

ARCHITECTUUR

Referentie architectuur van de VERA standaard

Edwin Versteegt

Jos van Dixhoorn

21 augustus 2015

Versie 1.0

A conceptual model is neither idle nor faithful: it is, or rather it is supposed to be and so taken until further notice, an approximate representation of a real thing.

Mario Bunge (1972) Philosophy of Physics. p. 125

An abstraction is one thing that represents several real things equally well.

Edsger W. Dijkstra

Inhoudsopgave

6

Context

Inleiding op het document en het strategisch kader voor de architectuur

11

Operationele uitwisseling

Architectuur van de uitwisseling van operationele informatie

20

Prestatie uitwisseling

Architectuur van de uitwisseling van prestatie informatie

24

Stappenplan

Stappenplan om op hoofdlijnen van beoogde functionaliteit naar koppelingen te komen.

26

Bijlage A Enterprise architectuur

Bezien vanuit VERA. Wat moet een enterprise architectuur bevatten voor VERA?

30

Bijlage B Service Oriëntatie

Korte inleiding op belangrijke principes bij service oriëntatie.

Inhoudsopgave

32

Bijlage C
Termen in
samenhang

Toelichting van
belangrijke concepten
en hun samenhang

36

Bijlage D
Begrippenkader

Definities van de
gehanteerde begrippen

Inleiding

Dit is de referentiearchitectuur van de VERA standaard voor gegevensuitwisseling. Doel van deze referentiearchitectuur is aangeven in welke context de VERA standaard is ontwikkeld en binnen welke kaders de VERA standaard ontworpen is. Daarbij worden de uitgangspunten en ontwerpkeuzes aangegeven en toegelicht die bij het ontwerp van VERA genomen zijn.

Leeswijzer

Dit document is opgebouwd als een compact hoofddocument dat een aantal onderwerpen bekend veronderstelt. De toelichting op deze onderwerpen is opgenomen in bijlagen waarnaar op diverse plaatsen in de tekst wordt verwezen. Hierdoor houden we het document gefocust op de kern waarbij de bijlagen ervoor zorgen dat zo veel mogelijk lezers het document ook kunnen volgen. In bijlage D is een begrippenkader waarin de belangrijkste begrippen uit dit document zijn opgenomen.

Het document concentreert zich op de kern met in bijlages toelichting en achtergronden bij onderwerpen

Architectuur van de VERA standaard

CONTEXT

Strategisch kader

De architectuur van de VERA standaard wordt gekaderd door de missie en visie van de stichting VERA en CORA

Stichting VERA heeft een missie, visie en kernwaarden opgesteld. Deze dienen als strategisch kader voor de architectuur van de VERA standaard.

Missie: VERA realiseert standaardisatie van informatie uitwisseling in de corporatiesector door ketenpartners te verbinden.

Visie: VERA voorziet in de standaard voor eenduidige informatievoorziening van woningcorporaties, met als doel vergroting van transparantie en de reductie van complexiteit en kosten, zodat zij zich kunnen concentreren op kerntaken zonder belemmerd te worden door IT-afhankelijkheden.

VERA hanteert hierbij de kernwaarden: Samen, Transparant en Onafhankelijk.

De missie en visie van stichting VERA bevatten twee belangrijke aspecten: standaardisatie en verbinden. Het eerste aspect vult stichting VERA in door de VERA standaard die zij beheert en publiceert. Het tweede wordt ingevuld doordat stichting VERA als sectorpartij van de corporaties verbindingen legt binnen en buiten de sector.

De architectuur van de VERA standaard is onder andere opgesteld om deze missie en visie in de praktijk te kunnen brengen. VERA opereert in de kaders die CORA als referentie architectuur voor de woningcorporatie schetst.

Principiële ontwerpkeuzen

Voor de architectuur van de VERA standaard zijn een aantal principiële ontwerpkeuzen gemaakt. Deze zijn richtinggevend voor de uitwerking van de architectuur.

VERA heeft twee toepassingsgebieden: uitwisseling van operationele informatie (in uitvoering van processen) en uitwisseling van prestatie informatie (voor stuurinformatie en verantwoording)

Standaardisatie van operationele informatie en een kwalitatieve uitwisseling van deze informatie draagt bij aan de kwaliteit van de prestatie informatie.

VERA hanteert een canoniek gegevensmodel

Dit is een belangrijke basis voor VERA. Het bevordert de uitwisselbaarheid en voorkomt belemmering door IT-afhankelijkheden bij het doorvoeren van wijzigingen in de processen, informatie en applicaties van een corporatie. Een canoniek model is een sectorgeneriek model (niet specifiek voor één applicatie of transactie) en zorgt voor herbruikbaarheid en eenduidigheid.

VERA specificeert applicatie interfaces en uitwisseling

De standaard draait om het uit kunnen wisselen van informatie. VERA specificeert de interface die een applicatie moet bieden om de uitwisseling mogelijk te maken en de manier waarop de informatie uitgewisseld kan worden. VERA specificeert dus nooit hoe een applicatie informatie en gegevens moet verwerken of opslaan. Wel is het aannemelijk dat de registratie van gegevens overeenkomsten vertoont met VERA om zo de realisatie van een VERA interface op de applicatie eenvoudiger te maken. VERA is immers gebaseerd op een sectorgeneriek gegevensmodel.

Principiële ontwerpkeuzen geven richting aan de uitwerking van de architectuur

Principiële ontwerpkeuzen

VERA standaardiseert semantiek, syntax en techniek

VERA bevat een canoniek gegevensmodel dat de betekenis (semantiek) van de gegevens definieert. VERA standaardiseert de vorm (syntax) waarin de gegevens uitgewisseld dienen te worden. Bijvoorbeeld de structuur van uit te wisselen xml. VERA standaardiseert de techniek van de uitwisseling. Bijvoorbeeld de protocollen die gebruikt worden en hoe beveiliging ingeregeld kan worden.

VERA bouwt voort op bestaande standaarden en technieken

Daarom is er voor de operationele gegevensuitwisseling gekozen om gebruik te maken van Standaard UitwisselingsFormaat (StUF) voor de syntax en de techniek van de informatie uitwisseling. StUF is een standaard die voor vergelijkbare functionaliteit in de gemeentemarkt is ontwikkeld.

Voor het toepassingsgebied van de operationele informatie uitwisseling definieert VERA een service bibliotheek

Daarmee bouwt VERA dus een service georiënteerde architectuur op voor de informatievoorziening van een woningcorporatie. In bijlage B is een korte introductie over service oriëntatie te vinden.

De service bibliotheek richt zich op hergebruik van bestaande VERA berichten

Beter een bericht met iets meer informatie dat in meerdere contexten te gebruiken dan meerdere berichten die elk in één specifieke context te gebruiken zijn.

Waaruit bestaat VERA?

De VERA standaard bestaat uit een aantal onderdelen zoals in de figuur is aangegeven.

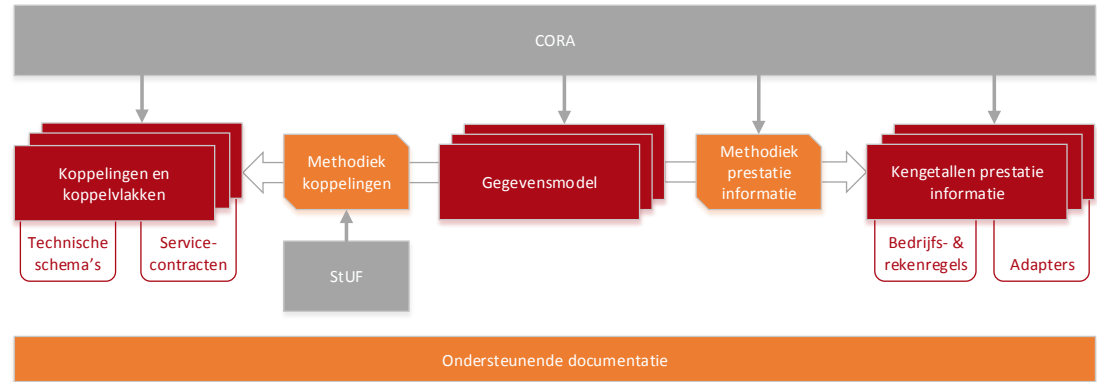
Allereerst bevat het logische gegevensmodellen. Deze bestaan uit de definities van de entiteiten en de attributen die VERA hanteert voor de uitwisseling van informatie.

