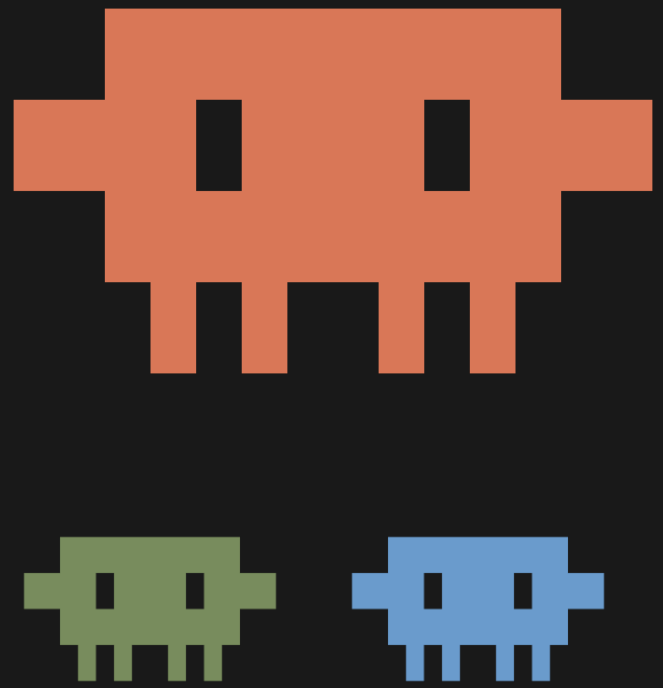
COMMUNITY OF PRACTICE · TYGRON AI 2026

Ik prompt niet meer. Ik geef doelen.



