

Introductie van een servicelaag

Introductie van een servicelaag voor canonieke data ontsluiting

Bert Dingemans

Abstract

Bij het inrichten van canonieke data ontsluiting wordt vanuit architectuur veelal de eis geformuleerd dat een aansluitende applicatie (bron of doel) een servicelaag dient te hebben. Echter wat houdt deze eis in en waarom wordt deze eis aan applicaties gesteld bij introductie van canonieke data ontsluiting?

Op deze vragen proberen we in dit white paper een antwoord te geven. Enerzijds door te beschrijven hoe de service inrichting binnen een organisatie is vormgegeven met behulp van een conceptuele servicebus. Deze conceptuele servicebus zorgt ervoor dat de inrichting van applicaties eenvoudiger wordt en dat deze applicaties de rol van afnemer of aanleveraar van gegevens gaan vervullen.

De inrichting van de conceptuele servicebus wordt in paragraaf 6 beschreven inclusief de verschillende rollen die een applicatie binnen deze inrichting kan vervullen. Daarnaast wordt aangegeven met welke technologie deze conceptuele servicebus is ingericht. Verder worden een aantal integratiepatronen behandeld. Bij het inzetten van servicetechnologie is standaardisatie een belangrijk aspect. Door deze standaardisatie kunnen applicaties onafhankelijk van de (technische) inrichting met elkaar communiceren. Standaardisatie is van belang voor zowel het transport van berichten als voor de inhoud van deze berichten.

Ter illustratie wordt een voorbeeld van een applicatie gegeven waarin de aansluiting op de conceptuele servicebus met behulp van een servicelaag. Dit voorbeeld laat een aantal mechanismen zien op basis waarvan de servicelaag van een applicatie op adequate wijze webservice technologie inzet.

Ten slotte wordt ingegaan op algemene SOA eisen en meer specifieke eisen op basis van de inrichting binnen de provincie. Deze eisen dienen door de leverancier beantwoord te worden bij een offerteaanvraag of aanbestedingstraject. De voorgaande hoofdstukken gelden hierbij als achtergrondinformatie om de leverancier in staat te stellen zich een beeld te vormen van de achtergrond van de eisen.

Inleiding

Vanuit doelarchitecturen wordt service orientatie als een belangrijk speerpunt gekenmerkt voor de toekomstige informatievoorziening. Echter service orientatie heeft vergaande gevolgen voor de inrichting van de informatievoorziening. Dit heeft tot gevolg dat er vanuit deze service orientatie gedachte eisen gesteld worden aan nieuw te selecteren en huidige (toekomstvaste) applicaties en componenten.

Dit document beschrijft enerzijds de gewenste inrichting van het applicatielandschap vanuit het perspectief van services. Anderzijds wordt een lijst van eisen en wensen uitgewerkt die ingezet kunnen worden bij de selectie van applicaties en componenten. Deze eisen en wensen kunnen ingezet worden bij aanbestedings- en softwareselectietrajecten.

Basisprincipes servicelaag

Voor het stellen van kaders gelden de principes zoals geformuleerd in de referentie architectuur. Deze zijn uitgewerkt in een aantal basis principes. Het volgende basisprincipe geldt voor de servicelaag.

SERVICE GERICHTE ARCHITECTUUR (SGA)

VOOR DE INRICHTING VAN PROCESSEN, INFORMATIEVOORZIENING EN APPLICATIES HANTEERT DE PROVINCIE GELDERLAND EEN SERVICE-GERICHTE ARCHITECTUUR (SGA): DE INFORMATIEVOORZIENING VAN DE FUNCTIES EN PROCESSEN IS VIA SAMENWERKENDE SERVICES;

Rationale:

