

# Nederlandse Notificatie Strategie

4<sup>e</sup> community bijeenkomst

9 september 2021

[Pleio website](#)

# Agenda

1. Recap
2. Update praktijkbeproevingen
3. Update berichtenstandaard
4. Afronding

# Recap

Samenvatting van concepten / essentie





Een berichtje sturen

Elkaar informeren

Een seintje geven

## Wat is notificeren?

Vertellen wat er  
(zoiest) gebeurt is

Aangeven dat iets  
gestart kan worden

Een opdracht geven

Een burger of bedrijf informeren over  
een besluit of statusverandering

Definitie Nederlandse Notificatie Strategie:

Zowel 'push'  
als 'pull'.

"Het door een aanbieder aan afnemers **verstrekken** van gegevens  
over **binnen zijn domein plaatsgevonden gebeurtenissen**"

Vraagt om goede  
analyse en overleg  
met afnemers.

Het is gebeurd en kan  
niet meer veranderen.

i.v.m. tijd

Inperking scope:

- *G2G: Overheid naar overheid*
- *M2M: Machine naar machine*
- *1:N: Van één aanbieder naar meer afnemers*
- *Gebeurtenissen met en zonder gegevenswijziging*
- *Open en gesloten data*

...



# Gebeurtenis

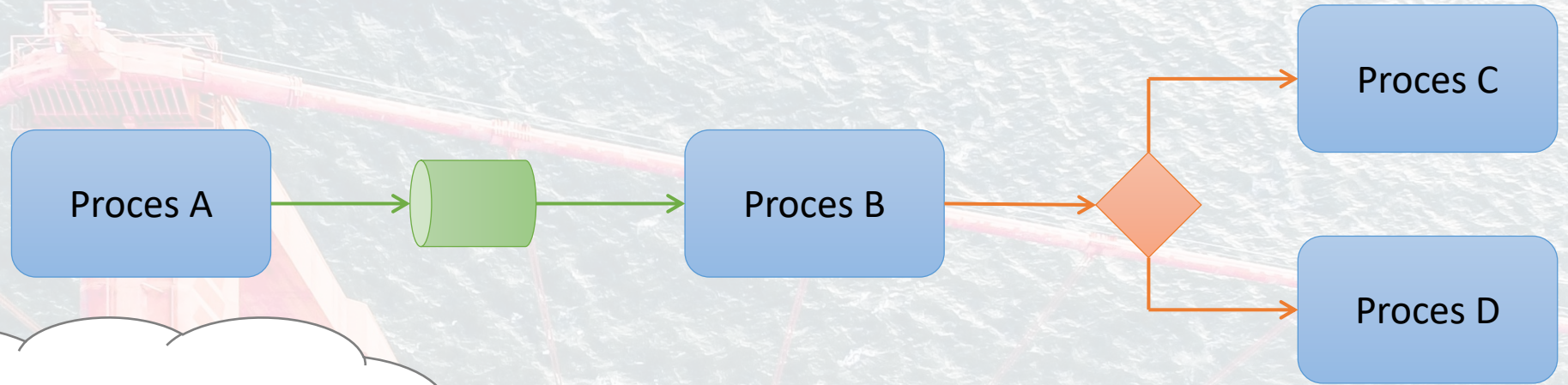
Bevragen bij  
de bron

<b>Gebeurtenis + identificatie (Informatiearm)</b>	Informatiearme gebeurtenis. Gegevens later/apart ophalen.	Verhuizing met in bericht BSN's. Nieuwe adres etc via aparte bevraging.
<b>Gebeurtenis + gegevens (Informatierijk)</b>	Gebeurtenis is het uitgangspunt. Gegevens horen bij gebeurtenis.	Verhuizing van gezin i.p.v. van aparte berichten per gezinslid.
<b>Gegevensmutatie + Aanleiding</b>	Oude & Nieuwe gegevens + Aanleiding. (Soort gebeurtenis)	Oude en nieuwe adres = Verhuizing Oude en nieuwe adres = Wijziging Straatnaam
<b>Gegevensmutatie</b>	Bij verandering: Oude & Nieuwe gegevens	Oude en nieuwe adres
<b>'Seintje'</b>	Seintje als er iets gebeurt/verandert	Internet of Things (Sensoren...)
Interpretatie concept	Toelichting	Voorbeeld