Hoe een team van AI-agents sensordata in de digital twin van Cartesius kreeg

Per Prins · Urban Sync

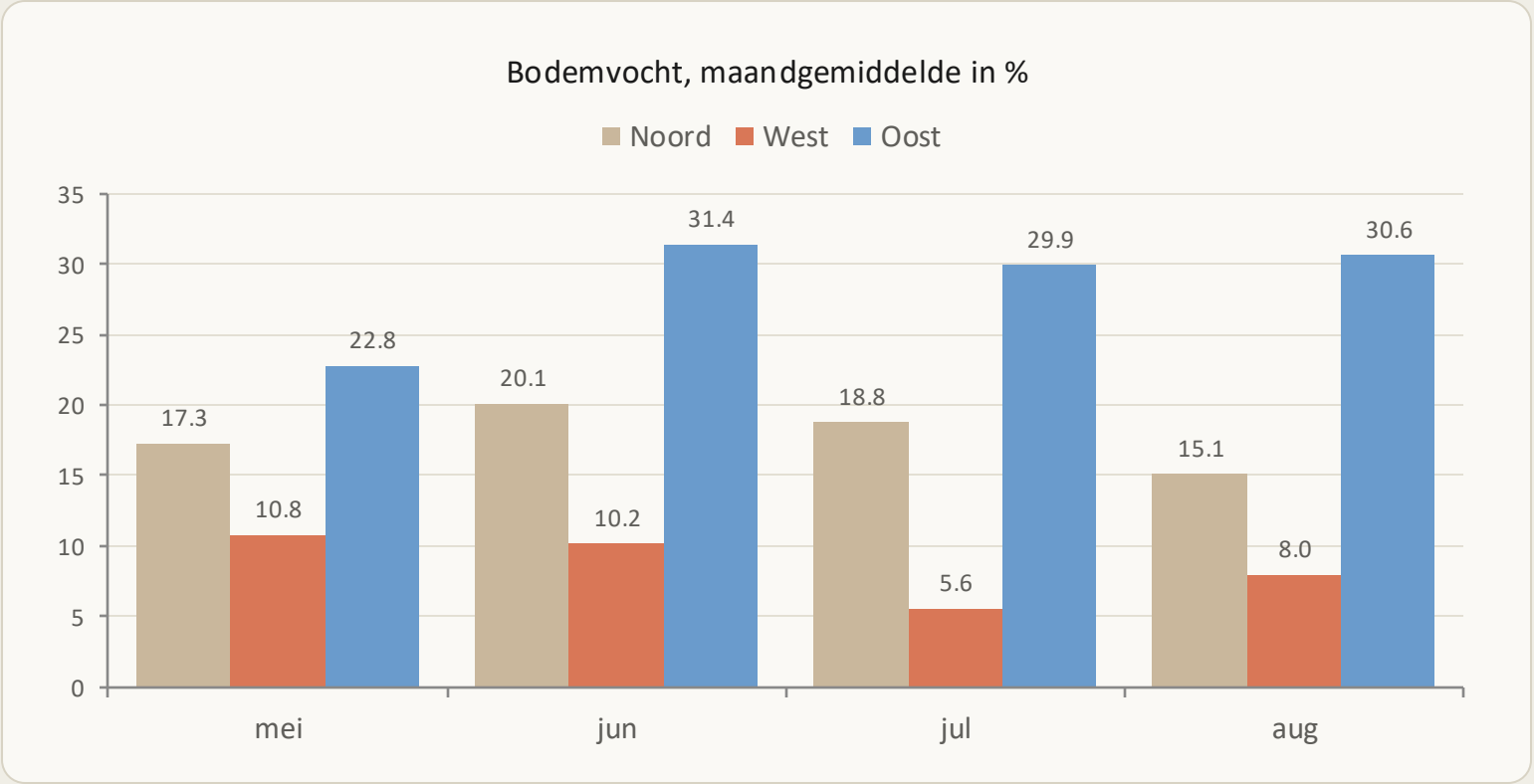
Drie sensoren, een factor vijf verschil

5,6%

bodemvocht bij Sensor West in juli

29,9%

bij Sensor Oost, zelfde maand, een paar honderd meter verderop



Pientere Tuinen via Good City Sense · mei is een deelmaand

De vraag Hoe krijg je de sensordata in de digital twin, zonder zelf te programmeren?

Van prompt naar doel

Vroeger: prompt engineering

