

DIDO – Dienstverlening Doelarchitectuur

Introductie

Concept, 27 september 2025



DIDO – Dienstverlening Doelarchitectuur

Introductie

Met dit soort tekstvakken geven we een toelichting op de architectuur.

Dit is een eerste concept. Bedoeld om doel, scope en inhoud af te kunnen stemmen met de Common Ground community.

Doel en scope

Op welk 'doel' richt de architectuur zich?

Tot welke scope beperkt de architectuur zich?

Vakapplicatie /
Taakspecifieke applicatie

interactie

proces

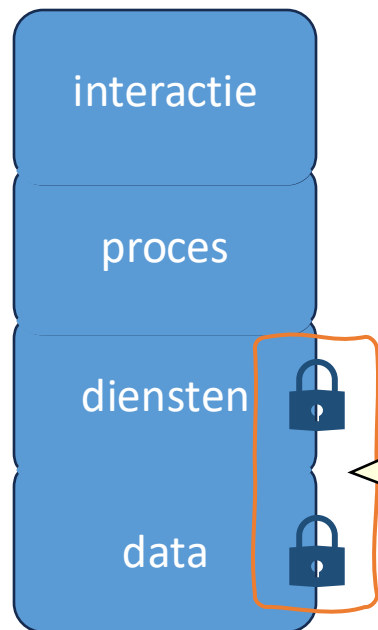
diensten

data

We werken binnen de gemeenten met meer dan 1000 verschillende applicaties.

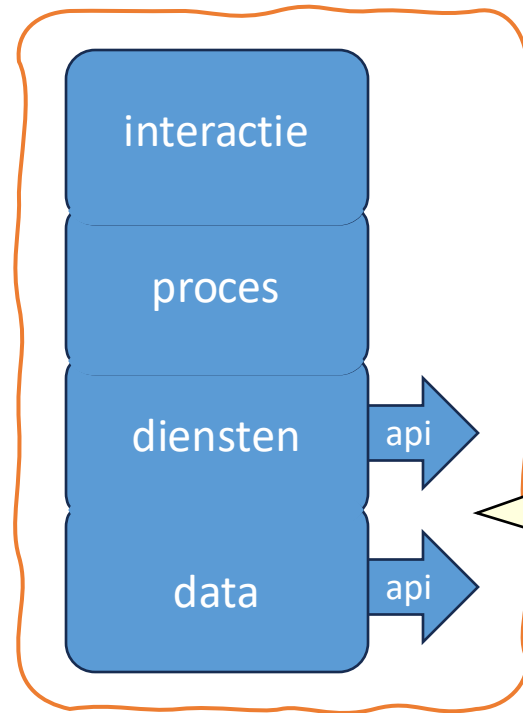
Meestal zijn dit 'klassieke' applicaties die voor een bepaald vakgebied of een bepaalde taak een oplossing bieden.

De applicatie biedt een 'totaaloplossing'. Van gebruikersinterface tot gegevensopslag.



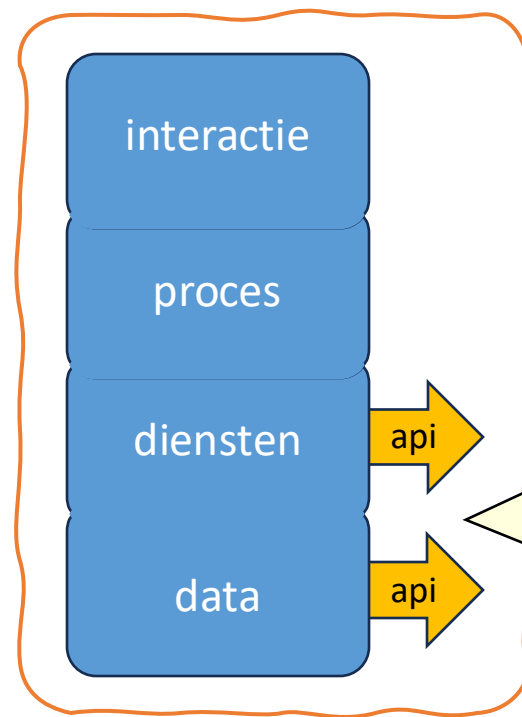
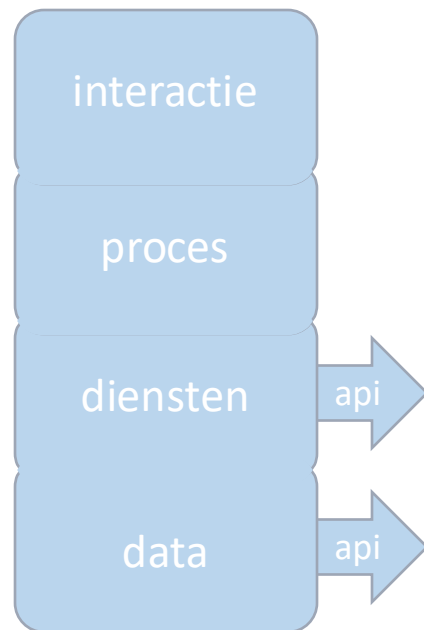
Lastige is dat we als gemeenten vaak niet bij de gegevens van de applicatie kunnen.

Dat is wel nodig. Bijvoorbeeld om gegevens te tonen via een portaal, in te zien via een klantcontactcentrum, of om ze duurzaam te bewaren.



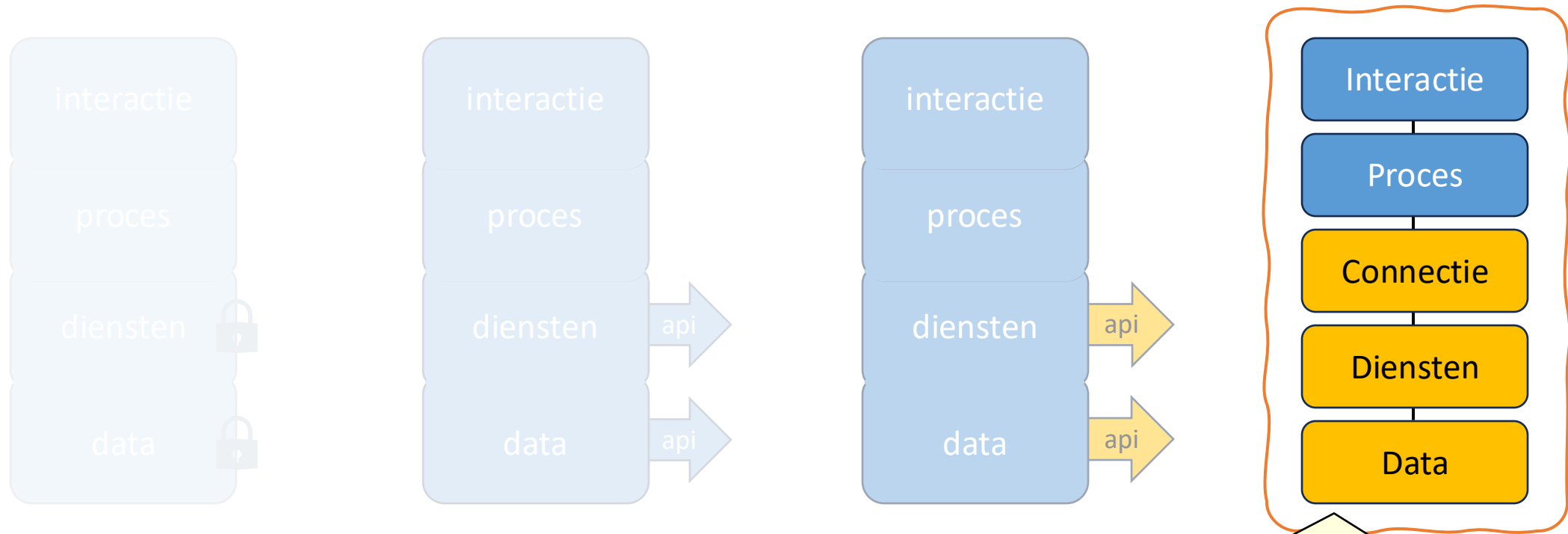
Een eerste stap is dat de leverancier de gegevens via een API toegankelijk maakt.