Daarbij zijn er twee methodieken in VERA opgenomen. Enerzijds om te komen tot definities van koppelingen en koppelvlakken voor het toepassingsgebied van de operationele uitwisseling. Anderzijds om te komen tot kengetallen voor het toepassingsgebied van de prestatie informatie.

Voor uitwisseling van operationele informatie zijn service contracten en technische schema's opgenomen voor de koppelingen en koppelvlakken.

Voor uitwisseling van prestatie informatie zijn bedrijfs- en rekenregels opgenomen en adapters opgenomen om te komen tot kengetallen.

Bij al deze onderdelen is in VERA ondersteunende documentatie opgenomen.



Architectuur van de VERA standaard

OPERATIONELE INFORMATIE

Operationele uitwisseling in architectuur

VERA is een domein architectuur voor de gegevensuitwisseling in en met de corporatie sector. Binnen deze sector is er met CORA een referentie architectuur beschikbaar, die de kaders geeft waarbinnen de VERA gegevensuitwisseling plaats vindt. In bijlage A is beschreven hoe de de relevante CORA modellen gebruikt kunnen worden in VERA.

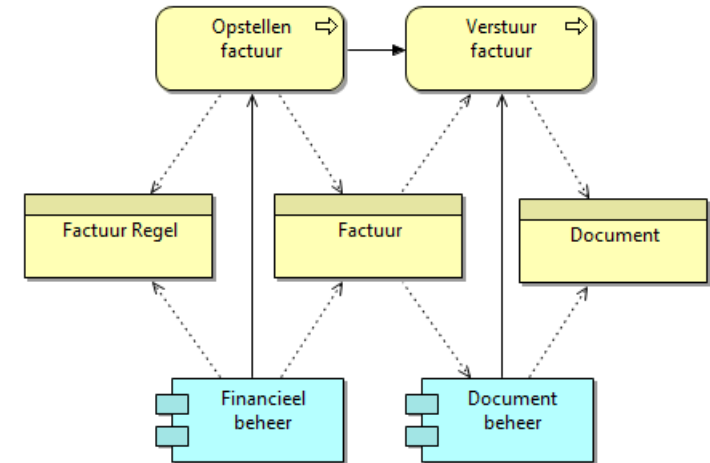
De uitwisseling van operationele informatie wordt aan de enterprise architectuur opgehangen als uitwisseling van informatie tussen logische informatie systemen in de context van een bepaald (referentie)proces.

Het (referentie)proces bepaalt welke informatie er uitgewisseld moet worden tussen welke logische informatie systemen en dus welke koppelingen en koppelvlakken nodig zijn. In bijlage C worden deze termen nader toegelicht.

Ook de benodigde integratie patronen en het initiatief voor het starten van een uitwisseling worden (voor een deel) bepaald door het proces. De gehanteerde definities voor de (CORA) bedrijfsobjecten die in het proces worden gebruikt vormen input voor het canonieke datamodel van VERA.

Als berichtenstandaard voor de operationele gegevensuitwisseling is voor StUF gekozen. StUF is een universele berichtenstandaard voor het elektronisch uitwisselen van gegevens tussen applicaties. Het domein van de StUF-taal omvat informatieketens en -functionaliteit. StUF is beschreven in XML en gebaseerd op geaccepteerde internetstandaarden. Het toepassingsgebied van StUF sluit goed aan bij dit toepassingsgebied van VERA. Door StUF te gebruiken hoeft VERA zelf geen berichtstandaard meer te ontwerpen.

Voordat we verder ingaan op de architectuur van de VERA koppelingen is het goed om kort stil te staan bij de opbouw van StUF.



StUF sectormodellen

StUF bestaat uit drie lagen: inhoud, logistiek (berichtenstandaard) en transport (protocolbindingen). Voor inhoud wordt er onderscheid gemaakt tussen zogenaamde horizontale en verticale sectormodellen. De logistiek en het transport zijn onafhankelijk van de inhoud en voor VERA zonder aanpassingen te gebruiken.

In onderstaande figuur is deze opbouw weergegeven met voorbeelden uit de gemeentemarkt en VERA.

Een sectormodel (zowel horizontaal als verticaal) bevat twee onderdelen. Ten eerste de gegevensstructuur (de VERA logische gegevensmodellen) inclusief een vertaling naar overeenkomende XML structuren. Ten tweede bevat het definities van berichten.

Een bericht is een afgeronde hoeveelheid informatie die op een bepaalde manier van een zender naar een ontvanger wordt verstuurd. Een berichtdefinitie beschrijft dus welk van de gegevens uit de gegevensstructuren zich in het bericht bevinden. Ook is er sprake van een berichttype waaruit een bepaalde betekenis is af te leiden, bijvoorbeeld een kennisgeving- of een vraagbericht.

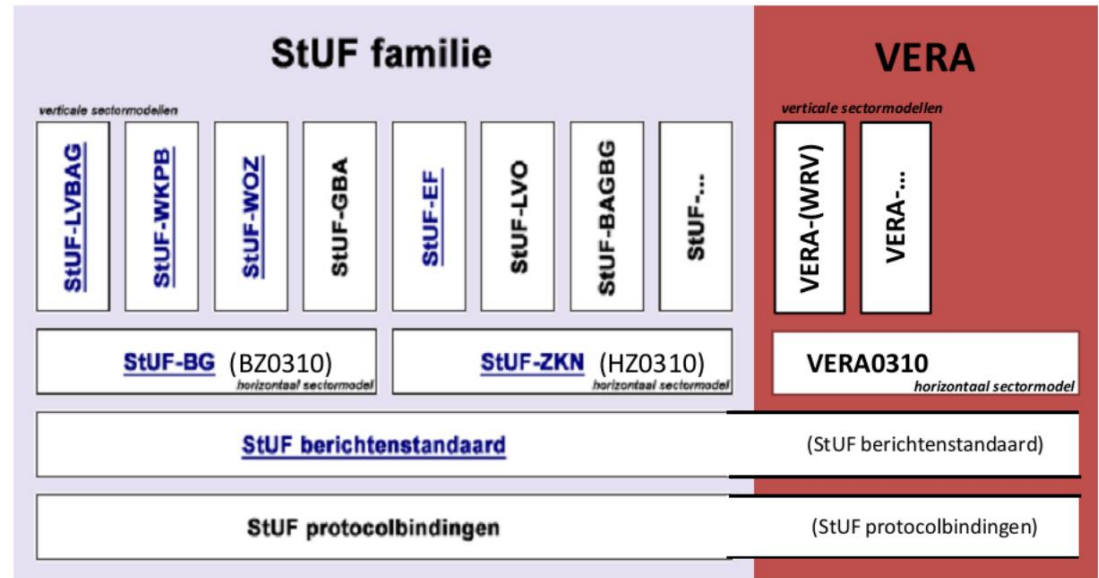
Een sectormodel bevat twee onderdelen: definities van de gegevensstructuur en definities van berichten

Een horizontaal sectormodel is een generiek model

Het bevat de gegevensstructuren en berichten voor de basale set van gegevens en berichtdefinities. Het horizontale sectormodel richt zich heel sterk op hergebruik van definities, berichten en syntax.