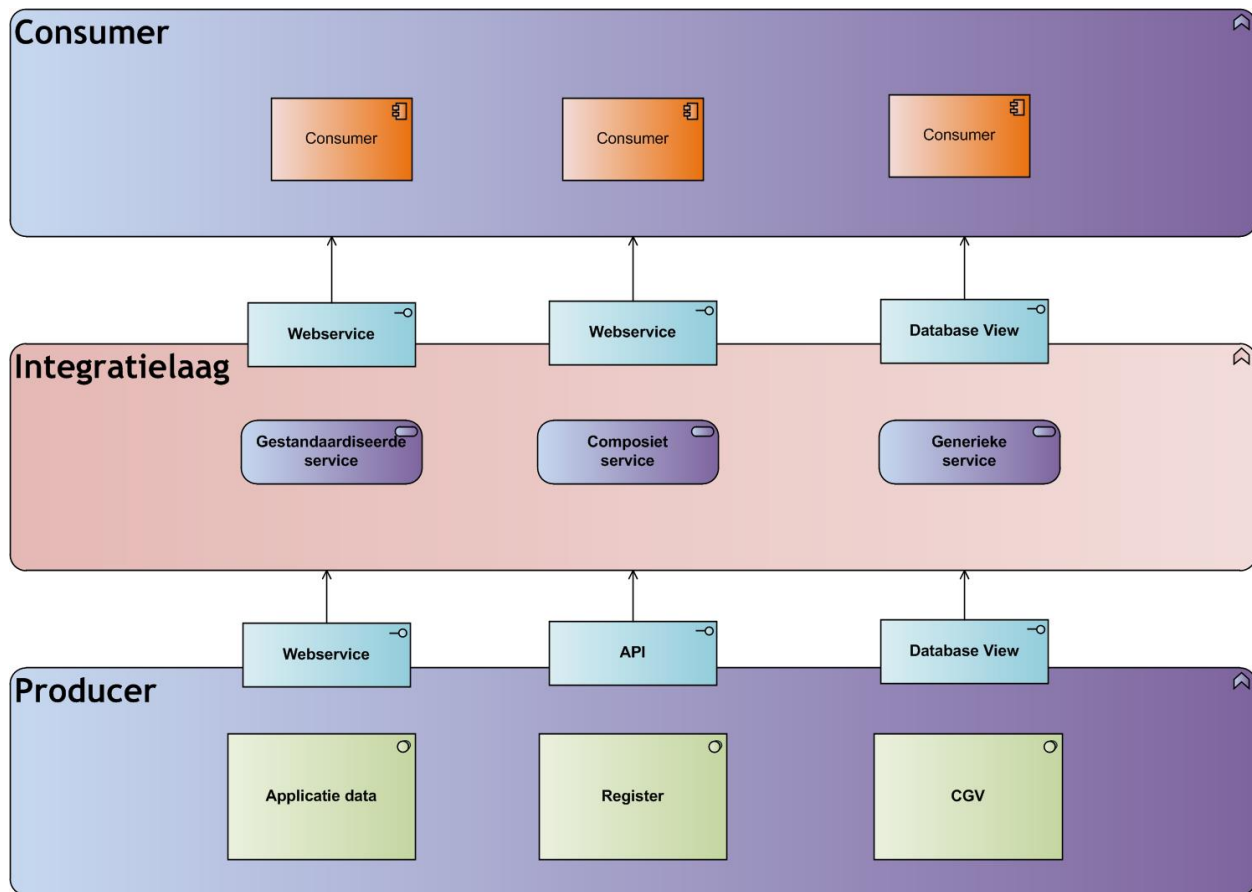
Voor de inrichting van processen, informatievoorziening en applicaties hanteren wij een service-georiënteerde architectuur. Informatievoorziening is gebaseerd op applicatiecomponenten die elk een specifieke bedrijfsfunctie ondersteunen. Interactie tussen deze functies en componenten vindt plaats op basis van gestandaardiseerde services om flexibiliteit en koppelbaarheid te vergroten. Deze services kunnen andere services aanroepen waardoor een gelaagde service-architectuur ontstaat.