We hebben dan toegang, maar moeten voor iedere applicatie de begrippen en structuur van die applicatie vertalen naar de begrippen en structuur die we nodig hebben voor inzage of archivering.



Het wordt een stuk realistischer als we met standaard APIs werken.

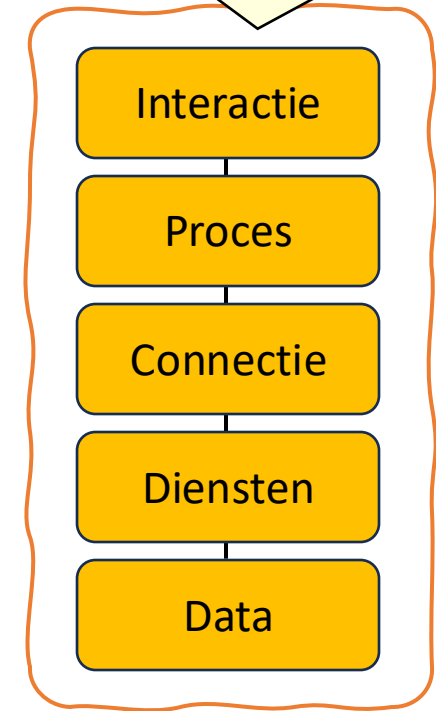
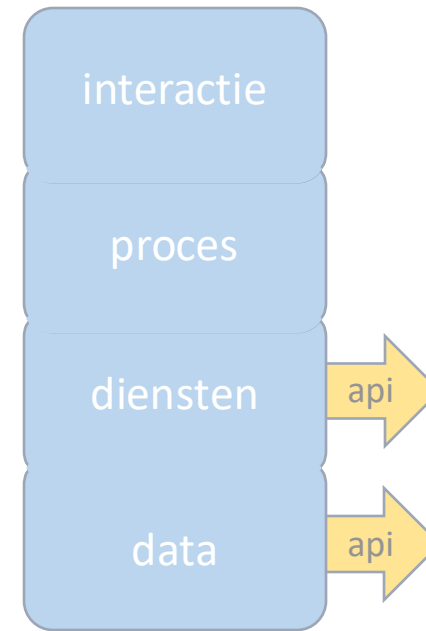
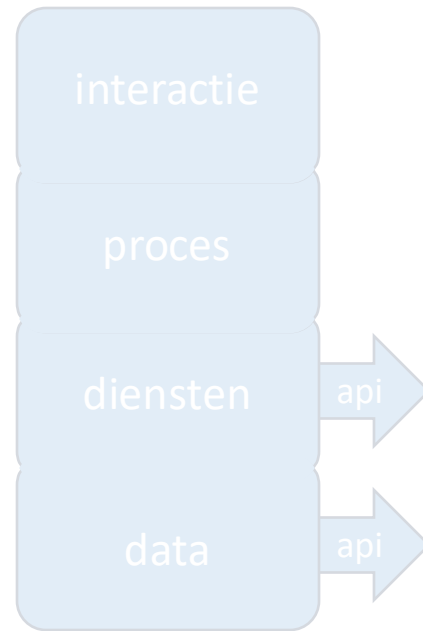
De leverancier vertaalt de interne begrippen en structuren dan naar de taal en structuur van de API.

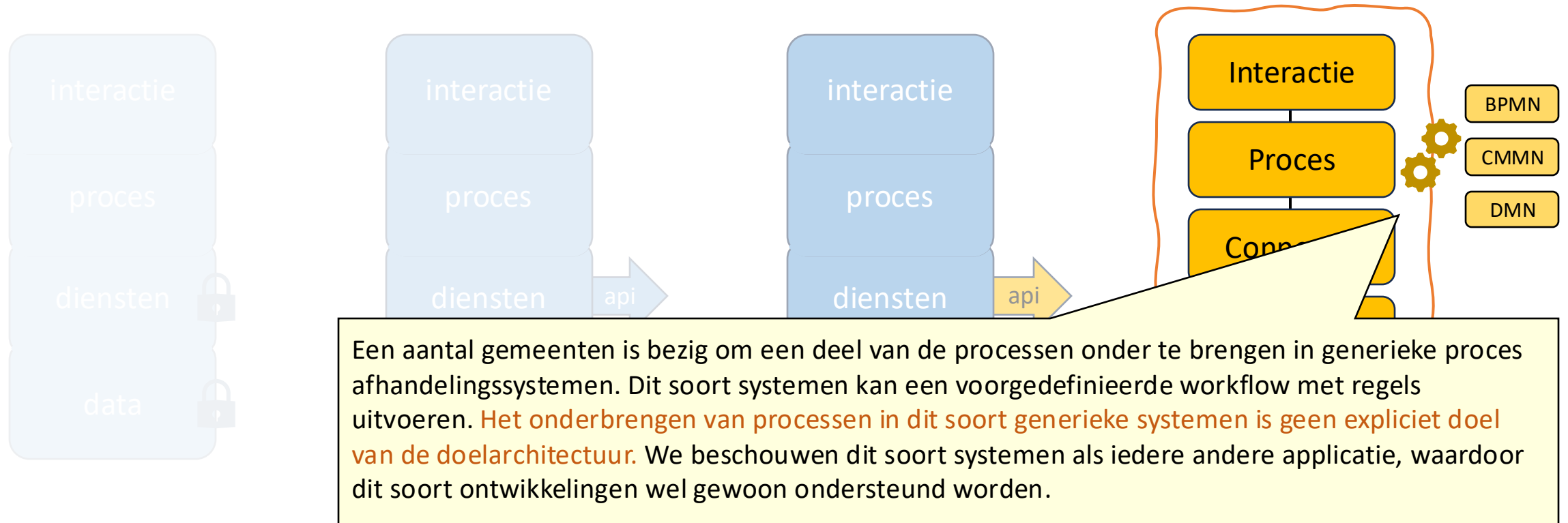


Uiteindelijk werken we toe naar scheiding van processen en gegevens.

Applicaties bieden de gebruikersinterface, de interactie en het proces.
Registers regelen de opslag en ontsluiting van de gegevens.

Het is denkbaar dat de interactie en processen verder geüniformeerd worden. Dat valt buiten de scope van deze doelarchitectuur.





Vakapplicatie

interactie

proces

diensten

data

Processen en gegevens van elkaar
scheiden is nog niet zo eenvoudig.

Vaak zitten er in een applicatie allerlei
soorten gegevens.

Vakapplicatie

interactie

proces

diensten

data

BAG

BRP

HR

...

landelijke registers

Zo hebben we gegevens uit landelijke registers die we nog vaak dupliceren.