Een verticaal sectormodel is een specifiek model voor een toepassingsgebied

Een verticaal sectormodel bouwt verder op het horizontale sectormodel door de gegevensstructuren en de berichten te importeren en specifiek te maken. Een verticaal sectormodel bevat rijkere domeinmodellen (incl. proces en service beschrijvingen) en is gericht op een specifiek toepassingsgebied.



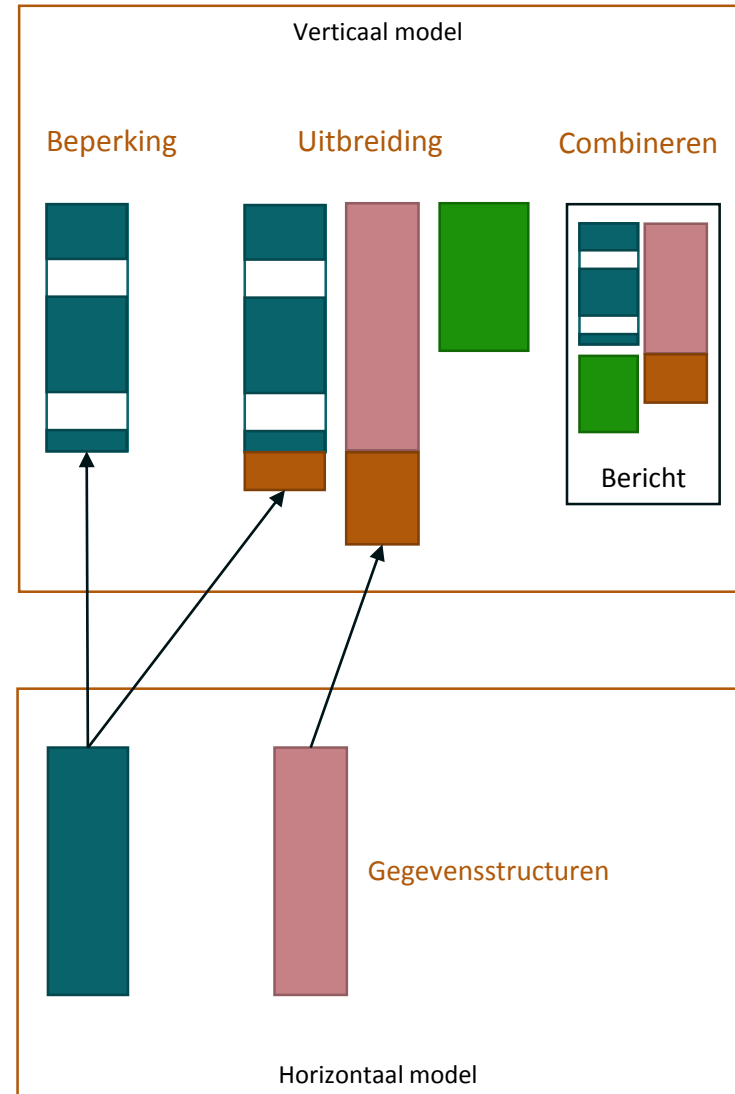
Van horizontaal naar verticaal sectormodel

StUF en daarmee VERA is zo opgebouwd dat het horizontale sectormodel de basis is. Het verticale sectormodel maakt gebruik van de gegevensstructuren uit het horizontale model. Deze gegevensstructuren kunnen worden beperkt en uitgebreid in het verticale sectormodel.

Beperking van de gegevensstructuren uit het horizontale sectormodel in het verticale sectormodel betekent dat het mogelijk is om in het verticale sectormodel te selecteren welke attributen van een entiteit uit het horizontale sectormodel nodig zijn.

Uitbreiding van de gegevensstructuren uit het horizontale sectormodel in het verticale sectormodel betekent dat het mogelijk is om in het verticale sectormodel attributen toe te voegen aan een entiteit uit het horizontale sectormodel (eventueel gecombineerd met een beperking). Daarnaast is het ook mogelijk om nieuwe entiteiten toe te voegen wanneer er entiteiten specifiek voor dit procesgebied nodig zijn.

Een berichtdefinitie bestaat primair uit twee onderdelen. Het eerste is de berichtcode die uitdrukt wat de betekenis van het bericht is. Dit wordt door StUF gedefinieerd, de in VERA ondersteunde berichtcodes zijn opgenomen in bijlage D.1 van VERA 3.0. Het tweede zijn de (**combinaties** van) gegevensstructuren die in het bericht opgenomen zijn. In het horizontale sectormodel bevatten berichten één of meer gegevensstructuren uit het horizontale sectormodel. In het verticale sectormodel bevatten berichten één of meer gegevensstructuren die zijn samengesteld zoals hierboven beschreven.



Soorten koppelingen

We onderscheiden twee soorten koppelingen: functionele koppelingen en data koppelingen

Functionele koppelingen sluiten aan bij verticale sectormodellen, data koppelingen bij horizontale sectormodellen

De koppelingen die vanuit een referentie architectuur geïdentificeerd kunnen worden zijn in twee categorieën te verdelen: functionele koppelingen, en data koppelingen.

Een functionele koppeling is een koppeling waarbij er in de uitvoering van een proces informatie van het ene logische informatie systeem naar het andere logische informatie systeem getransporteerd moet worden. Dit sluit aan bij het concept van de verticale sectormodellen in StUF.

Een data koppeling is een koppeling waarbij:

- Geraadpleegd worden uit de authentieke bron (kernregistratie)
- Gemuteerd worden in de authentieke bron (kernregistratie)
- gedupliceerd dienen te worden vanuit de authentieke bron (kernregistratie) in een andere registratie. Dit kan diverse aanleidingen hebben, bijvoorbeeld performance of beschikbaarheid.

Data koppelingen sluiten aan bij het concept van de horizontale sectormodellen in StUF.

Functionele koppelingen

Berichten voor een functionele koppeling worden bepaald door het procesmodel. Uit één of meer procesbeschrijvingen wordt geïdentificeerd welke gegevensuitwisselingen nodig zijn. Deze gegevensuitwisselingen worden beschreven in zogenaamde interactiepatronen.

Een interactie patroon is een opeenvolging van uit te wisselen berichten tussen logische informatie systemen die voorzien in een specifieke functionele behoefte. In een interactiepatroon kan sprake zijn van berichten die specifiek voor dit proces(gebied) zijn of van generieke berichten voor bepaalde stamgegevens.

De specifieke berichten en de bijbehorende gegevensstructuren worden opgenomen in het bijbehorende verticale sectormodel. De generieke berichten en de bijbehorende gegevensstructuren zijn onderdeel van het horizontale sectormodel.

Interactiepatronen zijn dus afgeleid van procesbeschrijvingen. Een interactiepatroon hoort daarom altijd bij een verticaal sectormodel van het desbetreffende proces(gebied).

De diverse berichten komen elk in een sectormodel terecht. Gezamenlijk vormen al deze berichten de service bibliotheek van VERA.