Implicaties

Introductie van service georiënteerd werken dient gestimuleerd te worden en heeft vanuit architectuuroogpunt altijd de voorkeur boven andere manieren van koppelen . Dit is enerzijds door het ontmoedigen van “silo-applicaties” en anderzijds door het stellen van eisen aan de service orientatie van aangeschafte toepassingen. Als er gesproken wordt over een servicegerichte architectuur dan worden in één adem ook webservices bedoeld. Werken met webservices is de techniek onder een servicegerichte architectuur. Data in authentieke registers wordt via gestandaardiseerde- en beschreven services ontsloten.

Architectuurschets servicelaag

De service of integratielaag kan het beste beschreven worden op basis van een schets. In onderstaande afbeelding wordt deze schets getoond op basis van de ArchiMate notatie.



In de afbeelding zijn drie applicatiefuncties (of logische applicaties) te onderkennen:

- **Consumer**, dit is de afnemer van bepaalde diensten of services die door andere applicatie functies worden aangeboden. De services die aangeboden worden aan deze consumers hebben bij voorkeur een generiek karakter. Let op dat het afnemen niet alleen bestaat uit het opvragen van data, de dienst kan ook bestaan uit het doen van mutaties op (generieke) entiteiten.,
- **Producer**, is een leverancier van bepaalde services. Dat kunnen gegevensverstreckende - maar ook gegevensbewerkende services zijn. Gegevensbewerkende services passen de status van (generieke) data entiteiten aan binnen registers en applicatiedata
- **Integratie- of servicelaag**, dit is een conceptuele servicebus die ervoor zorgt dat de applicatiespecifieke inrichting van de producers maar ook het landschap van producers ingekapseld wordt voor de consumers. Hiermee wordt bereikt dat de consumers minder afhankelijk worden van de producers Deze servicelaag bestaat uit een aantal services en bijbehorende interfaces die een fysieke implementatie van deze (generieke) services aanbieden aan de consumers.

Bovenstaande afbeelding geeft een samenvattende beeld, er zijn meerdere vormen van consumers en producers mogelijk in verschillende configuraties. Kenmerkend is wel de drieindeling die toegepast wordt waarbij applicaties en componenten die via projecten geïntroduceerd worden altijd tot de consumer en/of producer functie behoren. Je mag hiervan afwijken maar dan geldt de pas toe of leg uit, de services moet dan wel aan een aantal voorwaarde voldoen zoals, implementeert een open standaard (bijvoorbeeld CMIS) en is niet

afhankelijk van inrichting van onderliggende. Hierbij geldt dat deze toepassingen beide rollen kunnen vervullen. De integratielaag wordt namelijk als een generieke voorziening aangeboden. In het volgende hoofdstuk wordt hier nader op ingegaan. Naast deze interne integratielaag zijn er een aantal belangrijke integratiediensten en standaarden aanwezig. Denk bijvoorbeeld aan digikoppeling en digipoort.

Service-laag en applicaties

Service- of integratielaag

Dit is het generieke deel dat ingezet wordt voor het realiseren van een aantal belangrijke SOA principes zoals inkapseling en abstractie.

Regelmatig wordt in organisaties de discussie gevoerd of een enterprise service bus ingezet moet worden binnen de technische infrastructuur. Zolang het aantal services binnen de service of integratielaag nog beperkt in aantal is wordt hiervoor niet gekozen. Wel is het van belang dat er een goede administratie komt van de aanwezige (herbruikbare en generieke) services binnen deze servicelaag.

Bij de inzet van een conceptuele servicebus voor het verwerken van transacties (zie paragraaf 6.4.2 en verder voor de patronen) dient rekening gehouden te worden met het bewaken van deze transacties, bijvoorbeeld door inzet van een atomic service transaction of compensating service transaction patroon. Dit wordt ingericht binnen de tooling van de conceptuele servicebus. Biztalk of de OSB leveren hier bijvoorbeeld inrichtingsvoorbeelden van. De consumers en de producer dienen aan deze requirements geen beperkingen te stellen.