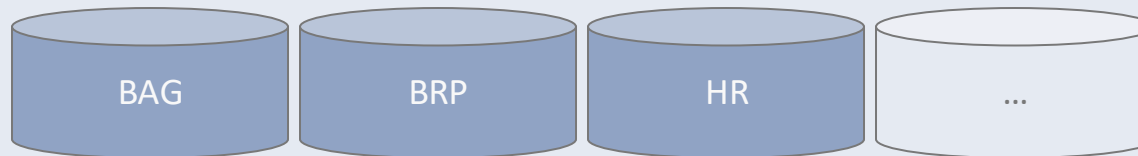
Vakapplicatie

interactie

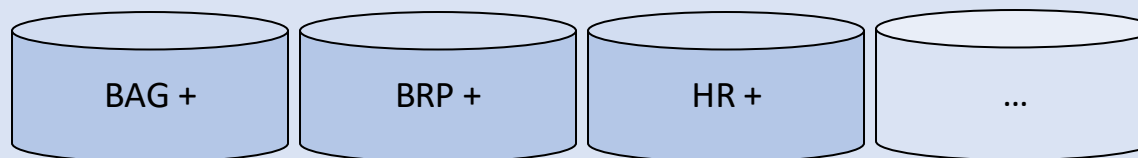
proces

diensten

data



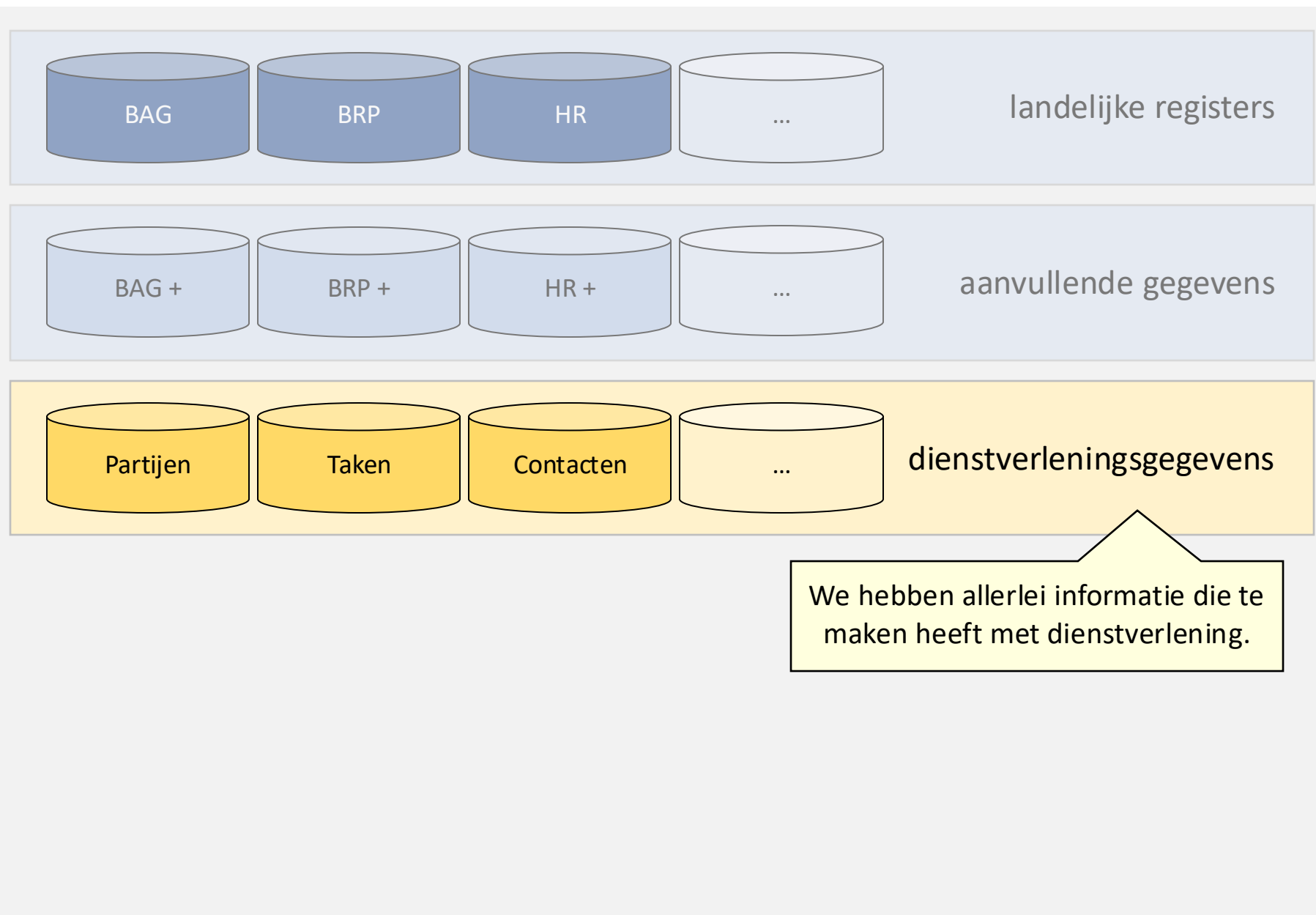
landelijke registers



aanvullende gegevens

We hebben aanvullingen die we niet in de landelijke registers kwijt kunnen.

Vakapplicatie



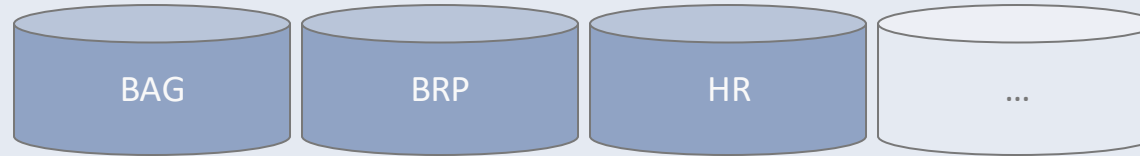
Vakapplicatie

interactie

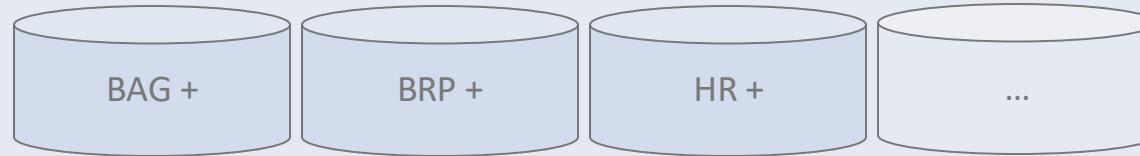
proces

diensten

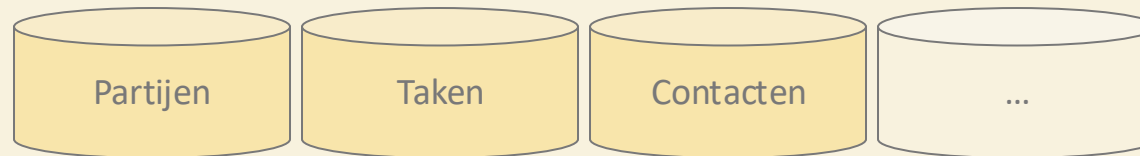
data



landelijke registers



aanvullende gegevens



dienstverleningsgegevens

niet gestandaardiseerd
vakgebied



We hebben de gegevens die specifiek zijn
voor het vakgebied.

Voor de meeste vakgebieden hebben we nog
geen standaard gegevensmodel en zeker
geen standaard registers. Applicaties regelen
daarom zelf de opslag.

Vakapplicatie

interactie

proces

diensten

data



landelijke registers



aanvullende gegevens



dienstverleningsgegevens

niet gestandaardiseerd
vakgebied



gestandaardiseerd
vakgebied

Het ideaal is een register
voor ieder vakgebied.

Vakapplicatie

interactie

proces

diensten

data



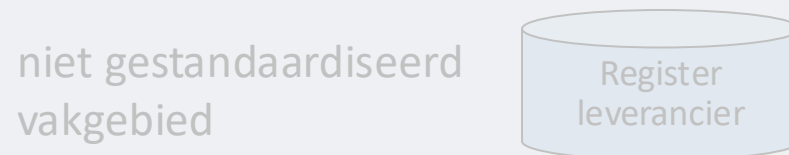
landelijke registers



aanvullende gegevens



dienstverleningsgegevens



niet gestandaardiseerd
vakgebied

Register
leverancier



gestandaardiseerd
vakgebied

Register
Vakgebied

overige
procesgegevens



Proces-
gegevens

Maar ook met standard registers voor vakgebieden, zullen applicaties nog steeds eigen procesgegevens hebben.