“Je bent een ervaren Python-ontwikkelaar. Schrijf een script dat de OGC API van Good City Sense ophaalt. Let op de coördinaten in RD. Gebruik de sessie-API van Tygron. Als een waarde ontbreekt, … En vergeet niet om … Als het misgaat, probeer dan …”

Ik bedenk het hoe. Elke stap opnieuw.



```
> /goal

Sensordata van Good City Sense staat
elke maand in Tygron

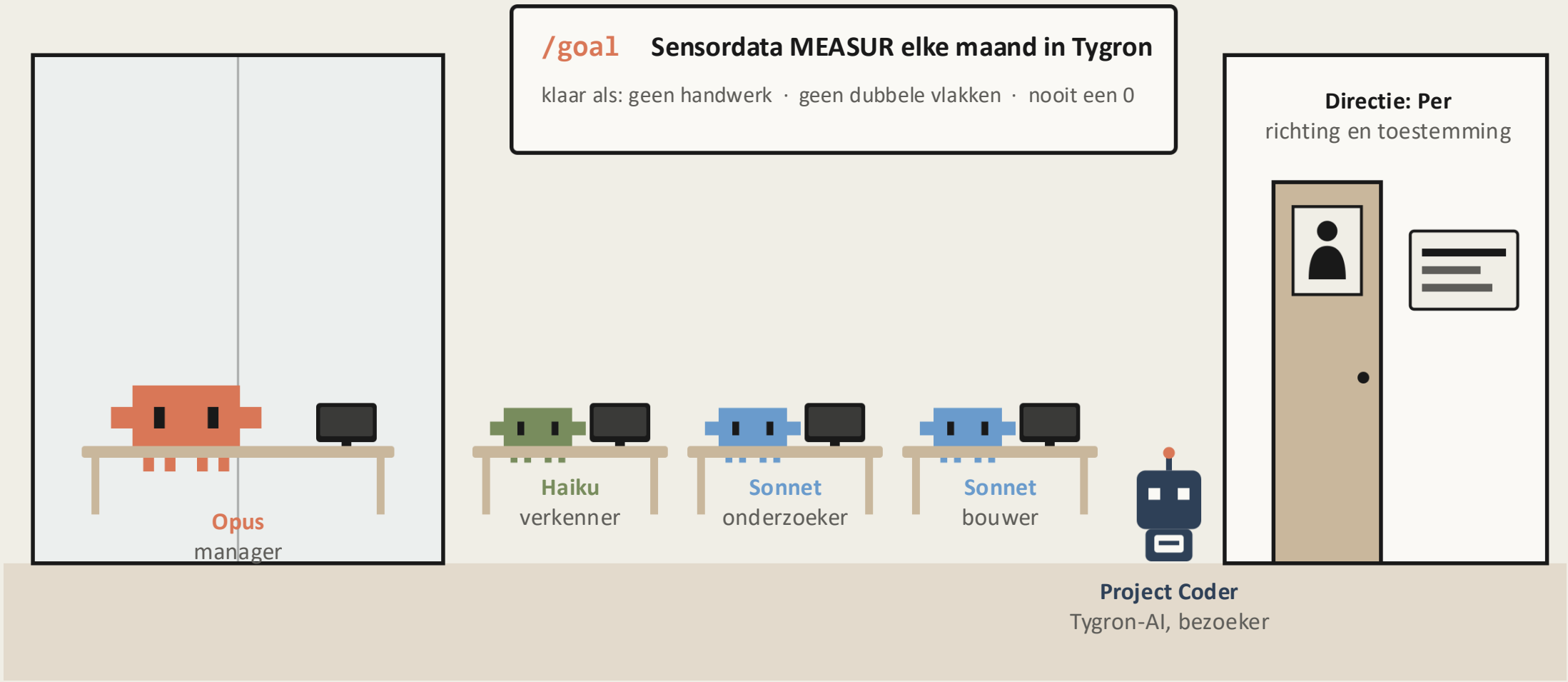
klaar als:

✓ nieuwe maand verschijnt zonder handwerk
✓ tweede run geeft geen dubbele vlakken
✓ ontbrekende meting wordt nooit een 0
```

