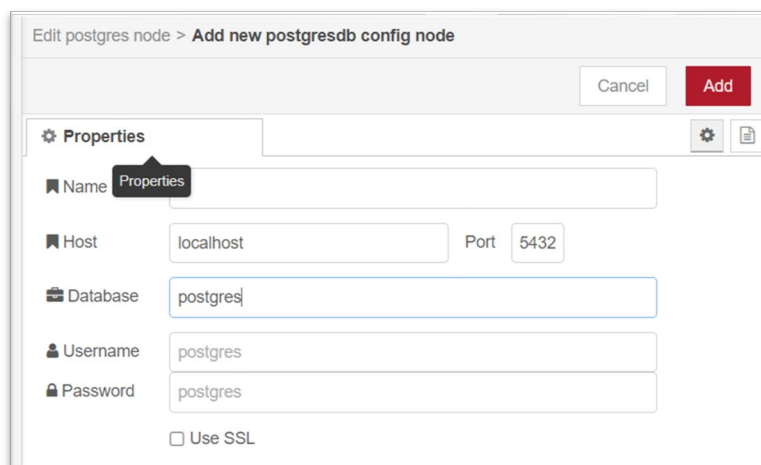
Het is hier dus van belang om je eigen meetresultaat in de payload te vinden. Wanneer deze gevonden is kan met de knop **'Copy path'**, ook weergegeven in figuur 16, het pad gekopieerd en geplakt worden op de juiste plek in de template node. De template ziet er dan ongeveer uit zoals weergegeven in figuur 17.



Figuur 17, eindresultaat template node

Stap 5. Het instellen van de postgres node

Bij stap 1 heb je de Postgres node geïnstalleerd. Deze ga je nu instellen. Dubbelklik op de node en klik op **het potloodje** naast server om de verbinding in te stellen. Het scherm ziet eruit zoals in figuur 18. In het veld **'Name'** vul je een naam naar keuze in, bij **'Host'** de link naar server van de database met de daarbij behorende **'Port'**. Bij **'Database'** vul je de naam in van de database in die je hebt aangemaakt in PostgreSQL. Tot slot moet je de inloggegevens van de database die je gebruikt hebt bij hoofdstuk 4.2 invullen. Klik dan op **Add**. Nu kun je de node nog een naam geven. Zorg er voor dat **Receive query output?** aangevinkt is. Klik op **Done**.



Figuur 18, Postgres node

Alle nodes zijn nu ingesteld. Om de laatste wijzigingen door te voeren moet de workflow nog uitgevoerd worden met de rode knop **Deploy** rechtsboven. Als alle stappen correct zijn doorlopen wordt de database nu automatisch gevuld met sensordata.

7. Het beheer van de sensoren

Nadat de structuur van sensor naar database is opgezet, is het belangrijk om de sensor op de juiste locatie te plaatsen en deze vervolgens goed te blijven beheren. Hoe dit vormgegeven moet worden, wordt in dit hoofdstuk van de handleiding verder toegelicht.

7.1 Het plaatsen van de sensor op locatie

Wanneer de sensor is geïnstalleerd moet deze natuurlijk nog op de juiste locatie geplaatst worden om de gewenste parameter te meten. Hierbij zijn verschillende zaken van belang. Ten eerste is het belangrijk om te controleren of de sensor gedurende een lange periode in de buitenlucht kan blijven. Wanneer dit het geval is, is het ook belangrijk om het device goed dicht te maken na het eventueel openmaken hiervan tijdens de installatie. Dit omdat er anders vocht in het device kan komen waardoor deze kapot kan gaan. Als er gecontroleerd is of het device goed dicht zit en deze geschikt is om in de buitenlucht te plaatsen moet hiervoor een locatie gekozen worden.