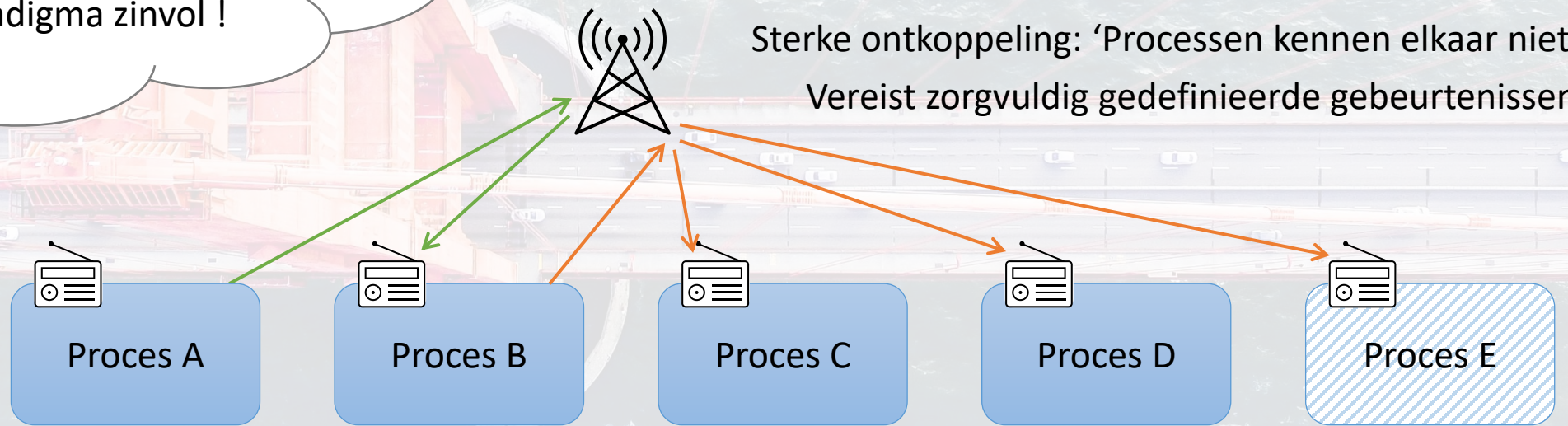
# 'Event driven' paradigma

## Workflow



Notificeren over gebeurtenissen is ook zonder adoptie van het event driven paradigma zinvol !

## Event Driven



Sterke ontkoppeling: 'Processen kennen elkaar niet'.  
Vereist zorgvuldig gedefinieerde gebeurtenissen.

# Praktijkbeproevingen



# Update: Beproeving gemeente Súdwest Fryslân

## Enkele relevante gebeurtenissen

1

Nieuw informatie ontvangen  
(document/email)

- Medewerker uit dienst
- Onderzoek omgevingsdienst afgerond
- Bedrijf failliet
- Bedrijf verhuisd