Vakapplicatie

interactie

proces

diensten

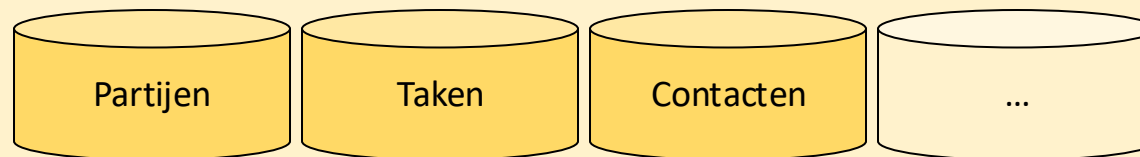
data



landelijke registers

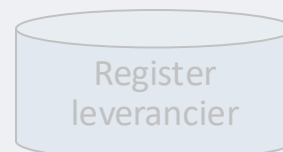


Voor wat betreft de gegevens, beperken we ons in de doelarchitectuur, tot de dienstverleningsgegevens.



dienstverleningsgegevens

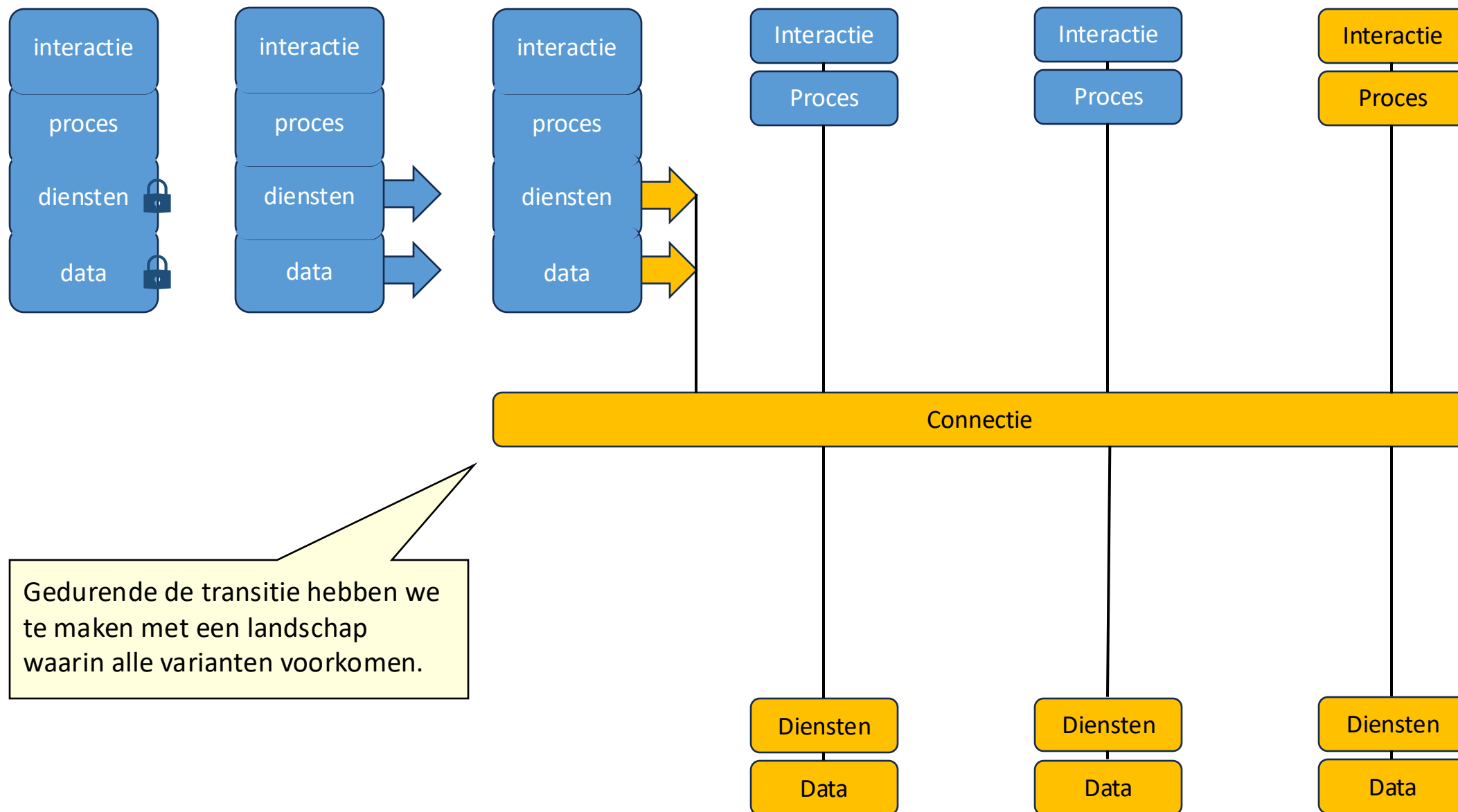
niet gestandaardiseerd
vakgebied

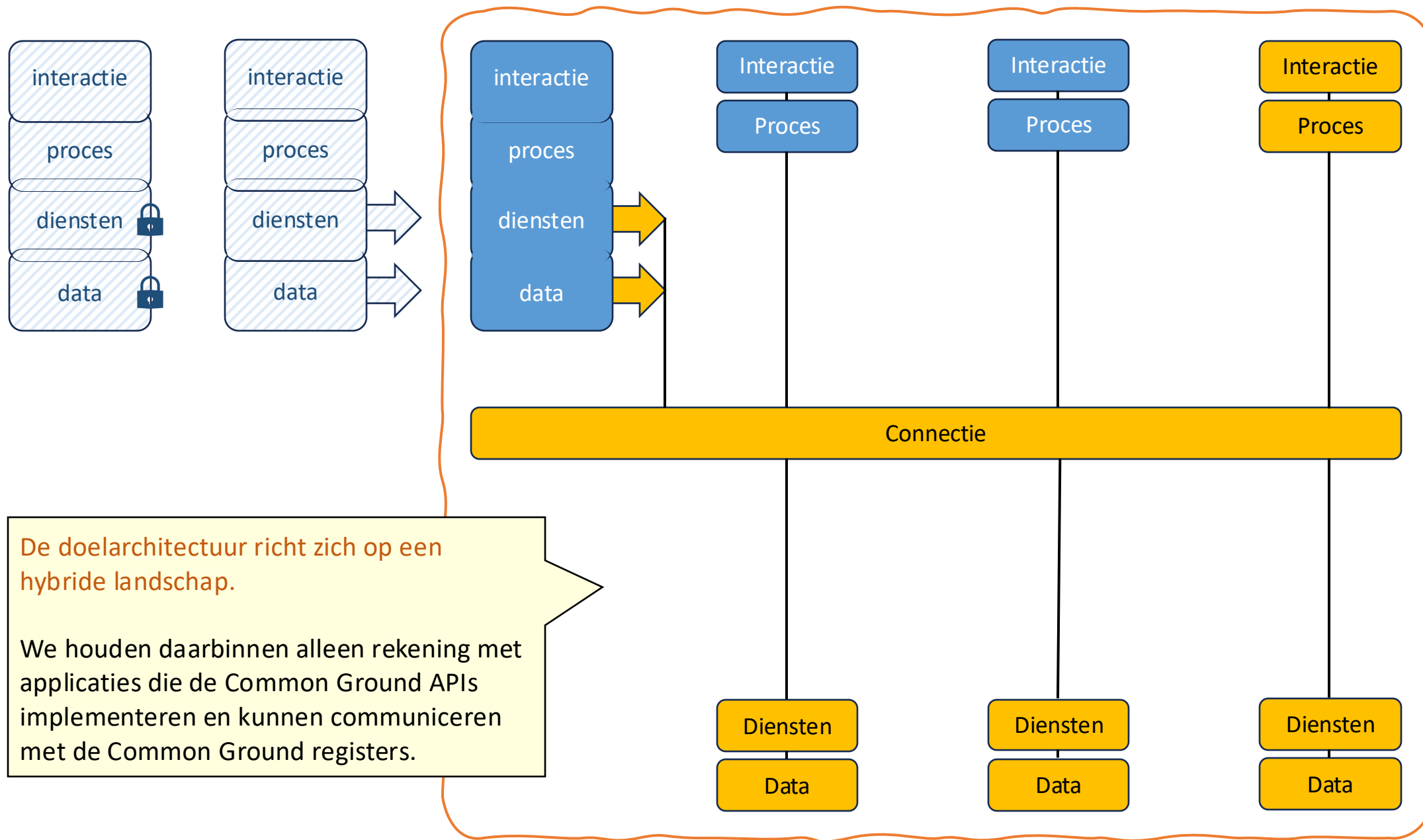


gestandaardiseerd
vakgebied

overige
procesgegevens







Generieke Dienstverleningselementen (GDE's)

De onderwerpen van de doelarchitectuur.