Voor toekomstvaste oplossingsrichtingen biedt een technologie leverancier tegenwoordig ook cloud versies aan. Hiermee is het mogelijk om integratie- en workflow oplossingen met een behoefte aan een hoge beschikbaarheid en elasticiteit in te richten op basis van deze cloud diensten. De keuze voor een cloudoplossing wordt in de rest van het document niet uitgewerkt maar is daarmee zeker wel een oplossingsrichting.

Applicatie als consumer

De consumer of afnemer is de toepassing die een bepaalde service afneemt of consumeert.

Bekendste vorm is dat deze afnemer via een service bepaalde gegevens opvraagt via de service. Een andere vorm is het doorgeven van mutaties aan bijvoorbeeld een register of het doen van een aanvraag in de vorm van een melding. Binnen digikoppeling wordt het onderscheid gemaakt tussen meldingen en bevestigingen. In dit document wordt een soortgelijke indeling gedaan.

Tussen de consumer en de producer kunnen verschillende uitwisselpatronen toegepast worden, deze zijn uitgewerkt in paragraaf 5.4. Een bijkomend aspect is dat de consumer of de producer buiten de eigen organisatiegrenzen kan liggen. Bijvoorbeeld het rechtstreeks bevestigen van de GBA houdt in dat de consumer intern is maar dat de producer van deze gegevens voor ons onbekend is. Het enige wat in deze voor de interne consumer relevant is is het contract van de service die de producer aanbiedt.

In hoofdstuk 8 worden een aantal requirements geformuleerd rond de consumers en producers binnen een servicelaag. In hoofdstuk 11 wordt dit gedetailleerd door een lijst van softwareeisen. Aan de inrichting van een consumer worden een aantal aanvullende eisen gesteld op het vlak van de inrichting. In de situatie waar de consumer via bevestigingen gegevens opvraagt uit registers

dient de applicatie erin te voorzien dat het interne datamodel dit ondersteunt. Veel huidige applicaties ondersteunen deze eis nog niet of onvoldoende.

Applicatie als producer

De producer of aanbieder is de toepassing die een bepaalde service aanbiedt. Een voorbeeld is het op basis van bevestigingen teruggeven van de gevraagde gegevens aan de afnemer. Een ander voorbeeld is het verwerken van bepaalde gestandaardiseerde deelprocessen die geïmplementeerd worden achter een service die de procesverwerking als dienst aanbiedt.

Voor de producer gelden dezelfde uitgangspunten op het vlak van integratiepatronen en requirements als de consumer.

Integratiepatronen

Bij service orientatie zijn een aantal vormen van communicatie (berichtuitwisseling) tussen twee applicaties te onderkennen, dit wordt omschreven als integratievarianten of -patronen. Deze patronen hebben een aantal kenmerkende eigenschappen. In dit hoofdstuk beschrijven we de beschikbare basisvarianten en lichten we deze toe met een voorbeeld of implementatie. Op basis van deze basisvarianten kunnen ingewikkelder patronen samengesteld worden door deze basispatronen met elkaar te combineren. Zie bijvoorbeeld <http://eaipatterns.com>.

Alle uitwerkingen zijn hierbij gebaseerd op een basispatroon van applicatiefuncties: consumer, servicelaag en producer. In dit patroon is de consumer het koppelvlak dat het initiatief neemt tot interactie, de integratielaag zorgt voor ontkoppeling en de producer reageert op de aanvragen (meldingen of bevestigingen) die binnenkomen.

Pull variant

Afbeelding



Toelichting

Een applicatie heeft extra gegevens nodig welke opgeslagen zijn in een andere applicatie. Deze gegevens worden op basis van een pull mechanisme (vraag en antwoord) opgehaald bij de andere applicatie en in de eigen toepassing gebruikt voor het nemen van een beslissing of voor verrijking van de eigen gegevensset. De Pull variant is ook bekend als bevestigingsvariant of request-response.