3

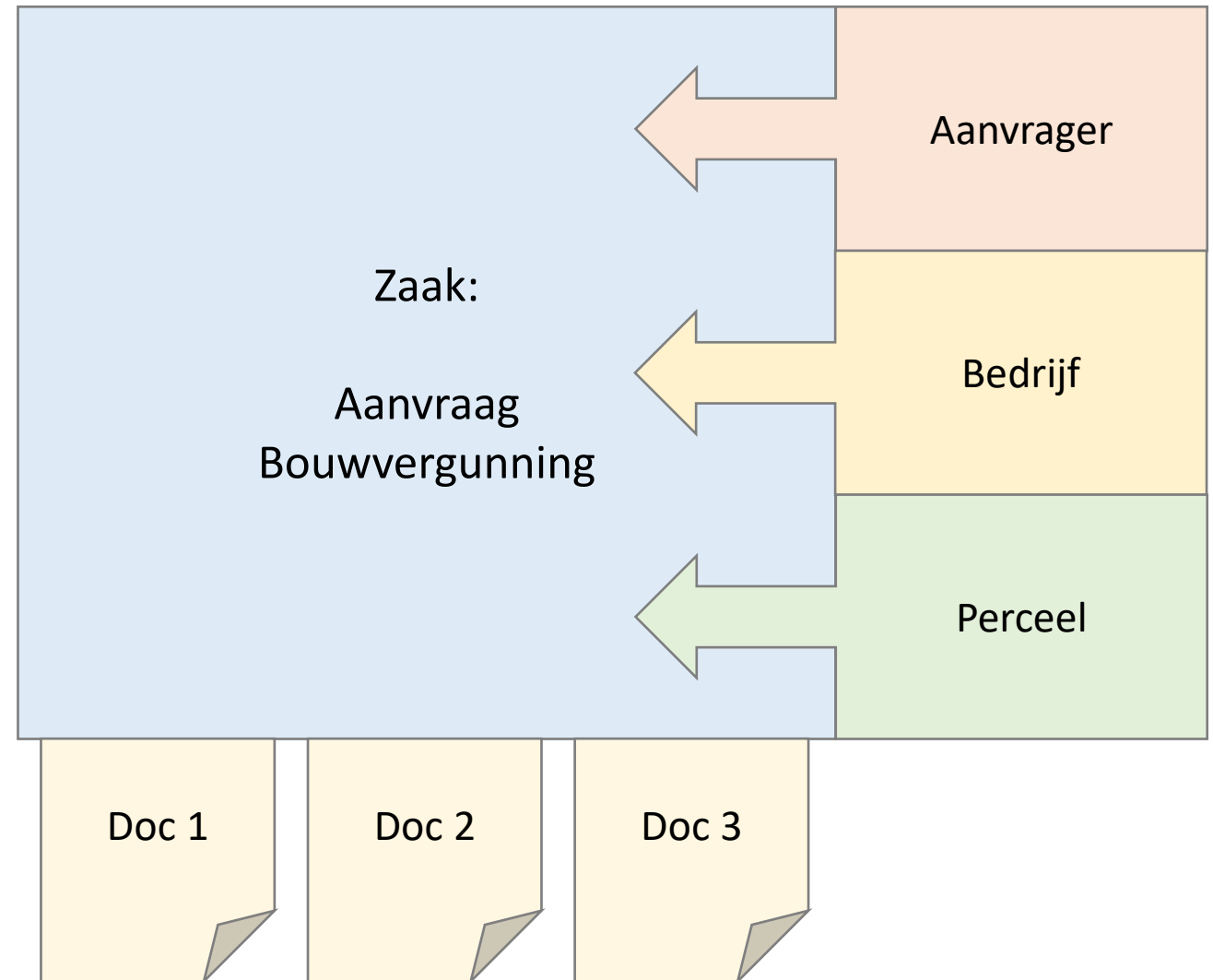
Betrokken persoon overleden

- Eigendom perceel overgedragen

...

2

Statusverandering publiceren  
(Te gebruiken voor 'XML generator'  
voor MijnOverheid lopende zaken)



# Update beproeving Logius

- Processimulatie
- Voorstel ingediend bij Routekaartoverleg (RKO) om Q4 een PoC te bouwen
  - Van BRP mutatieberichten naar notificatie volgens de conceptstandaard te leveren via REST

Berichtenstandaard



# Adoptie Cloud Events standaard

- We proberen afwijkingen te voorkomen. We proberen uitbreidingen te minimaliseren
- We beperken ons tot generieke functionaliteit. De domeinen bepalen de inhoud
- Functionele standaard is protocol onafhankelijk. Cloud Events biedt wel bindings voor allerlei protocollen zoals HTTP, REST, AMQP, MQTT, etc.
- We laten meerdere mechanismen toe zoals Push/Pull, Synchron/Asynchron
- Breed gebruik moet mogelijk zijn. Van eenvoudige notificatie tot event sourcing
- We minimaliseren de afhankelijkheid van IT-voorzieningen
- Als geen vrije interpretatie gewenst is, proberen we deze te voorkomen.