Hoe bepaal je met zoveel verschillende soorten applicaties de gemeenschappelijke elementen?

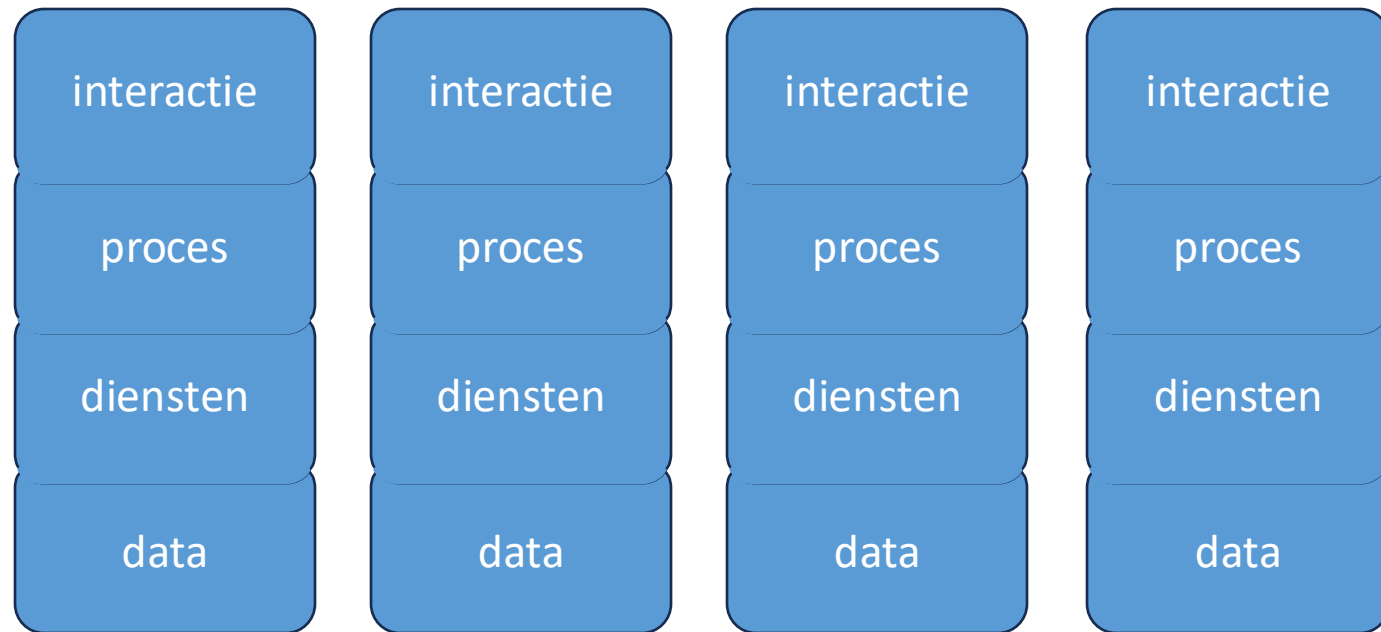
interactie

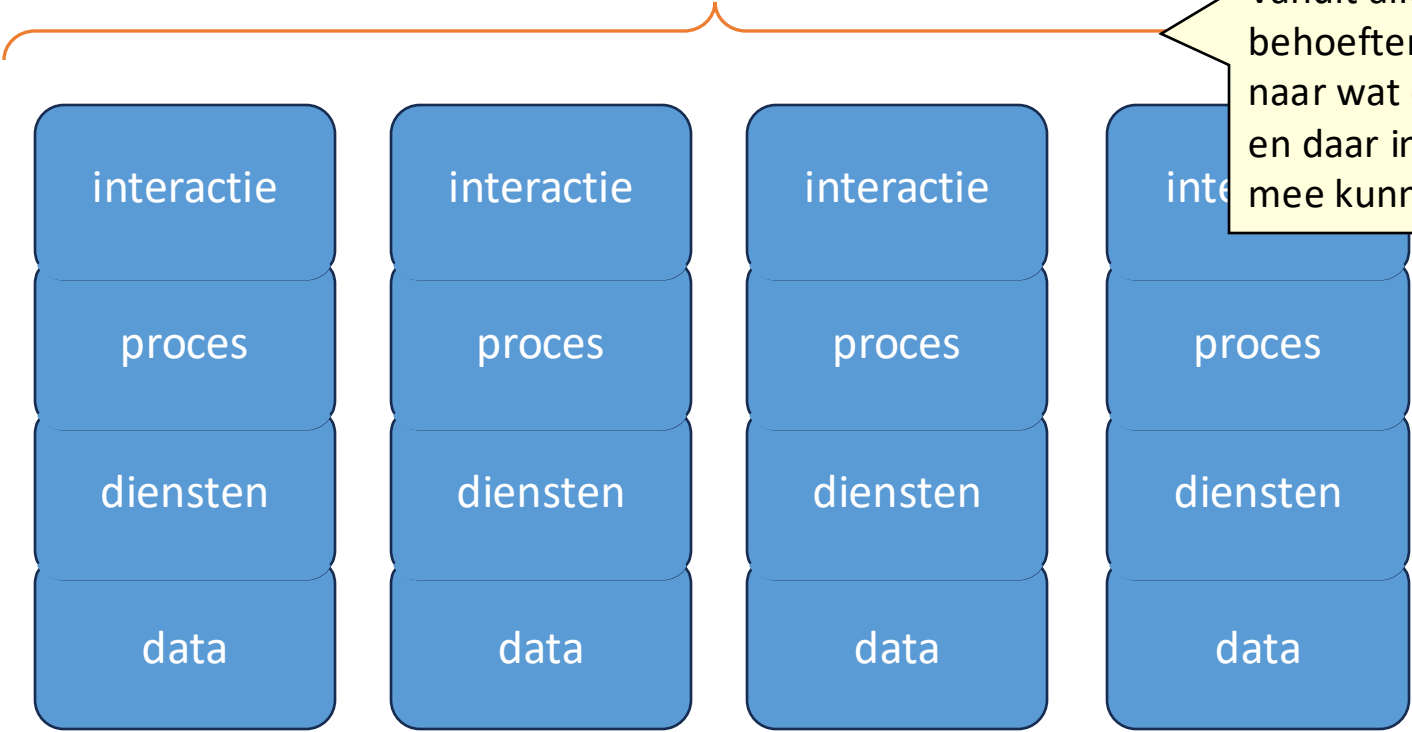
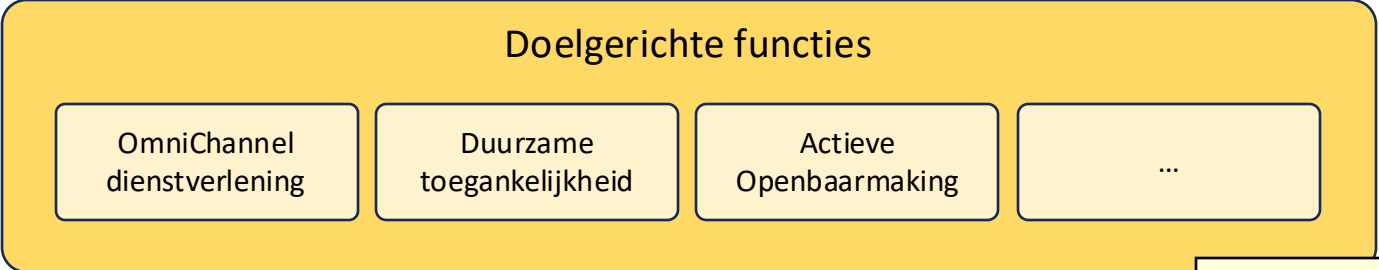
proces