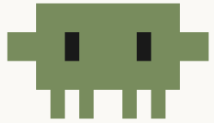
Ik zeg wat, en wanneer het klaar is.

Het team zoekt zelf het hoe, test tegen de ‘klaar als’ en stopt pas als het doel gehaald is.

Welkom op kantoor



Iedere collega heeft een functieomschrijving



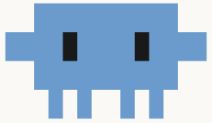
Verkenner

Haiku

Leest projectmap, mails en testrapporten. Vat samen.

skills zoeken, samenvatten

kosten €



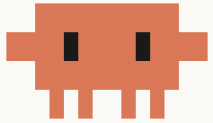
Onderzoeker / bouwer

Sonnet

Zoekt op het web, probeert API's uit, schrijft standaardcode.

skills API-test, Python

kosten €€



Manager

Opus

Plant, verdeelt, ontwerpt en lost vastgelopen problemen op.

skills orchestratie, review

kosten €€€



Project Coder

Tygron-AI, Qwen 27B

Voert stappen uit in het Tygron-project: overlays, panelen, areas.

skills platformkennis

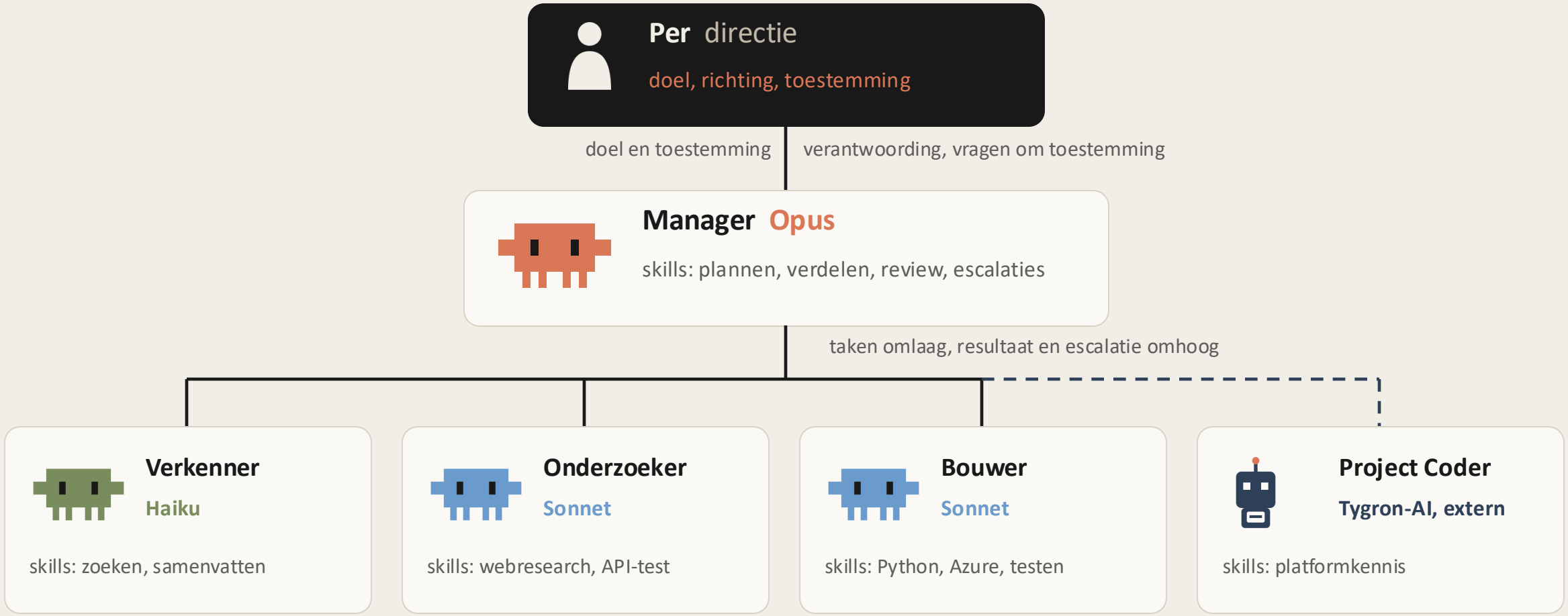
kosten **bezoeker**

```
---
name: verkenner
description: Leest de projectmap en vat samen.
model: haiku
---
```

Een functieomschrijving is een klein tekstbestand: naam, omschrijving, skills en model. De manager kiest op basis van de omschrijving wie het werk krijgt, en zet het goedkoopste model in dat de klus aankan.