Complexity is the enemy -  
Roger Sessions

Perfect  
Simple Plan

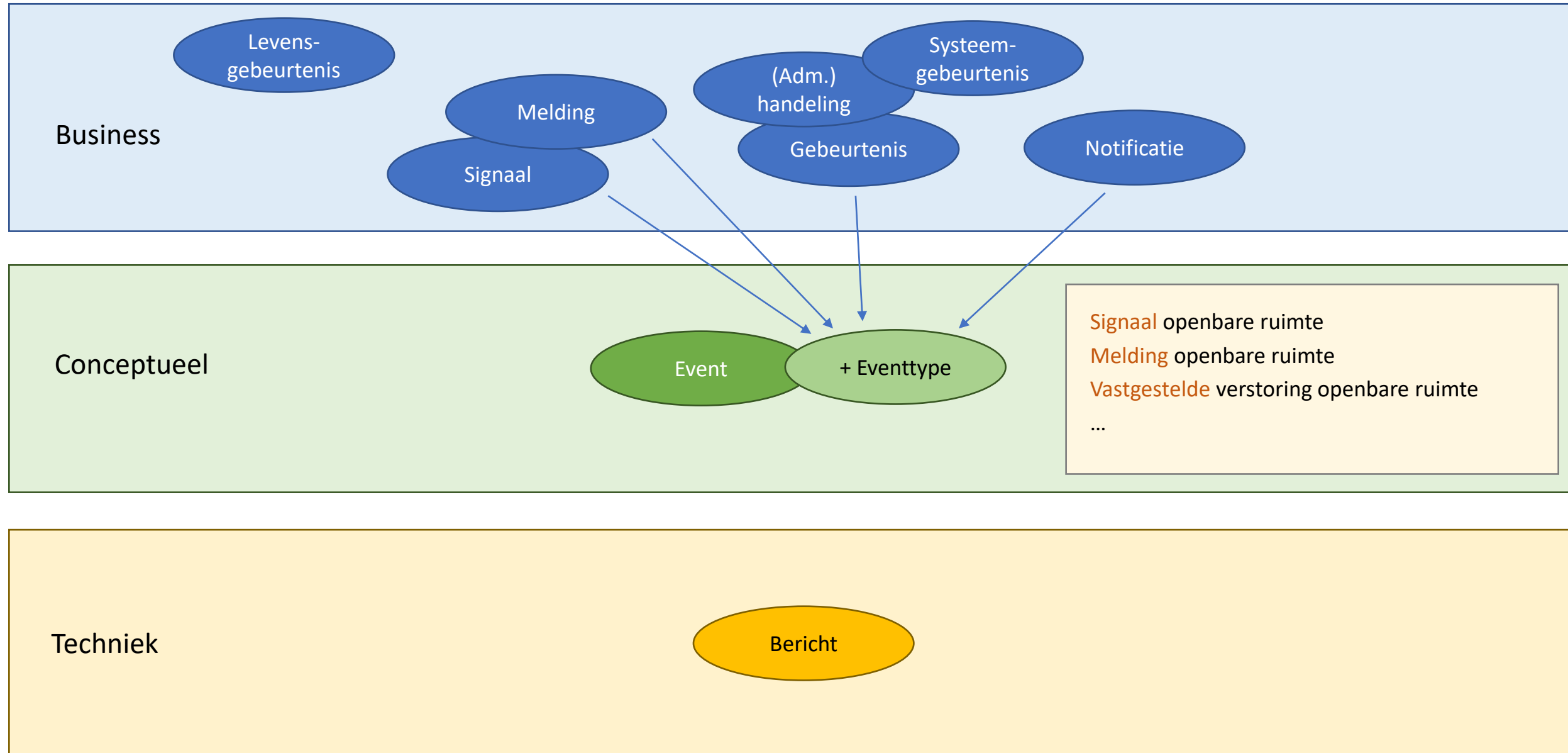


#### NIEUW NAAST OUD – We ontwikkelen nieuw naast bestaand

We ontwikkelen nieuwe voorzieningen conform de Common Ground-visie, maar houden er rekening mee dat bestaande voorzieningen nog 5 tot 10 jaar in gebruik zullen blijven. Bij nieuwe gemeentelijke en landelijke ontwikkelingen zijn de Common Ground-principes leidend. We steken energie in het bouwen van iets nieuws naast het oude. Als 'nieuw' en 'oud' elkaar ontmoeten, moeten oude voorzieningen gaan aansluiten op de nieuwe voorzieningen, eventueel na aanpassing daarvan. Wat nu als 'nieuw' wordt bestempeld, blijft uiteraard niet altijd 'nieuw'. Er zullen continu ontwikkelingen zijn die het nodig maken om te blijven vernieuwen. We formuleren daarom geen eindplaatje en doelarchitectuur, maar zien ontwikkeling als een continu proces ('permanent beta').