De locatie zal per sensor anders kunnen zijn. In het geval van een temperatuursensor kan hierbij bijvoorbeeld gekozen worden voor een plek recht boven op een regenboog dak. Indien er meerdere sensoren beschikbaar zijn is het mogelijk om op deze manier, door het plaatsen van de sensoren op verschillende soorten daken, temperatuurverschillen te meten. Een andere mogelijkheid is, bij het gebruiken van sensoren voor onder andere luchtdruk en luchtkwaliteit, het plaatsen van de sensor in de lucht boven bijvoorbeeld een groen dak versus een grijs dak. Weer een andere mogelijkheid is het plaatsen van een sensor in de bodem op bijvoorbeeld een akker of een groen dak. Dit maakt het mogelijk om de bodemvochtigheid, bodemtemperatuur en geleidbaarheid te meten. Het is dus bij verschillende soorten sensoren mogelijk om op verschillende locaties verschillende dingen te meten en te vergelijken.

Wanneer de sensor geplaatst is, is het vervolgens nog wel van belang om het device op een goede manier en locatie te plaatsen. Zorg hierbij dat deze stevig vastgemaakt wordt aan een vast en stabiel voorwerp. Een voorbeeld hiervan is een standaard welke langs de meetlocatie geplaatst kan worden. Tot slot is het belangrijk dat de batterij nog toegankelijk is zonder het device volledig los te moeten maken.

7.2 Het bijhouden van het batterijniveau

Bij het aanmaken van de database is er een kolom gemaakt waar het batterijvoltage van de sensor in weggeschreven wordt. Om dit voltage van de eigen sensor te vergelijken met het streefniveau van dit type sensor moet de handleiding van de sensor geraadpleegd worden. In deze handleiding staat een voltagewaarde waarbinnen de sensor goed zal werken. Het batterijvoltage van de eigen sensor is bij het precies volgen van deze sensordata handleiding weggeschreven in de tabel **Observation** in de database. Wanneer het voltage in de database na verloop van tijd scherp begint te dalen moet de batterij van de sensor vervangen worden. De periode waarna dit te verwachten zal zijn is afhankelijk van het sensortype dat gebruikt wordt en staat in de handleiding hiervan. Hoe de batterij vervangen moet worden staat ook beschreven in de handleiding van de sensor.

7.3 De sensor stuurt geen gegevens door, wat nu?

Het kan voorkomen dat de sensor een keer geen gegevens doorstuurt naar de database. Dit kan verschillende oorzaken hebben:

Het signaal naar de gateway is te zwak, waardoor de sensor geen verbinding kan maken en dus ook geen gegevens kan doorsturen.

Om dit probleem te voorkomen is het monitoren van de signaalsterkte van belang. In de payload van de sensor wordt de 'RSSI' meegenomen, wat staat voor Received Signal Strength Indicator. Deze indicator kan weggeschreven worden naar een database.

De signaalsterkte is verdeeld over verschillende categorieën. Deze zien er als volgt uit:

- > -100 dBm → Zeer sterk
- 100 -- 105 → Sterk
- 105 -- 110 → Matig tot sterk
- 110 -- 115 → Matig tot zwak
- 115 -- 120 → Zwak
- < -120 dBm → Zeer zwak

Uit tests is gebleken dat een signaal die regelmatig matig tot zwak is, af en toe volledig weg kan vallen waardoor er geen gegevens van de sensor in de database komen.

Als dit vaak voorkomt is de enige oplossing het aanschaffen van een gateway, die binnen uw organisatie geïnstalleerd wordt. Zo blijft het signaal altijd sterk en worden er geen metingen van de sensor gemist.

De sensor is niet juist verbonden in TTN, waardoor deze geen verbinding kan maken in node-RED. Hierdoor kan de sensor geen gegevens sturen.

Dit probleem kan optreden als van TTN omgeving gewisseld wordt. Bijvoorbeeld het overstappen van een testomgeving naar de uiteindelijke omgeving. Je zult dan zien dat de sensor geen verbinding maakt.

Om dit op te lossen, dient de sensor gereset te worden. Dan kan niet op een afstand uitgevoerd worden, maar moet fysiek bij de sensor gebeuren. Als je de sensor open schroeft, zie je daar een reset knop. Druk deze in om de sensor te resetten.

Nadat de sensor gereset is, moet deze opnieuw geïnstalleerd worden, kijk hiervoor naar het hoofdstuk 'Het installeren van de sensor'.

8. Frequently Asked Questions (FAQ)

Ik heb geen verbinding kunnen maken met de sensor in The Things Network, hoe kan ik dit oplossen?