diensten

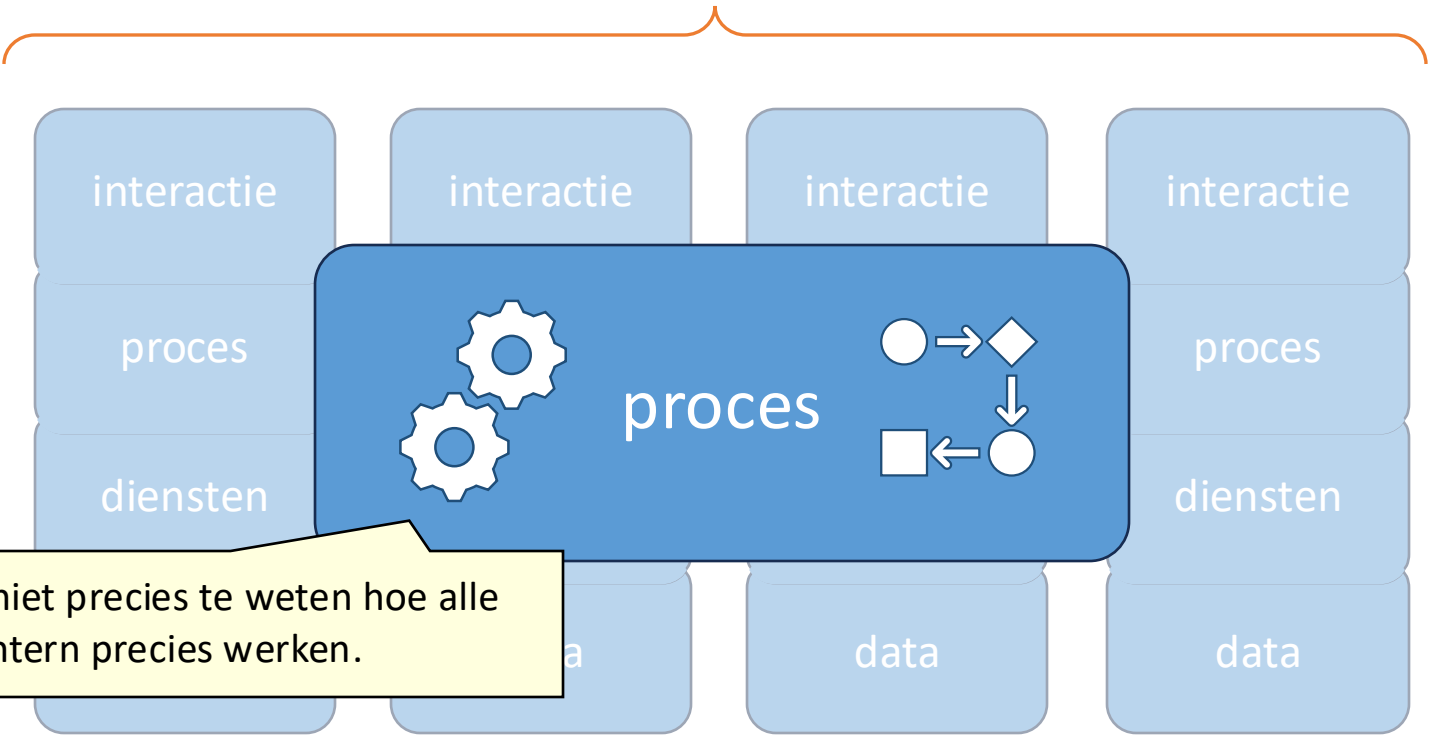
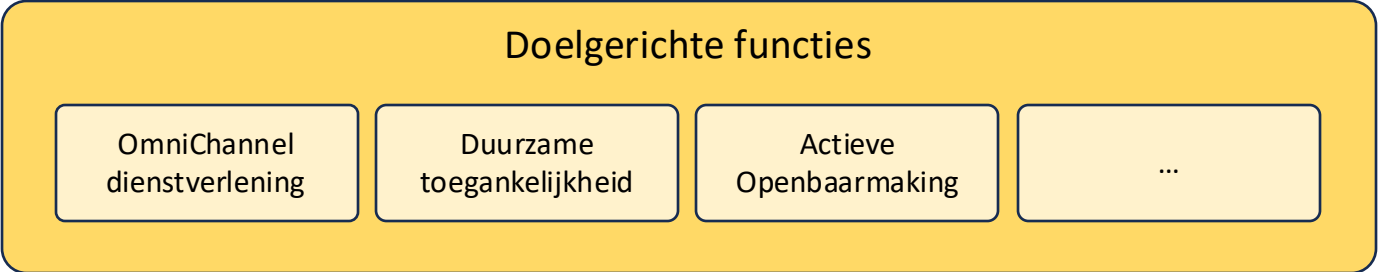
data

Laat onderstaande 4 applicaties
symbool staan voor een compleet
gemeentelijke landschap

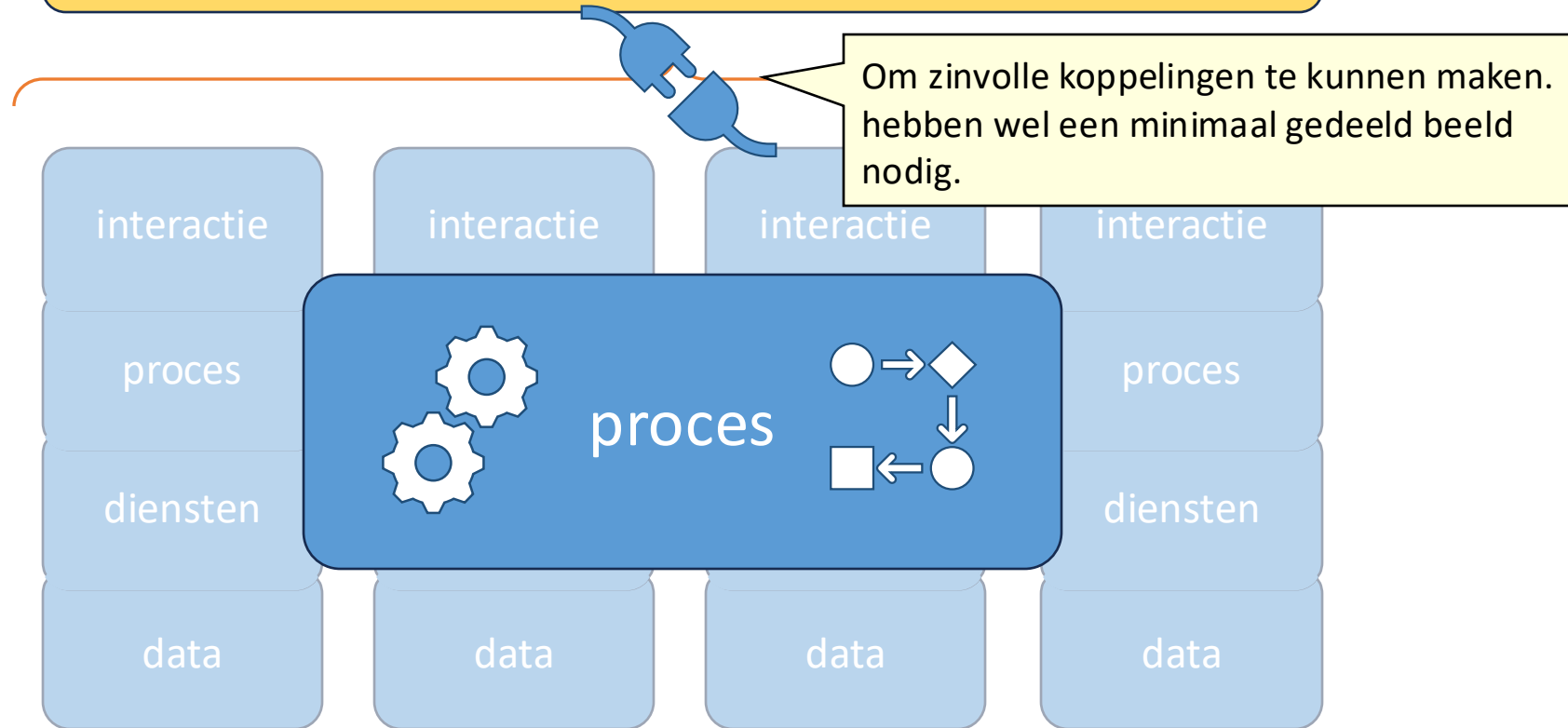
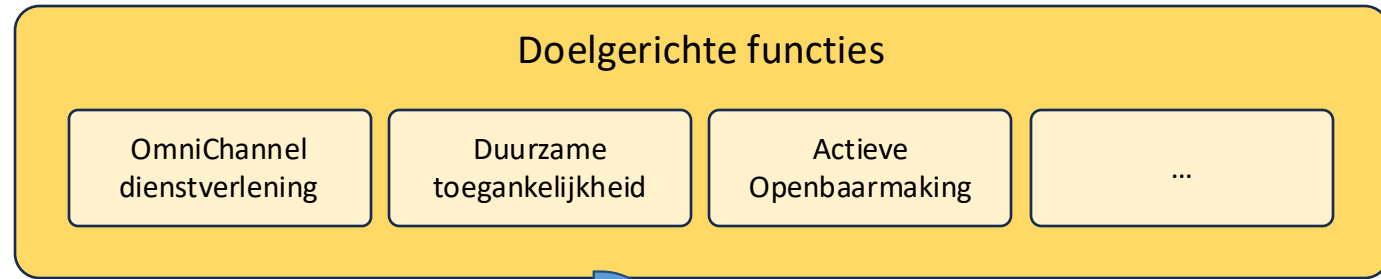