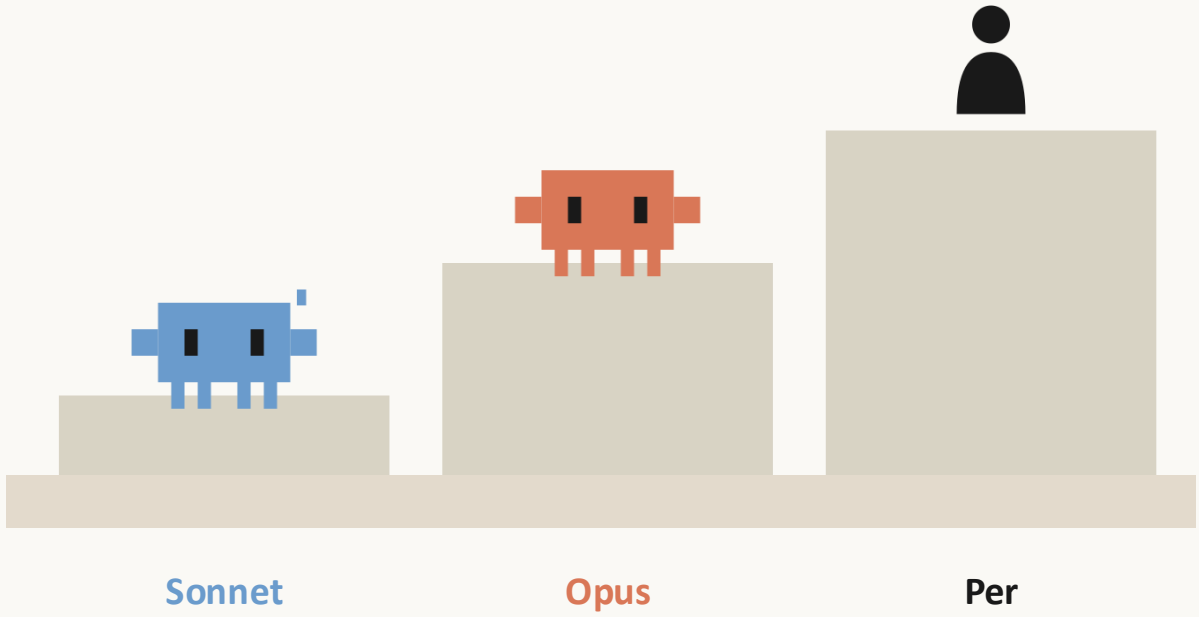


Wie verantwoordt aan wie



Vastgelopen? Dan escaleren

voorbeeld uit MEASUR



1

Een collega loopt vast

De import naar Tygron weigert de matrix uit de API-documentatie. Na een paar pogingen stopt hij en meldt het.

2

De manager neemt het over

Opus probeert een plat array per geometrie. Dat werkt. Hij legt het vast in de projectmap, zodat niemand er opnieuw over struikelt.

3

Alleen een keuze komt bij mij

Bijvoorbeeld: zetten we de Trigger live, met een sessietoken naar buiten? Dat beslis ik.

Ik kom alleen in beeld voor richting en toestemming

Komt bij mij

- Het doel en de ‘klaar als’
- Het sessietoken afgeven: dat is de sleutel van de sessie
- Een Trigger of Azure Function live zetten
- Kiezen tussen routes als er geen duidelijke winnaar is
- Alles wat niet terug kan

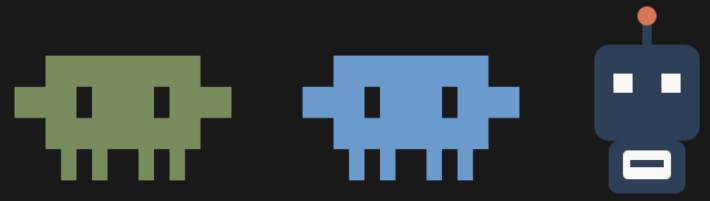
Regelt het kantoor zelf

- API's uitzoeken en uitproberen
- Python schrijven, testen en debuggen
- Alles vastleggen in markdown in de projectmap
- Opdrachten aan de Project Coder geven en het resultaat controleren
- Elkaars werk nakijken

DE USE CASE

MEASUR op Cartesius

Sensordata in de digital twin, gebouwd door het kantoor



DE USE CASE

Van sensor tot paneel, zonder handwerk

Sensoren
Pientere Tuinen, 3 stuks

→

Good City Sense
OGC API (pygeoapi)

→

Vertaalstuk
Python in een Azure Function

→

Tygron sessie-API
areas, overlay, panelen

→

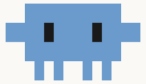
Digital twin
tijdbalk, dashboard, sensorpanelen

Trigger vóór elke herberekening roept het vertaalstuk aan: op update drukken en de nieuwe maand staat erin.



Verkennen

Mails, testrapporten en Tygron-documentatie lezen. Drie routes in kaart: Point Measurements, OGC API, Trigger direct. Alle drie vallen af.



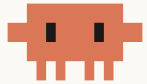
Bouwen en testen

Het vertaalstuk: ontdekt zelf maanden en grootheden, nooit dubbele vlakken, de nulregel: een ontbrekende meting wordt nooit een 0.



Uitvoeren in Tygron

Overlay op bodemvocht en een sjabloonpaneel per sensor, via een opdracht in gewone taal.



Controleren

Kijkt alles na, repareert wat halfaf bleef en legt de API-lessen vast in de projectmap.



De Tygron-AI als specialist van buiten



Opdracht aan de Project Coder Maak een overlay op bodemvocht en een sjabloonpaneel dat per sensor de waarden toont.

Ging goed

3 in plaats van 134

panelen. De AI bedacht zelf een filter op attribuut. Dat had ik niet voorgeschreven. Overlay en sjabloonpaneel ook zelf aangemaakt.