De sensor verbindt automatisch met een gateway die de data doorstuurt naar TTN. Via

TTNmapper kun je checken of er een gateway in de omgeving is.

Mocht het nog niet lukken op een connectie te vormen dan kun je de sensor resetten en opnieuw verbinden. Verwijder de sensor dan uit TTN en kijk in de gebruikershandleiding van de sensor om er achter te komen hoe je deze reset.

De sensor heeft geen verbinding in node-RED, hoe los ik dit op?

Controleer of de verbinding gegevens goed ingevuld zijn (url, poort, gebruikersnaam en wachtwoord). Ook kan je proberen een nieuwe API key te gebruiken. Deze kun je aanmaken in TTN.

Zijn sensoren duur?

Nee. Sensoren zijn voor een heel schappelijke prijs al te verkrijgen. Uiteraard zijn er ook enorm dure modellen beschikbaar, met daarbij komende voordelen. Toch zijn er met een budget van € 80,00 al een heel aantal betrouwbare sensoren beschikbaar.

Kan ik met weinig IT en/of data kennis aan de slag met sensoren?

Om aan de slag te gaan met sensor is het handig om enige ervaring te hebben met SQL en Python. Daarnaast is algemene IT ervaring handig bij het oplossen van problemen. Deze kennis is voornamelijk nodig voor het opzetten van het sensor netwerk. Als hier iemand anders bij kan helpen is er verder in het proces deze kennis niet nodig. Dit komt omdat de sensoren (als het goed is) autonoom meten. Zo kan je toch zelf met sensordata weken.

Waarom verstuurt mijn sensor soms geen data?

Dit kan komen door verschillende redenen: de batterij kan leeg zijn, het device kan kapot zijn of er is geen goede verbinding met het LoRaWAN netwerk.

Wat zijn de randvoorwaarden voor het installeren van een sensor?

- Redelijke kennis van Geo-ICT
- Toegang tot een geo-database
- Toegang tot een Node-RED server
- Toegang tot een LoRaWAN Gateway

Waar is de handleiding van sensoren te vinden?

Handleidingen van mogelijk te gebruiken sensoren zijn te vinden op de website van de fabrikant, en meestal ook op de pagina's van webshops waar deze sensoren te koop zijn.

Waar moet ik op letten bij het installeren of bevestigen van een device?

Allereerst zorg dat het device stevig bevestigd is op een plaats buitenbereik van het normale publiek. Zorg hierbij dat het device nog open gemaakt kan worden. Voorkom dat het device wordt vastgemaakt aan objecten die niet van jezelf zijn zonder toestemming van de beheerder. Dingen als bliksemgeleiders zijn niet geschikt, bij bliksem inslag zou het device kapot gaan.

Bijlage 1: Lijst met Dragino sensoren

Sensornaam	Meting van	Documentatie
LSN50 v2-D20	Temperatuur (°C)	Klik hier
LSE01	Bodemtemperatuur (°C) Volume vocht (V / V%) Microsiemens per cm (uS / cm)	Klik hier
LHT65N-E5	Relatieve luchtvochtigheid (%RH) Temperatuur (°C) Illuminatie (Lux)	Klik hier
LHT65N	Relatieve luchtvochtigheid (%RH) Temperatuur (°C)	Klik hier
LLDS40	Afstand (cm)	Klik hier
NLMS01	Bladvochtigheid (% vocht) Bladtemperatuur (°C)	Klik hier
NSPH01	PH-waarde	Klik hier
WSS-07	Fotonen inlicht (umol/m2/s)	Klik hier
WSS-06	Energie inhoud (W/m2)	Klik hier
WSS-05	Temperatuur (°C) Luchtvochtigheid (%RH) Illuminatie (Lux) Luchtdruk (hPa)	Klik hier
WSS-03	CO2 (ppm) Pm2,5 (ug/m3) Pm10 (ug/m3)	Klik hier
WSS-02	Windsnelheid (m/s) Windrichting (angel)	Klik hier
WSS-01	Regenmeter (mm)	Klik hier

Bron: Productenlijst Dragino