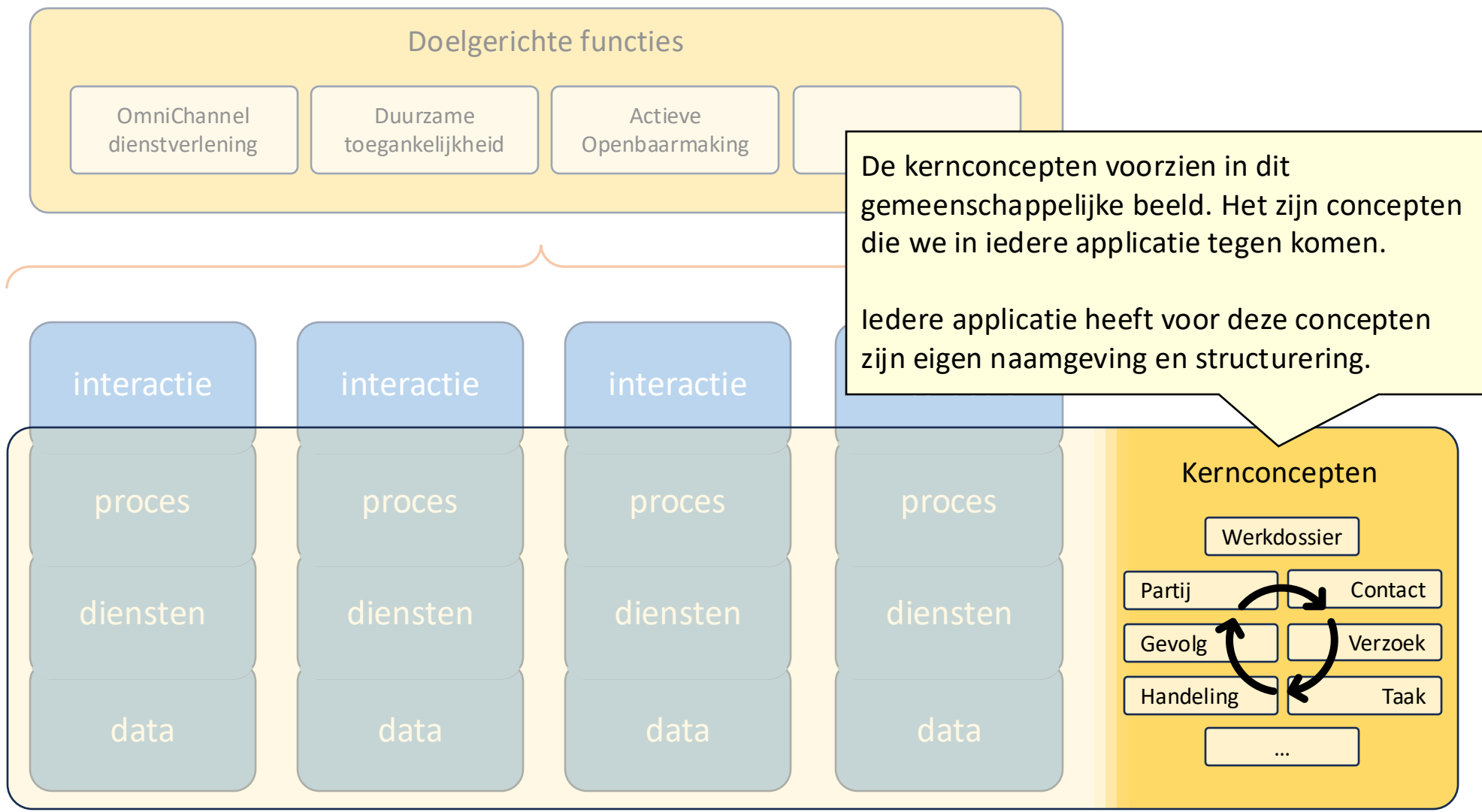


Vanuit allerlei verschillende behoeften willen we kunnen kijken naar wat er in de applicaties gebeurt en daar in bepaalde situaties ook mee kunnen interacteren.

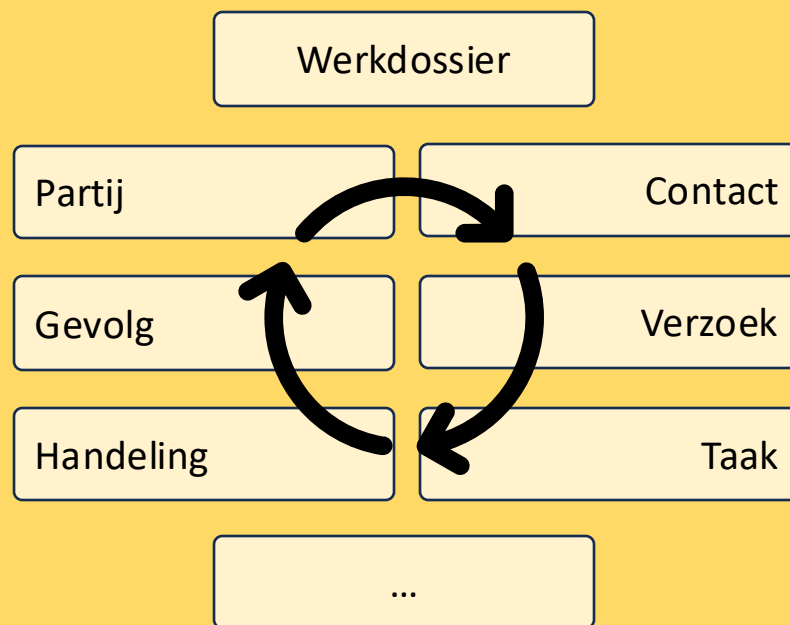


We hoeven niet precies te weten hoe alle applicaties intern precies werken.





Kernconcepten



Aan diverse van deze concepten wordt op dit moment in de Common Ground community gewerkt.

We hebben die ontwikkelingen als uitgangspunt genomen voor de doelarchitectuur.

```
graph TD; W[Werkdossier] --- C1(( )); C1 --- P[Partij]; P --- C[Contact]; C --- V[Verzoek]; V --- T[Taak]; T --- C2(( )); C2 --- G[Gevolg]; G --- H[Handeling]; H --- D[...]; D --- W
```

Werkdossier

Partij

Contact

Verzoek

Taak

Gevolg

Handeling

...