Kenmerken

- Initiatief ligt bij het bevestigende systeem
- Bidirectioneel berichtenverkeer
- Veelal gebaseerd op hoge frequentie en aantallen
- Minder geschikt bij hoge betrouwbaarheidseisen

Toekomstvaste integratievormen

- Biztalk

- .Net/WCF
- OSB
- Mule ESB

Voorbeeld

Dit mechanisme wordt toegepast binnen het CDO relatie koppelvlak, zoals de webservice ZoekRelatie. Deze stuurt een bericht met de zoekgegevens van een specifieke relatie. Het antwoordbericht bevat de detailgegevens van de gevraagde relatie.

Push variant

Afbeelding



Toelichting

Binnen de push variant verandert de status van één of meer entiteiten binnen het applicatiedomein. Vervolgens wordt deze statuswijziging gecommuniceerd naar één of meer afnemende applicaties. Op basis van de ontvangen gegevens bepaalt de afnemende applicatie of mutaties nodig zijn binnen de eigen entiteiten.

Kenmerken

- Initiatief ligt bij het systeem waar de mutaties plaatsvinden, bijvoorbeeld statusveranderingen in registers
- Unidirectioneel berichtenverkeer
- Veelal lage frequentie en aantallen
- Minder geschikt bij hoge betrouwbaarheidseisen (geen check of berichten aankomen)

Toekomstvaste integratievormen

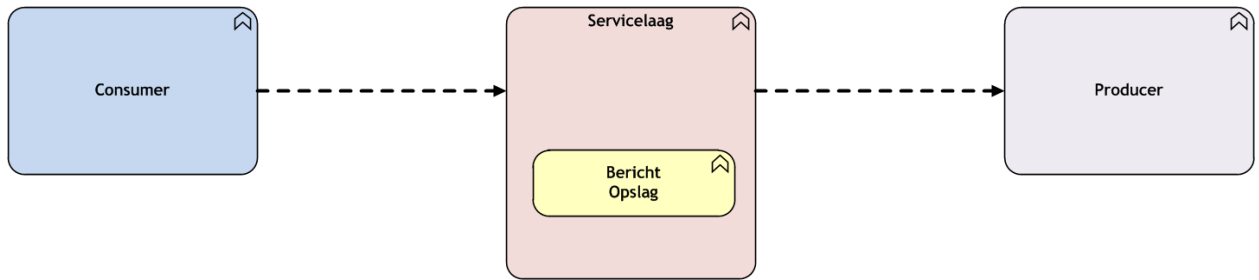
- Biztalk
- OSB
- Mule ESB

Voorbeeld

Binnen de BAG is een register ingericht met alle ingevoerde gegevens van adressen en gebouwen. Voert de registerbeheerder een mutatie door dan moeten de afnemende informatiesystemen hierover geïnformeerd worden.

Push variant store en forward

Afbeelding



Toelichting

Binnen de push variant verandert de status van één of meer entiteiten binnen het applicatiedomein. Vervolgens wordt deze statuswijziging gecommuniceerd naar één of meer afnemende applicaties. Echter men moet zeker zijn dat alle afnemende applicaties het bericht ontvangen voor verwerking. De queue houdt bij of alle afnemende applicaties het bericht hebben afgenomen. Op basis van de ontvangen gegevens bepaalt de afnemende applicatie of mutaties nodig zijn binnen de eigen entiteiten. Dit patroon wordt ook wel aangeduid met de term ‘melding’ of Fire and Forget.

Kenmerken

- Initiatief ligt bij het systeem waar de mutaties plaatsvinden bijvoorbeeld statusveranderingen in registers
- Unidirectioneel berichtenverkeer
- Veelal lage frequentie en aantallen
- Geschikt bij verschillende beschikbaarheid van bron en doelsysteem
- Toepasbaar bij lage eisen verwerkingsnelheid
- Zeer geschikt bij hoge betrouwbaarheidseisen

Toekomstvast integratievormen

- Biztalk
- OSB
- Mule ESB

Voorbeeld

Digikoppeling kent de meldingsvariant die is gebaseerd op het ebMS protocol. Dit protocol is gebaseerd op een afleverzekerheid van de berichten binnen een adapter.