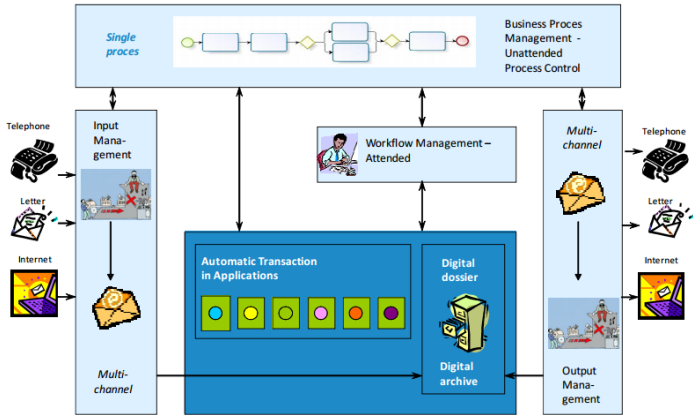
Ontwerp- en integratiepatronen

De uitwisseling van berichten die in een interactiepatroon is beschreven wordt opgebouwd uit combinaties van twee ontwerp- en drie integratiepatronen.

De ontwerp patronen zijn push / pull en synchroon / asynchroon. De integratiepatronen zijn: notificatie, synchroon verzoek-response en asynchroon verzoek-response. In bijlage C zijn deze uitgebreid beschreven.

De interactiepatronen maken gebruik van deze patronen om de opeenvolging van uit te wisselen berichten te specificeren.

E-Dienstverlening



Functionele koppelingen in een specifiek toepassingsgebied zijn onderdeel van het verticale sectormodel dat bij dat toepassingsgebied hoort. Een soort functionele koppelingen waar we extra bij stil staan gaat om e-dienstverlening.

Voor e-dienstverleningsprocessen (zowel op een klant contact centrum als via selfservice omgevingen) is over het algemeen sprake van veel interactie tussen frontoffice en backoffice. Een groot deel gaat om het ontsluiten van gegevens uit kernregistraties (data koppelingen) maar er is ook een deel wat om procesautomatisering (functionele koppelingen) gaat. Het is vanwege deze bijzondere positie van dit procesgebied dat we in deze paragraaf expliciet aangeven hoe we dat in de structuur van sectormodellen onderbrengen.

Onderbrengen in andere verticale sectormodellen

Vanuit versionering is er voor gekozen om de interactiepatronen van dit procesgebied op te nemen in elk verticaal sectormodel waar ze betrekking op hebben. Hierdoor kunnen de e-dienstverleningsprocessen en interactiepatronen mee ontwikkelen (en versioneren) met het volledige proces dat volgt op een interactie. Voor die e-dienstverleningsprocessen die niet specifiek aan een verticaal sectormodel te koppelen zijn is er echter wel een apart e-dienstverlening verticaal sectormodel.

Ontsluiten basisgegevens

Voor de benodigde basis gegevens in de e-dienstverleningsprocessen wordt gebruik gemaakt van de data koppelingen. Deze keuze is gemaakt omdat het minder complex is voor beheer door stichting VERA en voor implementatie door leveranciers dan bijvoorbeeld een systematiek die het volledig dynamisch, tijdens uitvoering, samenstellen en verwerken van gegevens mogelijk maakt.

Data koppelingen

De data koppelingen bevinden zich in het horizontale sectormodel. Het horizontale sectormodel bevat breed toepasbare gegevensdefinities en berichten. Bijvoorbeeld basisstructuren voor natuurlijk persoon of eenheid met daarbij berichten voor bijvoorbeeld het raadplegen of muteren van deze gegevens.

Zoals eerder aangegeven bevatten data koppelingen (en dus het horizontale model) berichten voor raadplegen of muteren van gegevens in de authentieke bron of het repliceren van gegevens vanuit de authentieke bron. De gegevensstructuren (en de bijbehorende berichten) zijn zo opgezet dat ze in meerdere processen bruikbaar zijn. Beter een bericht met iets teveel informatie maar wel de mogelijkheid tot herbruik dan individuele berichten.

Het initiatief voor het doorgeven van de meest actuele informatie aan andere registraties (om die synchroon te houden) ligt in principe bij de authentieke bron (kernregistratie). Dat betekent dat er een push systematiek gebruikt wordt. Als alternatief kan de actuele informatie bij de bron bevestigd worden (pull systematiek).

VERA definieert daarom kennisgevingsberichten en vraag/antwoord berichten op het horizontale sectormodel voor de hierboven beschreven functies. Zoals gezegd bevat het horizontale sectormodel alleen de basale gegevens die in meerdere verticale sectormodellen terug komen. Voor het opvragen van specifieke informatie in de context van een bepaald proces worden berichten in het desbetreffende verticale sectormodel opgenomen.

Data koppelingen zijn opgezet rondom uitwisseling van data met authentieke bronnen

Uitgangspunten en ontwerpkeuzen

In de voorgaande paragrafen is beschreven hoe berichten ingedeeld worden en hoe de verschillende soorten koppelingen passen op de indeling in sectormodellen van StUF. De uitgangspunten en ontwerpkeuzes die daarin gemaakt zijn, zijn in deze paragraaf overzichtelijk bij elkaar geplaatst.

Horizontaal sectormodel

- Is er één in VERA.
- Bevat gegevensstructuren van basale gegevens (gegevens die in meerdere sectormodellen nodig zijn). Bijvoorbeeld: eenheid, natuurlijk persoon
- Bevat generieke berichtdefinities (die in meerdere interactiepatronen worden gebruikt)
 - voor het ophalen, en wijzigen van de basisgegevens in een vraag-antwoord patroon.
 - Bevat generieke berichtdefinities voor het versturen van notificaties voor de basisgegevens
- Hergebruik zeer belangrijk: gegevensstructuren en berichten zijn bruikbaar in meerdere contexten

Verticaal sectormodel

- Zijn er meerdere in VERA, één per proces(gebied).
- Bevat specifieke gegevensstructuren voor een bepaald proces(gebied). Bijvoorbeeld onderhoudsopdracht.
- Bevat verbijzonderingen (of evt. restricties) van gegevensstructuren uit het horizontale model. Bijvoorbeeld kandidaat als verbijzondering van natuurlijk persoon.
- Bevat specifieke berichtdefinities die gebruik maken van de gegevensstructuren uit hetzelfde verticale sectormodel.
- Bevat interactiepatronen bij die beschrijven welke berichten in welke volgorde uitgewisseld moeten worden om functionaliteit voor een proces te realiseren.
- Bevatten per proces(gebied) ook interactiepatronen voor klantinteractie (frontoffice)

Architectuur van de VERA standaard

PRESTATIE INFORMATIE

Relatie met operationele informatie

Uitwisseling van prestatie informatie en het komen tot goede prestatie informatie is naast de operationele uitwisseling een ander toepassingsgebied van VERA. Het is in de basis ook fundamenteel anders dan de uitwisseling van operationele informatie.

Operationele informatie gaat om informatie die bij de uitvoering van één specifiek proces hoort. Prestatie informatie gaat juist om inzicht over processen heen waarbij analyses en trendontwikkelingen een belangrijke rol spelen.

De overeenkomst is dat beide gebruik maken van gegevens om informatie uit te wisselen. In de VERA standaard brengen we die twee toepassingsgebieden bij elkaar door van één canoniek gegevensmodel uit te gaan.