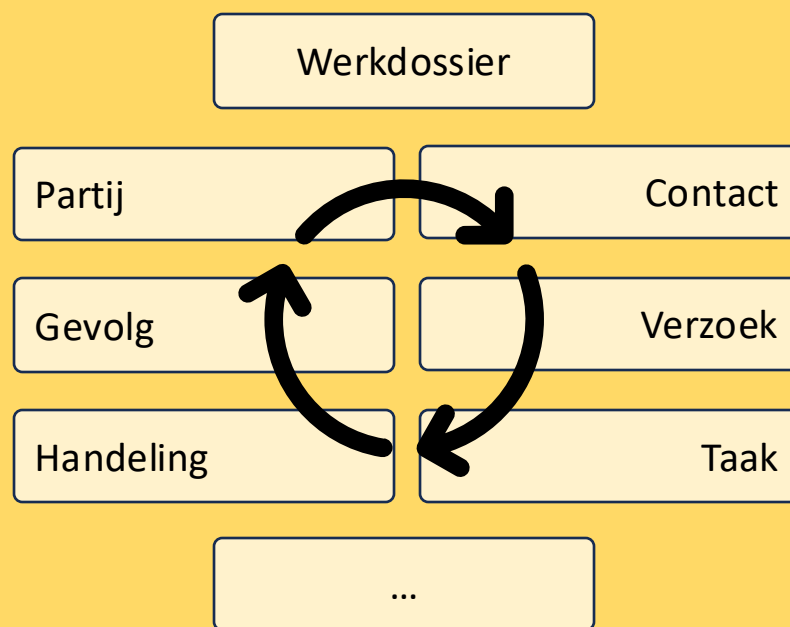
Zo kan een verzoek een ingevuld formulier zijn of een dossier van een aanvraag dat ook gestart kan zijn vanuit een mailtje.

Een contact is bij de ene gemeente iets dat heeft plaatsgevonden, en kan bij een ander bijvoorbeeld als terugbelverzoek nog even open staan.

Voor velen is een terugbelverzoek een typisch voorbeeld van een taak. Terwijl sommige gemeenten geen taken kennen en voor alles een zaak starten.

Laten we bij een uitkering of een vergunning een zaak langdurig openstaan of introduceren we hiervoor een product? En wat is dat product dan precies?

Kernconcepten

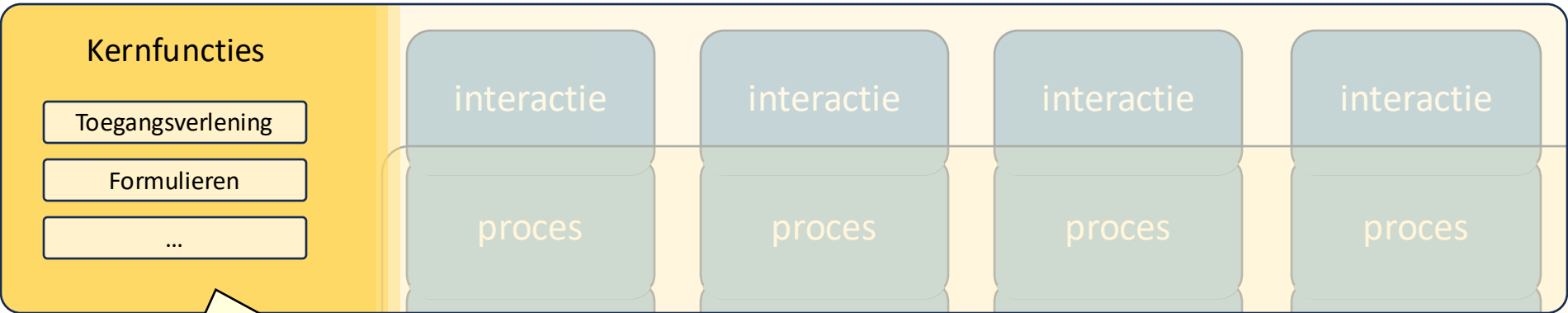


Als we tot daadwerkelijk gemeenschappelijke APIs en registers willen komen, zullen we over dit soort verschillen overeenstemming moeten bereiken.

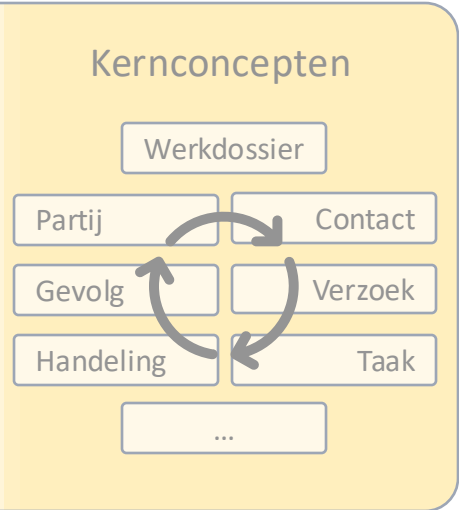
De doelarchitectuur doet hiervoor een voorstel dat we zorgvuldig gaan bespreken. We gebruiken daarbij regelmatig en bewust andere terminologie.

We zijn soms zo gewend aan bepaalde termen en de voor ons logische betekenis, dat het kan helpen om even te werken met woorden die deze betekenis nog niet hebben.

Het is nog te vroeg om hieraan conclusies te verbinden over de bestaande componenten. Zeker is dat die het startpunt zijn en dat we in de community zorgvuldig gaan kijken hoe wat we uiteindelijk zouden willen past op wat we al hebben.



De derde groep elementen zijn de kernfuncties. Dit zijn functies die in vrijwel alle applicaties voorkomen waarbij het onwenselijk is om deze telkens opnieuw te ontwikkelen. Denk aan toegangsverlening, het aanbieden van formulieren of het verzenden van een brief...



De rol van de doelarchitectuur

Er is al veel gebouwd en een deel daarvan draait al in productie bij de koploper gemeenten.

Tegelijkertijd lopen de koplopers ook tegen de nodige knelpunten aan. Soms in de techniek, soms in de semantiek (bedoelen we wel hetzelfde).

realisatie

Nu

Er is in de afgelopen jaren vanuit de VNG aan allerlei vraagstukken gewerkt:

- federatieve service connectiviteit
- logboekdataverwerking
- notificaties (cloud events)
- interactiepatronen en API's
- toegangsverlening (FTV)
- registerbouw (UBB)

Daarnaast blijven we betrokken bij landelijke ontwikkelingen (zoals de GDI) en houden we de Europese goed in de gaten.

realisatie

innovatie

Nu

Die innovatie moet uiteindelijk leiden tot een fundament voor de realisatie.

Fundament

Patronen

Toegang, notificaties, logging, APIs, caching, registers, ...

Techniek

Afspraken ivm overdraagbaarheid, onderhoudbaarheid, ...

realisatie

innovatie

Nu

Doel

Toepassing

Componenten

Standaarden

Het is “een verbouwing met de winkel open”.
We werken aan concrete toepassingen en
blijven ondertussen innoveren om de
knelpunten die we tegenkomen op te lossen.

caching, registers, ...

Techniek

Afspraken ivm overdraagbaarheid, onderhoudbaarheid, ...

realisatie

innovatie

Nu

Doel

Toepassing

Componenten

Standaarden

realisatie

Fundament

Patronen

Toegang, notificaties, logging, APIs, caching, registers, ...

Techniek

Afspraken ivm overdraagbaarheid, onderhoud

Doelarchitectuur

De doelarchitectuur moet gaan helpen om overzicht en inzicht te krijgen: Waar werken we aan, welke knelpunten moeten we oplossen, wat moet eerst, wat kan later, etc.

Nu

Doel

Toepassing

Componenten

Standaarden

realisatie

Doelarchitectuur

Fundament

Patronen

Toegang, notificaties, logging, APIs, caching, registers, ...

Techniek

Afspraken ivm overdraagbaarheid, onderhoudbaarheid, ...

innovatie

Nu