Push variant met terugmelding

Afbeelding



Binnen de push variant verandert de status van één of meer entiteiten binnen het applicatiedomein. Vervolgens wordt deze statuswijziging gecommuniceerd naar één of meer afnemende applicaties. Deze afnemende applicatie meldt vervolgens terug dat de gegevens

correct ontvangen en verwerkt zijn. Veelal wordt in de terugmelding extra informatie zoals een uniek registratienummer teruggemeld.

Kenmerken

- Initiatief ligt bij het systeem waar de mutaties plaatsvinden bijvoorbeeld statusveranderingen in registers
- Bidirectioneel berichtenverkeer
- Veelal lage frequentie en aantallen
- Hoge eisen aan beschikbaarheid van het doelsysteem
- Veelal hoge eisen verwerkingsnelheid
- Minder geschikt bij hoge betrouwbaarheidseisen

Toekomstvaste integratievormen

- Biztalk
- OSB
- Mule ESB

Voorbeeld

Toepassing binnen meldingen zoals subsidieaanvragen. Hierbij doet de aanvrager een melding, bijvoorbeeld de aanvraag van een subsidie. Hierbij worden de gegevens verwerkt in het subsidieregister en deze geeft een melding terug aan de aanvrager waarin het unieke registratienummer (dat bijvoorbeeld bij vervolginteracties gebruikt moet worden) is opgenomen.

Push-Pull variant

Afbeelding



Toelichting

Binnen de push variant verandert de status van één of meer entiteiten binnen het applicatiedomein. Vervolgens wordt deze statuswijziging gecommuniceerd naar één of meer afnemende applicaties, meestal met een bericht met enkel de identificerende gegevens. Deze afnemende applicatie bepaalt of de statuswijziging relevant is voor de eigen entiteiten. Zo ja dan worden de detailgegevens van de gemuteerde entiteiten in het bronsysteem op basis van een pull variant opgehaald op een voor de afnemende applicatie logisch moment. Dit patroon wordt gebruikt als er relatief veel gegevens overgedragen moeten worden en als niet zeker is of al die gegevens voor alle afnemende applicaties relevant zijn. Het stelt aanvullende eisen aan de consumer en producer applicatie (opslag van sleutels e.d.)

Kenmerken

- Initiatief van het push gedeelte ligt bij het systeem waar de mutaties plaatsvinden bijvoorbeeld statusveranderingen in registers; het initiatief voor het pull gedeelte ligt bij de andere systemen.
- Bidirectioneel berichtenverkeer
- Veelal lage frequentie en aantallen
- Omvangrijk pull bericht en klein pushbericht
- Extra registratie van ontvangen push berichten noodzakelijk in afnemende toepassing
- Veelal hoge eisen aan beschikbaarheid van het bronsysteem
- Hoge betrouwbaarheidseisen aan push kanaal, minder hoog aan pull kanaal