Totstandkoming prestatie informatie

Bij het tot stand komen van prestatie informatie zijn twee routes te beschouwen:

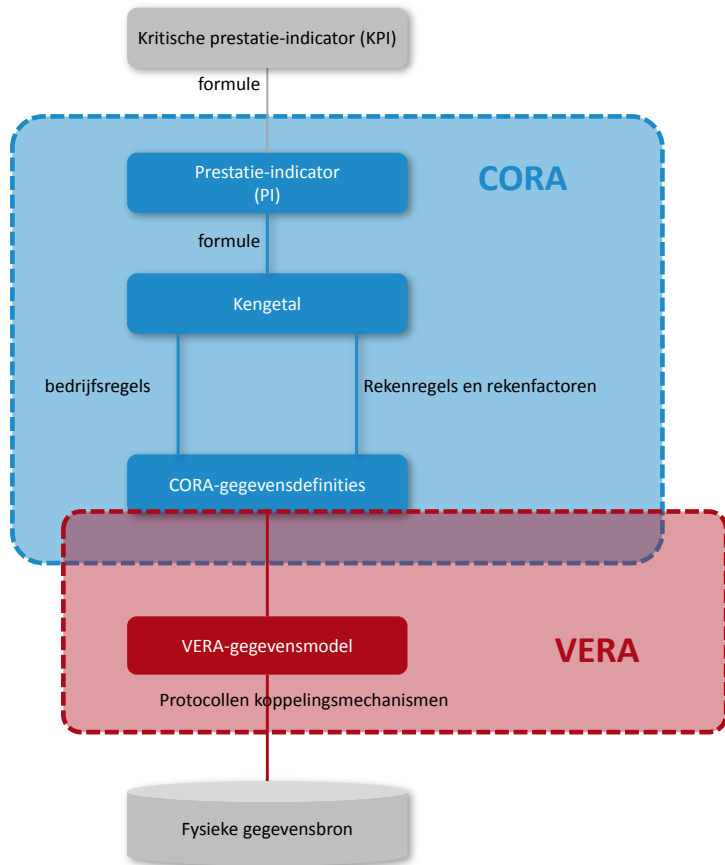
- Proces gedreven: aan de hand van de berichten die uitgewisseld worden kan onder andere door middel van filtering en aggregatie van de berichten prestatie informatie berekend worden
- Data gedreven: aan de hand van gegevens die in registraties te vinden is kan onder andere door middel van filtering en aggregatie van de gegevens prestatie informatie berekend worden

De VERA standaard biedt met de uitwisseling van operationele informatie de mogelijkheid om via de proces gedreven route tot prestatie informatie te komen. De in VERA opgenomen uitwerking concentreert zich op de data gedreven route.

Voor de totstandkoming van prestatie informatie is historie van gegevens een belangrijk onderdeel. De koppelvlakken moeten daarom zowel met formele als met materiële historie om kunnen gaan en daarmee vraagt dit ook een zelfde ondersteuning van de informatie systemen. In de operationele uitwisseling met behulp van StUF zijn er uitgebreide voorzieningen getroffen om dit in het operationele berichtenverkeer mee te nemen.

Fundamenteel is het verschil van focus op één instantie van één bepaald proces versus inzicht op meerdere instanties binnen een proces of inzicht over processen heen. Overeenkomst is dat beide gebruik maken van gegevens om tot informatie te komen

Gegevensmodellen



In het model van prestatie indicatoren is aangegeven dat deze worden opgebouwd door middel van een formule op kengetallen. Deze kengetallen komen vervolgens weer tot stand op basis van bedrijfs- en rekenregels op de gegevensmodellen. De gegevensmodellen worden ontsloten uit verschillende fysieke gegevensbronnen.

Voor VERA is met name relevant hoe de uitwisseling tot stand kan komen tussen de fysieke gegevensbron op basis van een VERA gegevensmodel en de bedrijfs- en rekenregels voor de kengetallen.

Voor de logische gegevensmodellen (die nog los staan van de implementatie) wordt ook bij de uitwisseling van prestatie informatie gebruik gemaakt van een horizontaal en een verticaal model.

Prestatie informatie heeft eigen verticale sectormodellen

Het horizontale model voor de prestatie informatie is gelijk aan het horizontale model dat ook bij de uitwisseling van operationele informatie wordt gebruikt. Het verticale model is specifiek voor de uitwisseling van prestatie informatie. Op die manier kunnen operationele uitwisseling en prestatie uitwisseling onafhankelijk van elkaar wijzigen en ook eigen gegevensstructuren gebruiken.

De verticale sectormodellen voor prestatie informatie bevatten alleen gegevensstructuren en geen StUF berichten. Dit omdat de StUF berichtensystematiek minder goed geschikt is voor het uitwisselen van prestatie informatie.

Voor implementatie moeten de horizontale en verticale logische gegevensmodellen samengevoegd worden tot één fysiek model. Dit fysieke model moet applicaties in staat stellen om geregistreerde gegevens te ontsluiten voor filtering en aggregatie.

Het fysieke gegevensmodel, dat wordt samengesteld uit de logische gegevensmodellen wordt in de architectuur gerealiseerd in de basis adapter. Dit is nader beschreven in VERA 3.0 – bijlage C.1 – prestatie meting.

Uitgangspunten en ontwerpkeuzen

- Focus op data gedreven procesinformatie
- In ontsluiting van de gegevens wordt formele en materiële historie meegenomen
- Prestatie informatie heeft eigen verticale sectormodellen
- Een verticaal sectormodel voor prestatie informatie bevat geen StUF berichten
- VERA moet een andere technische systematiek dan StUF definiëren voor het uitwisselen van prestatie informatie middels basis- en steradapters.

Architectuur van de VERA standaard

STAPPENPLAN

Realiseren koppelingen

Stap 1: Breng het ketenproces in kaart

- Bepaal de koppelmomenten
- Bepaal welke informatieoverdracht daar plaats vindt
- Bepaal tenminste use cases voor raadplegen, opvoeren, wijzigen en verwijderen van gegevens in de keten

Stap 2: Bepaal de bedrijfsobjecten en hun onderlinge relatie

- Bepaal 1-1; 1-n; n-1 relaties tussen de objecten
- Abstractieniveau Bron: CORA met ondersteuning van VERA gegevensmodel

Stap 3: Bepaal welk systeem welk(e) proces(stap) ondersteunt

- Logische informatie systemen en authentieke bronnen
- Bron: informatiedomeinen CORA

Stap 4: Bepaal uitwisseling van VERA dataobjecten tussen de systemen

- Vertaal van bedrijfsobject naar VERA dataobject
- Bepaal de te ondersteunen verplichte en optionele interactiepatronen
- Bij nieuw proces en nieuwe applicaties zijn essentiële vragen:
 - Moet het zelfstandige entiteit zijn?
 - Potentiële attribuutuitbreiding?
 - Verplichte velden?

Stap 5: Generieke berichten bepalen

- Zijn er meer (potentiële) gebruikers van het dataobject?
- Bepaal tevens attributen die in nabije toekomst nodig zijn
- Global business identifier in het canoniek dataformaat (geen GUID)
- Als het masterdata betreft, dan ook beheerproces inregelen

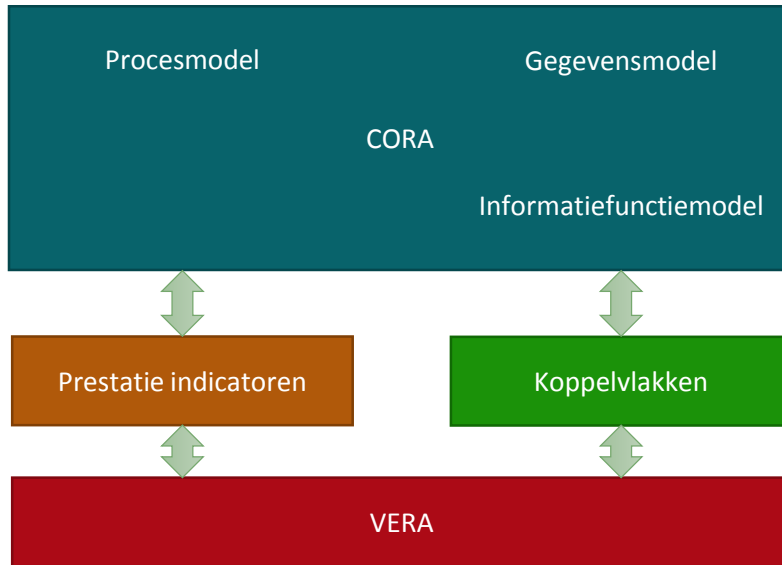
Stap 6: Vertaal VERA koppelvlakken naar het logische datamodel van bron - en doel systeem

- Indien er al een VERA koppelvlak is voor een systeem, dan hoeft dat voor dat koppelvlak niet meer gedaan te worden
- Randvoorwaarde aan de applicaties: data kan geïmporteerd en geëxporteerd worden; randvoorwaarde aan specificaties: mogelijkheden aan applicaties
- Fieldmapping bepalen

Bijlage A

ENTERPRISE ARCHITECTUUR

Metamodel enterprise architectuur



De informatiedomeinen in het informatiefunctiemodel staan model voor de logische informatiesystemen

Met een blik vanuit VERA op CORA hebben we het bovenstaande metamodel voor ogen. In goed overleg zal er gewerkt worden aan een verdere aansluiting tussen CORA en VERA.