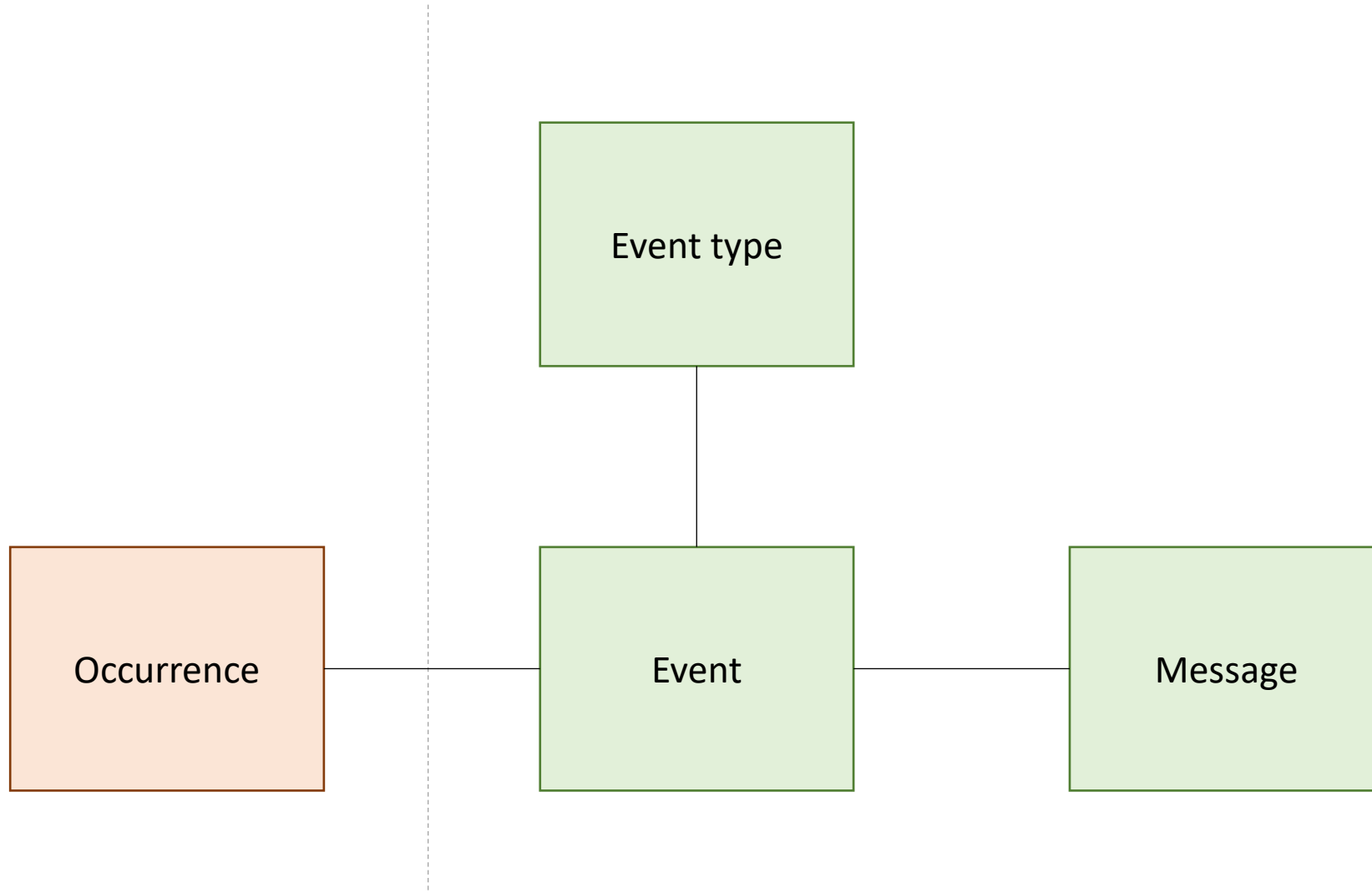
#### Implicaties:

# Vertaling van business events naar cloud events

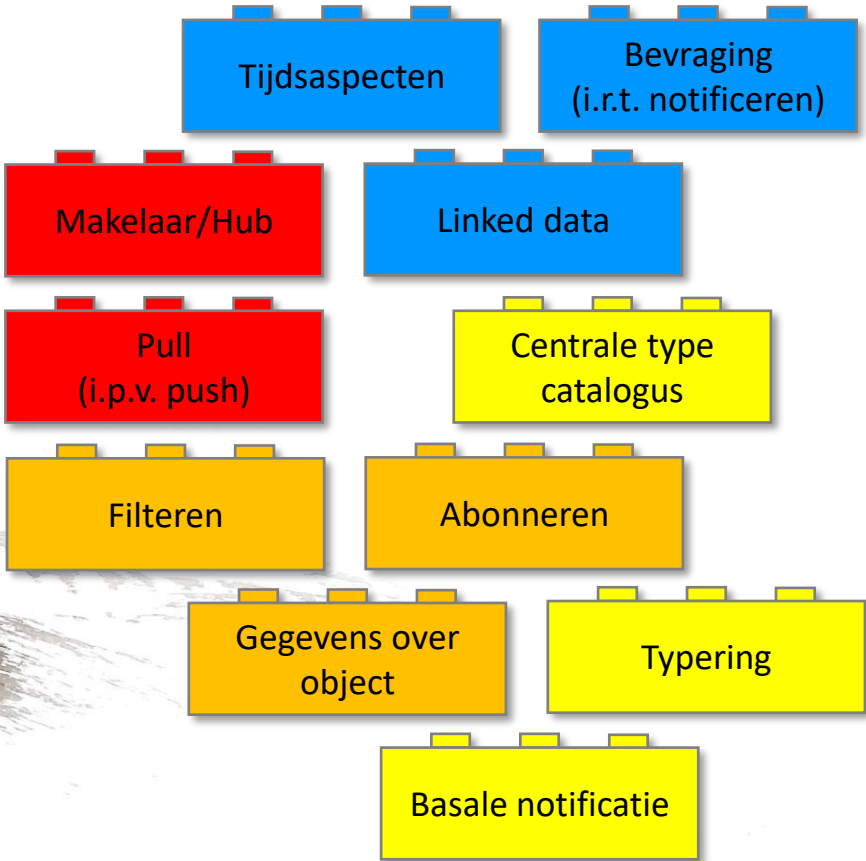