Beschikbaar bij integratievorm

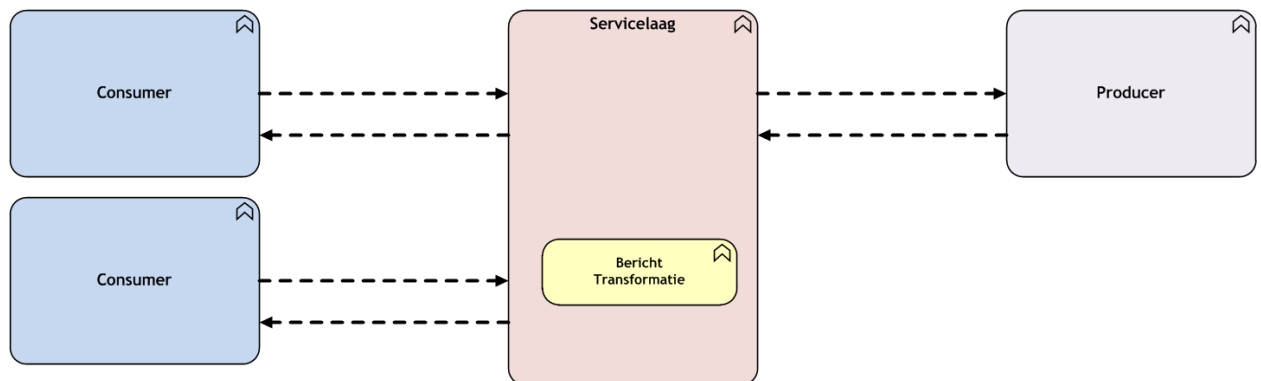
- Biztalk
- WCF
- OSB
- Mule ESB

Voorbeeld

Binnen de basisregisters is deze vorm geïmplementeerd binnen verschillende abonnementsvormen.

Pull variant met transformatie

Afbeelding



Toelichting

Het vragende informatiesysteem heeft behoefte aan gegevens afkomstig uit meerdere bronsystemen. Er wordt daartoe een vraag uitgezet die door een broker (o.i.d.) wordt doorgestuurd naar meerdere bronsystemen. Deze retourneren elk een antwoord. De broker verzamelt deze retourberichten en transformeert deze tot één retourbericht voor het aanvragende informatiesysteem.

Kenmerken

- Initiatief ligt bij het vragende systeem dat gegevens nodig heeft afkomstig uit een aantal bronsystemen
- Bidirectioneel berichtenverkeer
- Veelal hoge frequentie en aantallen

- Omvangrijk response bericht en klein requestbericht
- Gegevensset opgebouwd uit gegevens afkomstig uit meerdere bronsystemen
- Veelal hoge eisen aan beschikbaarheid van de bronsystemen
- Broker functionaliteit in de infrastructuur

Beschikbaar bij integratievorm

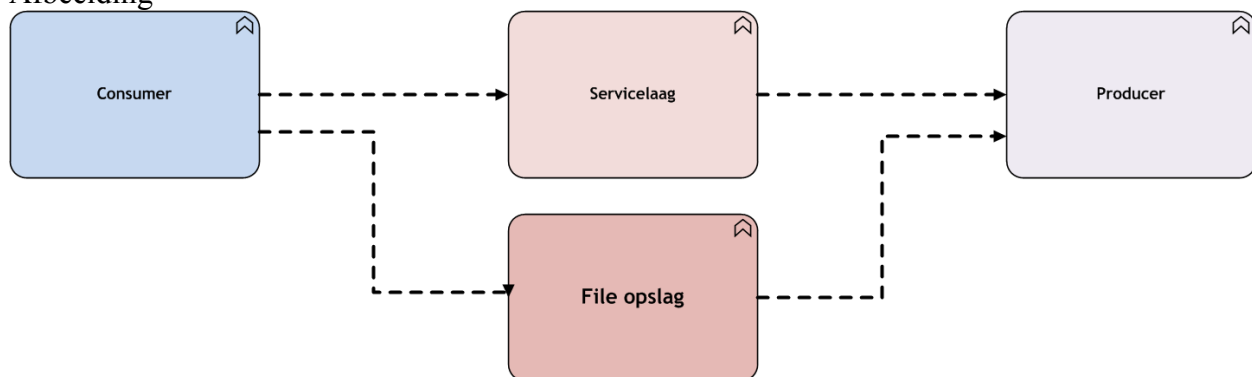
- Biztalk
- OSB
- Mule ESB

Voorbeeld

Canonieke data ontsluiting (CDO) is een voorbeeld van deze variant. Hierbij wordt een gestandaardiseerde interface beschikbaar gesteld van een register, de inrichting van het CDO transformeert de gegevens ten behoeve van de standaard interface .