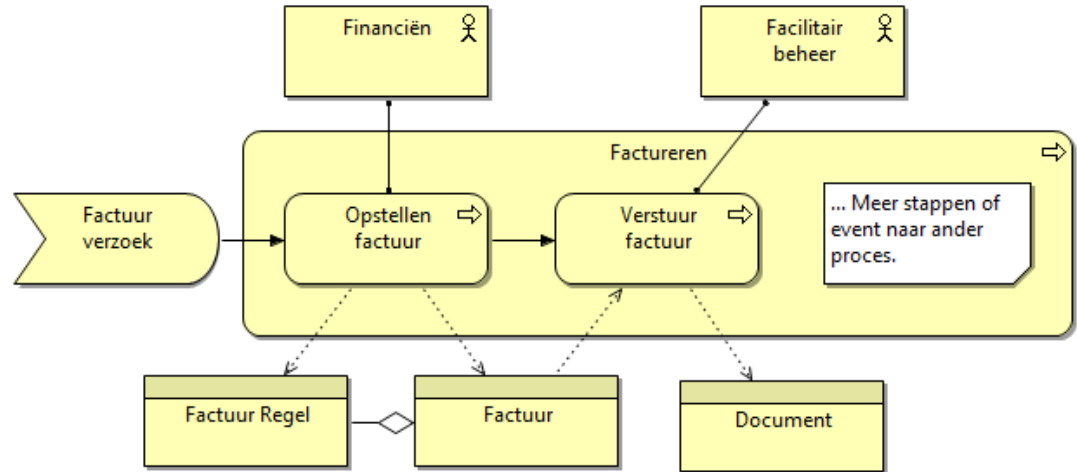
De figuur is een uitwerking van het stappenplan beschreven in dit document. Het globale Procesmodel in CORA zal meer in detail uitgewerkt moeten worden bij aanvang van dit stappenplan. Het gegevensmodel van CORA geeft een beeld van de bedrijfsobjecten. De informatiedomeinen in het informatiefunctiemodel staan model voor de logische informatiesystemen. De koppelvlakken zijn een essentieel en volledig nieuw onderdeel die tijdens uitvoering uitgewerkt zal moeten worden.

Processen

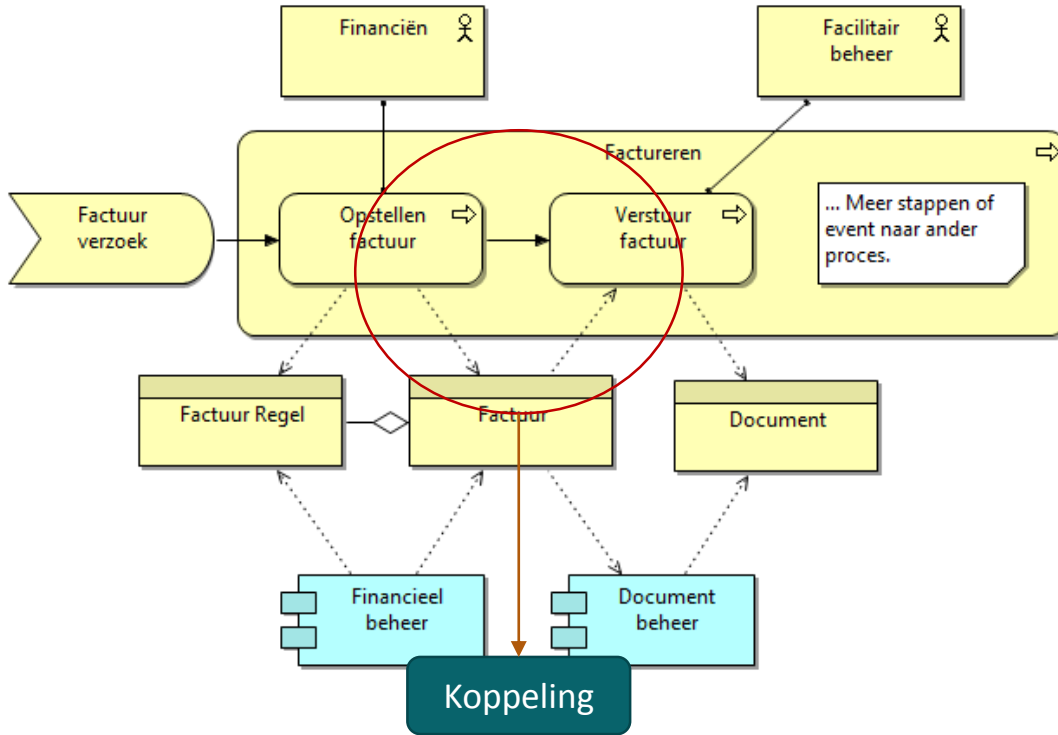
Een organisatie (bv. een corporatie) voert diverse processen uit die elk een (deel)dienst of (deel)product leveren voor de markten of doelgroepen die zij bedienen. Deze processen worden uitgevoerd door bedrijfsfuncties (implementatieonafhankelijk) die verdeeld zijn over organisatieonderdelen (implementatie-specifiek).

Processen zijn op verschillende abstractieniveaus te onderscheiden, een gangbare indeling is te vinden in CORA 3.0 Tabel 5.1. Een bedrijfsreferentiearchitectuur beschrijft over het algemeen de keten-, bedrijfs- en werkprocessen.

Een werkproces heeft in principe één bedrijfsobject (gegevensdefinitie) als onderwerp van handeling. Bedrijfsobjecten zijn aan werkprocessen te relateren door lees of schrijf relaties aan te brengen. Werkprocessen kennen vaak een hoge mate van herbruikbaarheid, bijvoorbeeld "Onderhouden Relatie".



Logische informatiesystemen en koppelingen



Een referentiearchitectuur is implementatie-onafhankelijk, er kan dus niet over concrete applicaties uit de markt gesproken worden (implementatie-specifiek). Daarvoor wordt gebruik gemaakt van iets wat een logisch informatiesysteem wordt genoemd (informatiedomein in CORA).

Een logisch informatiesysteem is eigenaar van een bepaalde gegevensverzameling. Logische informatiesystemen zijn daarom te onderscheiden door te analyseren en te clusteren welke gegevens door welke werkprocessen worden behandeld. De informatiedomeinen uit CORA bieden een goed overzicht van logische informatiesystemen.

Voor VERA is relevant door welke logische informatiesystemen een proces wordt ondersteund. Er is sprake van een potentiële koppeling wanneer twee opeenvolgende processen door andere logische informatiesystemen worden ondersteund, of het resultaat van een proces wordt later in een ander logisch informatiesysteem gebruikt (in een ander proces). Welke gegevens de koppeling moet uitwisselen wordt door de gegevensdefinities (de bedrijfsobjecten) uitgedrukt.