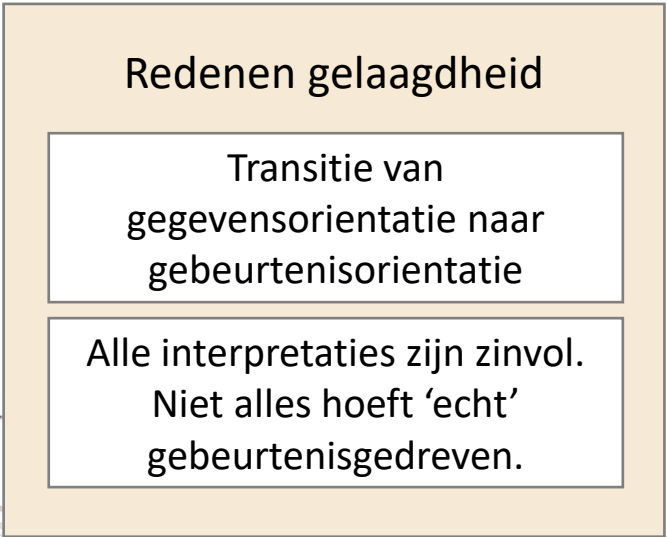
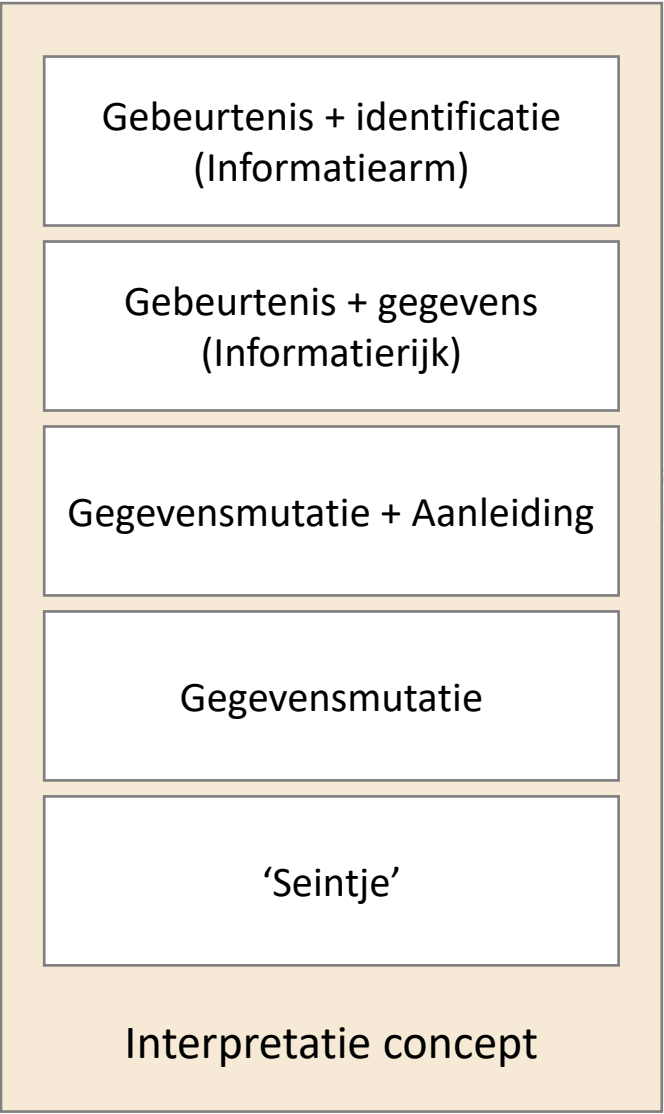