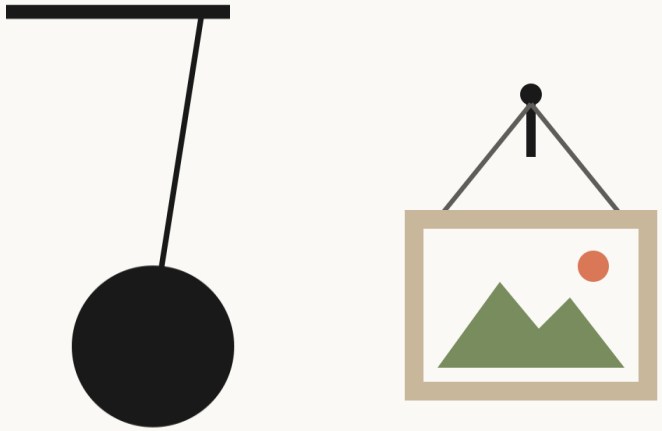
Bleef halfaf

10.000 tokens

budget op dit kanaal, toen was het op. Legenda en kleur bleven halfaf, en dat meldde de AI zelf niet. Opus zag het bij de controle en repareerde het via de sessie-API.

Les Opdrachten klein houden, en het werk van de specialist altijd laten nakijken.

Geen sloopkogel voor een schilderijhaakje



Een groot model voor een kleine klus is traag, duur en kost onnodig energie.

Taak	Wie
Projectmap doorzoeken, samenvatten	**Haiku** snel, goedkoop
API's testen, standaard Python	**Sonnet** het werkpaard
Plannen, ontwerpen, vastgelopen bug	**Opus** alleen als het moet
Stappen binnen het Tygron-project	**Tygron-AI** kent het platform

Tygron kiest hierom voor kleine, gespecialiseerde modellen. **Maxim vertelt waarom.**

Wat werkt, wat schuurt, en de vangrails

Werkt

- Alles in markdown in de projectmap: elke sessie begint met context
- Twee AI's vangen elkaars fouten op
- Herhaalbaar: een tweede run geeft geen dubbele vlakken

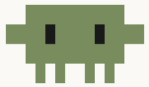
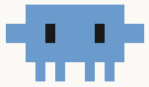
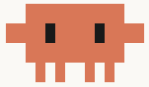
Schuurt

- API-documentatie is onvolledig: veel proefondervindelijk werk
- Tokenlimiet van 10.000: grote opdrachten opknippen
- Refresh haalt zelf niets op: de Trigger is de schakel

Vangrails

- Het sessietoken is de sleutel: nooit in een slide of mail
- AI plus Triggers betekent internettoegang: bewust kiezen
- Toestemming voor alles wat niet terug kan

Deze presentatie kwam ook van kantoor

	**Haiku**	las projectmap 221 en vatte de use case samen
	**Sonnet**	zocht de beschrijving van deze sessie op tygron.com
	**Opus**	schreef de rode lijn, tekende het kantoor en bouwde deze slides
	**Per**	koos het concept en gaf de opdracht om te bouwen

Kosten

Het zoekwerk liep op de goedkope modellen. Opus alleen voor ontwerp en opbouw.

Mijn inbreng

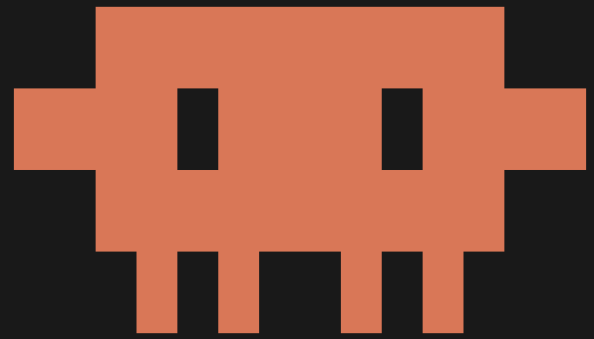
Twee berichten: een doel, en een keuze.

Zelfde werkwijze als in MEASUR, ander doel.



LIVE

Geef doelen, geen prompts.



```
> /goal ...
```

We geven samen één doel en kijken mee in de webviewer van Tygron, in 2D en 3D.

Daarna: Frank over skills van agents in Tygron