Of een potentiële koppeling voor VERA daadwerkelijk waarde heeft is in de markt te peilen door dit expliciet te vragen en door te kijken in welke mate de twee verschillende logische informatiesystemen van de koppeling in de praktijk ook in twee losse applicaties (implementatie specifiek) bestaan.

Bijlage B

SERVICE ORIËNTATIE

Service Oriëntatie

Service-oriëntatie, vertaling van Service-oriented architecture (SOA), is een architectuurmodel, geen technologie op zich. Centraal bestaat een SOA-opgebouwd systeem uit servicecontracten. Hierbij is sprake van afnemers van diensten en leveranciers.

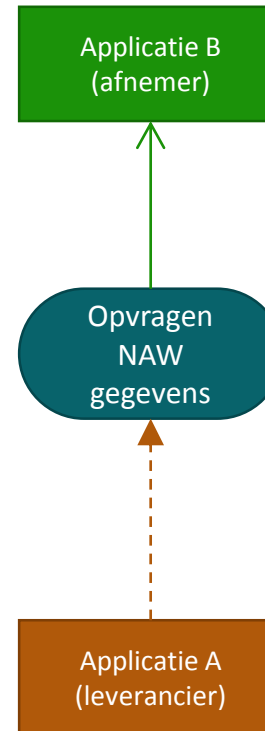
Kenmerken

Kenmerkend voor de diensten (en contracten) is:

- Een service is virtueel: de afnemer heeft geen weet van de implementatie van de dienst. De dienst is onafhankelijk van de afnemer. Scheiding van verantwoordelijkheid is expliciet vastgesteld. Voordeel hiervan is onafhankelijkheid van veranderingen van afnemer en leverancier. Voldoen aan het contract staat immers centraal;
- Een service is gestandaardiseerd: er is slechts één implementatie aanwezig van een verantwoordelijkheid. Voordeel is het rationaliseren van de standaard.
- Een service is modulair (vervangbaar) en compositieerbaar. Een standaard is niet flexibel. Flexibiliteit wordt bereikt door combineren (componeren) van standaards tot een nieuwe standaard;
- Een service is abstract: generiek, niet afgestemd voor 1 specifieke afnemer, maar op een doelgroep van afnemers;
- Losgekoppeld: afnemer en leverancier zijn maximaal onafhankelijk van implementatie van beide. Elke service is daarom autonoom. Er bestaat geen directe link of relatie tussen verschillende services. Services zijn zich ook niet van elkaar bewust.

Technologie

Om invulling te geven aan deze SOA principes zijn er verschillende technologieën beschikbaar. Bijvoorbeeld: SOAP, XML, ESB, Web services. Deze technologieën worden dus gebruikt om een architectuurmodel te realiseren, maar zijn geen doel op zich.



Applicatie B gebruikt de service opvragen NAW gegevens. Applicatie B noemen we een 'afnemer' van de service.

De service opvragen NAW gegevens heeft een service contract wat de functionele en niet-functionele eisen beschrijft die de service invult.

Applicatie A levert de service opvragen NAW gegevens. Applicatie A noemen we een 'leverancier' van de service

Bijlage C

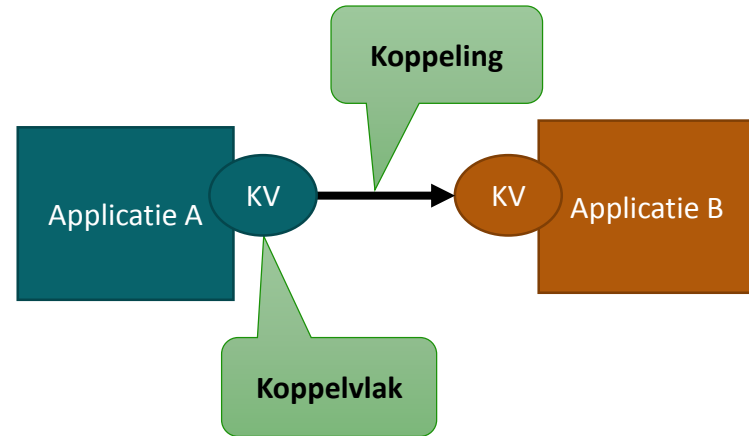
TERMEN IN SAMENHANG

Koppeling en koppelvlak

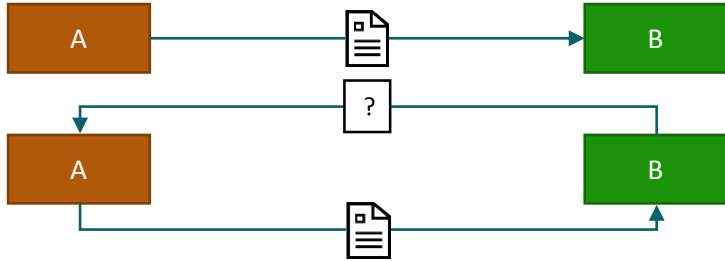
In VERA wordt in de terminologie een belangrijk onderscheid gemaakt tussen koppeling en koppelvlak.

Een koppeling is de samenwerking tussen twee (of meer) applicaties om een bepaald proces te automatiseren of digitaliseren. Een koppeling bestaat vaak uit de specificatie van het uitwisselen van berichten met gegevens en de condities waaronder deze uitwisseling plaatsvindt. Een koppeling maakt gebruik van koppelvlakken.

Een koppelvlak is een service of interface op een applicatie die in staat is om berichten met gegevens te versturen en te verwerken. Een koppelvlak ontsluit functionaliteit en gegevens binnen bepaalde eisen. De eisen (functioneel en niet-functioneel) worden primair bepaald door het proces en de benodigde koppeling en zijn vastgelegd in het servicecontract of de koppelvlakspecificatie. Hetzelfde koppelvlak kan in meerdere koppelingen gebruikt worden.



Ontwerppatronen



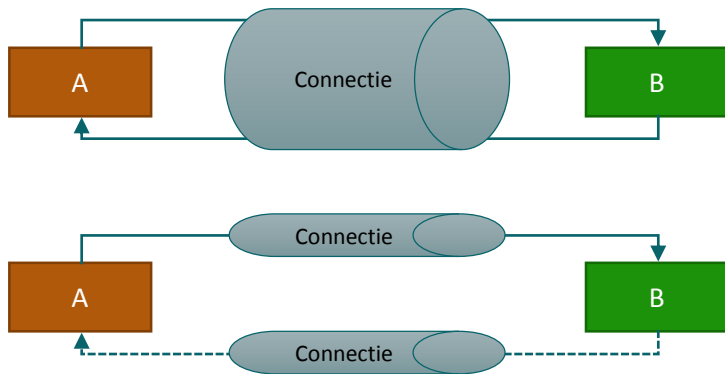
Push vs Pull

De ontwerppatronen push en pull gaan over waar het initiatief ligt om een uitwisseling uit te voeren, of anders gezegd wie er de uitwisseling start.

Push: A stuurt de informatie naar B. A neemt dus het initiatief.

Pull: B haalt de informatie op bij A door een vraag aan A te stellen. B neemt dus het initiatief.

Het onderscheid tussen push en pull is met name relevant wanneer de laatste versie van een gegeven uit A ook in B nodig is. Bij een push neemt A het initiatief om de laatste versie naar B te sturen. Bij een pull haalt B de laatste versie op bij A. Dit kan bijvoorbeeld om gegevens van een huurder gaan, maar ook om te verwerken facturen of opdrachten.