# Conceptueel



# Gelaagde standaard

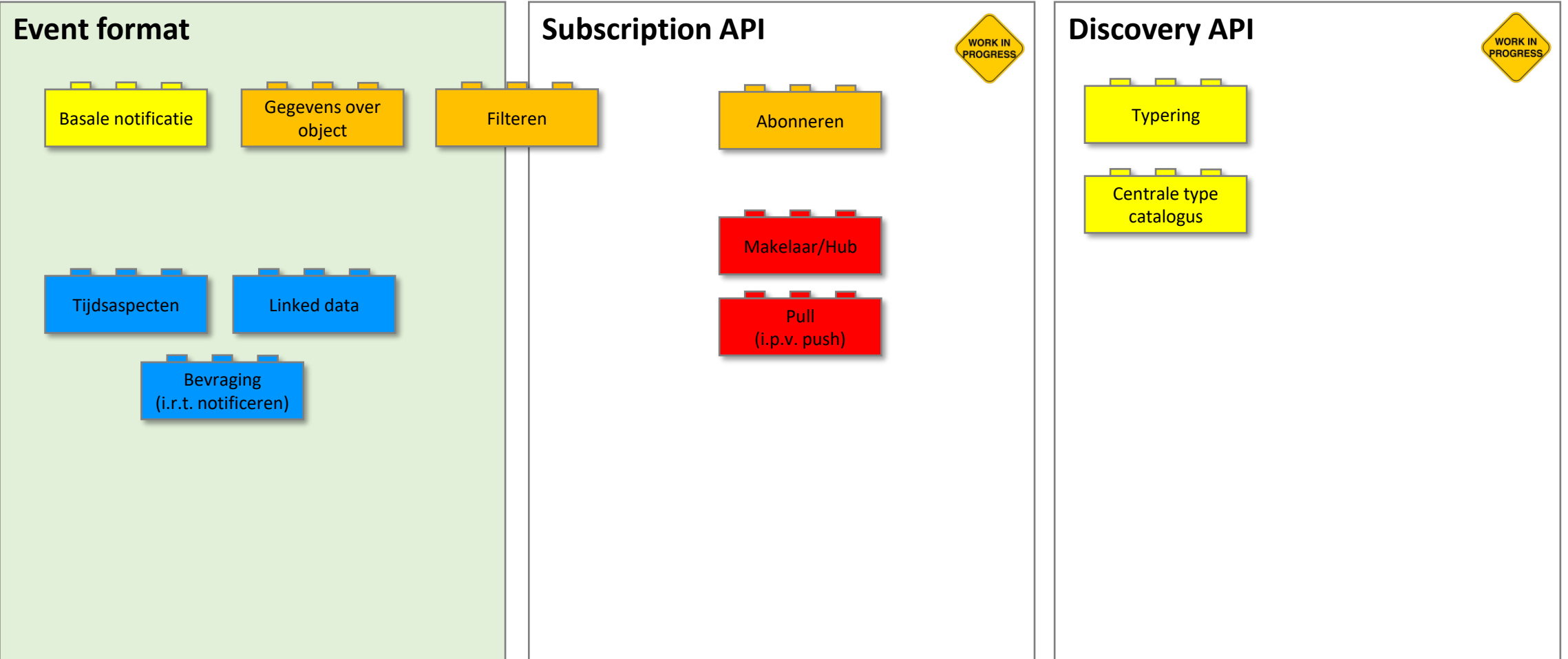


# Cloud events adoptie: Event attributen en extensions





# Cloud events adoptie: Verdere standaardisatie



# Conceptstandaard voor RESTful notificeren

1. Cloud events is protocol neutraal
2. Cloud events kent meerdere 'protocolbindings' waaronder een binding voor HTTP/REST

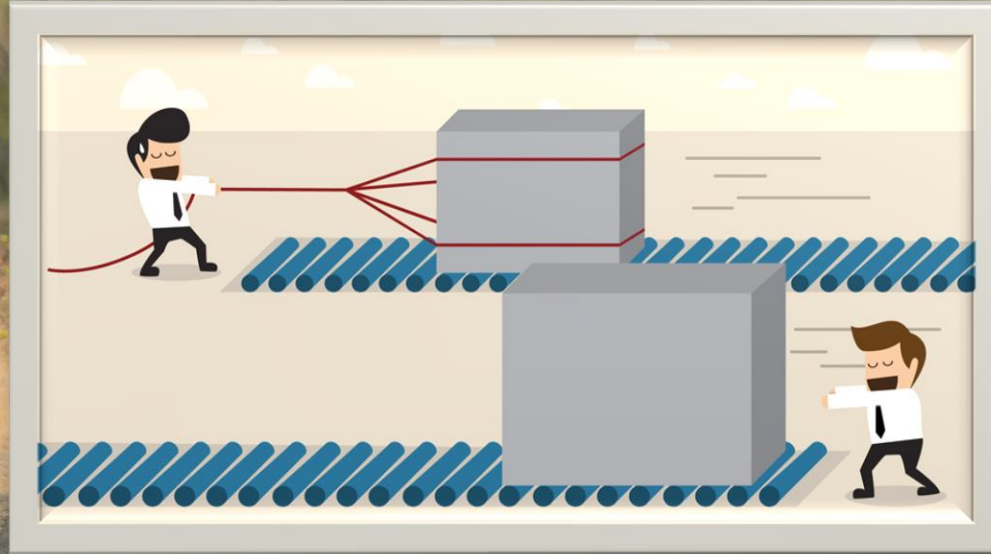


Waarom concept standaard voor RESTful?

Hoe te komen tot een concept standaard?

1. Onderzoeken of de REST implementatie van Cloud events bruikbaar is
2. Controleren of alle concepten uit ZGW Notificeren ondersteund kunnen worden (Conceptuele backwards compatibility).

# Conceptstandaard voor RESTful notificeren



Hoe belangrijk is 'RESTful pullen van events' ?

Nu en in de toekomst ?

Moeten we het (nu) willen standaardiseren ?

Afsluiting / Q&A