Synchroon vs Asynchroon

Een synchroon of asynchroon ontwerppatroon gaat over de verbinding die gebruikt wordt voor het transport van de informatie. Verbinding is hier bedoeld in de meest algemene en abstracte zin van het woord.

Synchroon: A verzendt een bericht naar B en krijgt op dezelfde verbinding een antwoord terug. A en B moeten dus per definitie tegelijkertijd beschikbaar zijn.

Asynchroon: A verzend een bericht naar B en sluit de verbinding. Een eventueel antwoord van B wordt op een andere verbinding verstuurd. Hier kan een aanzienlijke tijd tussen zitten.

Asynchrone communicatie schaal over het algemeen beter. Bovendien hoeft A de verbinding niet open te houden maar kan deze meteen sluiten wanneer het bericht bij B of tussenliggende middleware is afgeleverd.

Integratiepatronen

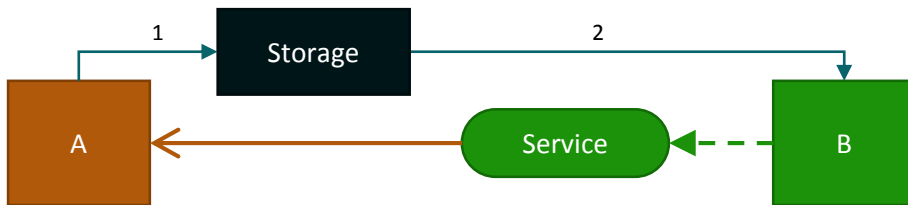
Er zijn drie verschillende integratiepatronen onderkend in VERA, namelijk: request-reply, store-forward en publish-subscribe.

Deze integratiepatronen maken gebruik maken van de hiervoor beschreven ontwerp patronen. Voor store-forward en publish-subscribe is over het algemeen aanvullende middleware nodig.

Een uitleg over het gebruik van services en de gehanteerde notatiewijze vind u in bijlage B. In alle drie de uitwerkingen van de patronen op deze pagina is er sprake van informatie die van A naar B (en evt. C) moet, de dunne genummerde blauwe pijlen geven de volgorde en stroom van de informatie c.q. berichten aan.

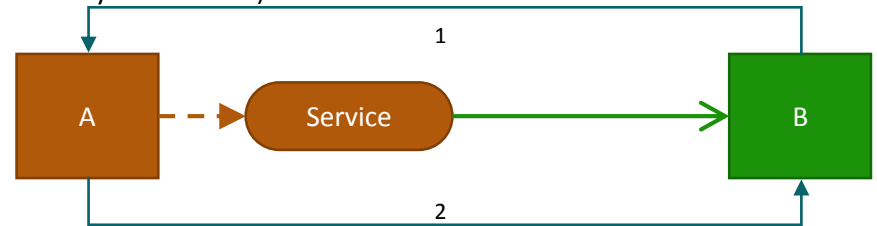
Store-forward

Voor meldingsfunctionaliteit waarbij A functionaliteit in B wil uitvoeren maar het resultaat niet direct nodig heeft voor verdere verwerking. Het initiatief ligt bij A die een push van de informatie uitvoert via asynchrone communicatie. Wordt ook wel fire-and-forget genoemd.



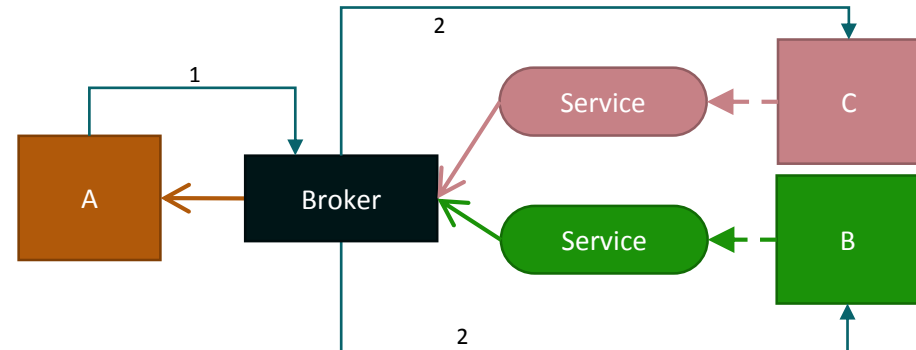
Request-reply

Voor raadpleging- of bewerkingfunctionaliteit waarbij B het antwoord van A nodig heeft voor verdere verwerking. Het initiatief ligt bij B, een raadpleging is een pull van informatie en een bewerking is een push van informatie. In praktijk altijd via synchrone communicatie (alhoewel het ook asynchroon kan).



Publish-Subscribe

Voor notificatiefunctionaliteit waarbij een of meerdere applicaties op de hoogte gesteld moeten worden van een bepaalde gebeurtenis zodat ze daar actie op kunnen ondernemen. Het initiatief ligt bij A die een push van de gebeurtenis uitvoert via asynchrone communicatie.



Bijlage D

BEGRIPPENKADER

Begrippenkader

Authentieke bron

Die bron, waar de oorspronkelijk ingezamelde, correcte én volledige informatie zich bevindt. Alle services moeten daar informatie ophalen. Authentieke gegevensbronnen vormen de spil van het hedendaagse databeheer. Door allen deze data te gebruiken en eventuele fouten te melden, krijgen we steeds betere gegevens. Zie ook CORA principe 6.

Bericht

Een bericht is een afgeronde hoeveelheid gegevens (met een begin en een eind) die van een verzender naar een ontvanger verstuurd wordt.

Service

Een mechanisme om toegang te krijgen tot één of meer functies, waarbij toegang wordt verschaft op basis van een vooraf beschreven interface en wordt verleend in overeenstemming met de beleidslijnen en beperkingen van de service beschrijving. Zie ook bijlage B.

Informatie

Informatie bestaat uit ‘nieuws’, dat kennis en begrip toevoegt, wat effect heeft op menselijk handelen. Informatie is gebaseerd op gegevens.

Interactiepatroon

Een opeenvolging van uit te wisselen berichten die voorzien in een specifieke functionele behoefte.

Interface

Zie koppelvlak.

Formele historie

Wat was – met de kennis *van een bepaalde datum* in het verleden – de waarheid over een bepaald gegeven. Meer informatie is te vinden in: “representatie formele en materiele historie” een bijlage van GEMMA.

Gegevens

Gegevens zijn (fysieke) representaties van termen en concepten, die in een bepaalde context betekenis kunnen hebben voor mensen.

Historie

Zie materiele historie en formele historie.

Kernregistratie

Zie authentieke bron.

Koppeling

Een koppeling bestaat uit een set van procesmatige, semantische, syntactische en technische afspraken die nodig zijn om de communicatie tussen twee partijen mogelijk te maken en goed te laten verlopen en daarmee een bepaald proces te digitaliseren.

Koppelvlak

Een koppelvlak is de interface of API die een applicatie aanbiedt voor het ontsluiten van functionaliteit en gegevens.

Logisch informatie systeem

Een logisch informatie systeem is een systeem voor het verwerken van (formele) gegevens. Een logisch informatie systeem is een implementatie onafhankelijke weergave van applicaties die bij corporaties gebruikt worden. Een applicatie kan meerdere logische informatie systemen realiseren.

Begrippenkader

Materiele historie

Wat was – met de kennis *van dit moment* – de waarheid over een gegeven op een bepaalde datum in het verleden. Meer informatie is te vinden in: “representatie formele en materiele historie” een bijlage van GEMMA.

Middleware

Middleware omvat de systeemsoftware die de informatie-uitwisseling faciliteert tussen applicaties.

Registratie

Verzameling van gegevens die is vastgelegd.