Push variant met grote berichten

Afbeelding



Toelichting

In sommige gevallen zijn de te verzenden gegevensset dusdanig groot qua omvang dat berichtenverkeer niet geschikt is. De (omvangrijke) gegevensset wordt dan opgeslagen op een medium en vervolgens worden de afnemende informatiesystemen met berichtenverkeer geïnformeerd over het beschikbare bestand. Veelal zijn in dit bericht gegevens over de beschikbaarheidsperiode en beveiliging (zoals versleuteling) opgenomen. Ook bekend als claim check patroon. Dit patroon dient als alternatief voor bestandsuitwisseling en FTP/SFTP.

Kenmerken

- Initiatief ligt bij het bronsysteem dat gegevens aanbiedt
- Unidirectioneel berichtenverkeer
- Veelal lage frequentie en aantallen
- Zeer omvangrijke berichten
- Veelal hoge eisen aan beschikbaarheid van opslagmedium van de grote berichten
- Kan voldoen aan hoge beveiligingseisen

Toekomstvaste integratievormen

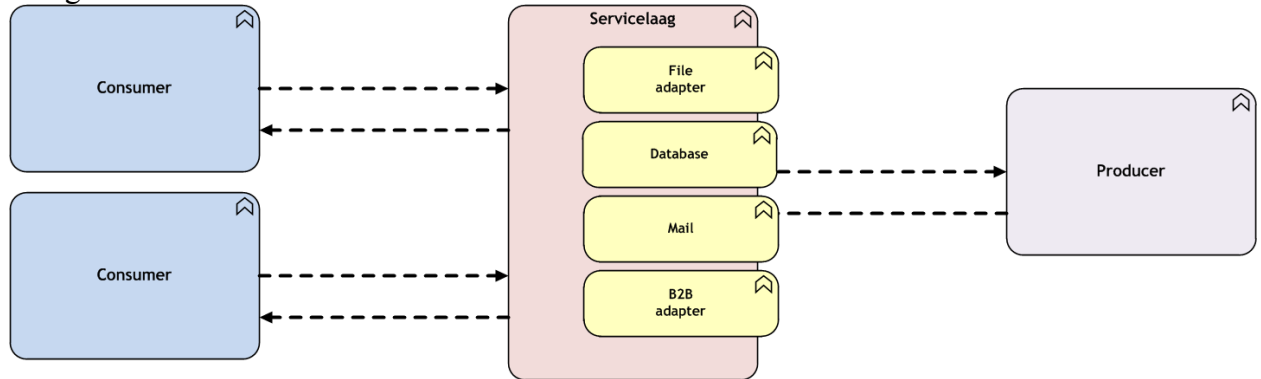
- Biztalk

- OSB/B2B adapter
- Mule ESB

Digikoppeling kent een extensie voor grote berichten. Hiermee is het mogelijk om omvangrijke registergegevens beschikbaar te stellen aan meerdere afnemers.

End point varianten

Afbeelding



Toelichting

Een endpoint of adapter bevindt zich op het grensvlak tussen applicatie en integratielaag.

Endpoints zijn specifiek qua technologie en (deels) specifiek voor de applicatie.

Binnen de berichtuitwisseling zijn meerdere endpoint varianten beschikbaar. In de afbeelding zijn de meest bekende opgenomen. De ontwikkeling van endpoints is aan verandering onderhevig, vraag dus een actueel overzicht op bij afwijkende wensen. Met name de B2B endpoint kent een aantal extra transformatie mogelijkheden. Implementatie van endpoints is ook bekend onder de noemer adapter.

Kenmerken

- Transformatie tussen berichtstandaard en specifieke endpoint functionaliteit
- Transformatie van berichten
- Eisen aan beschikbaarheid afhankelijk van zowel bron als doelsysteem

Toekomstvaste integratievormen

- File adapter
 - Biztalk
 - Oracle
 - Mule
- Mail
 - Biztalk
 - Oracle
 - Mule
- B2B adapters ¹
 - Biztalk
 - Oracle
- Database
 - WCF
- Webservices

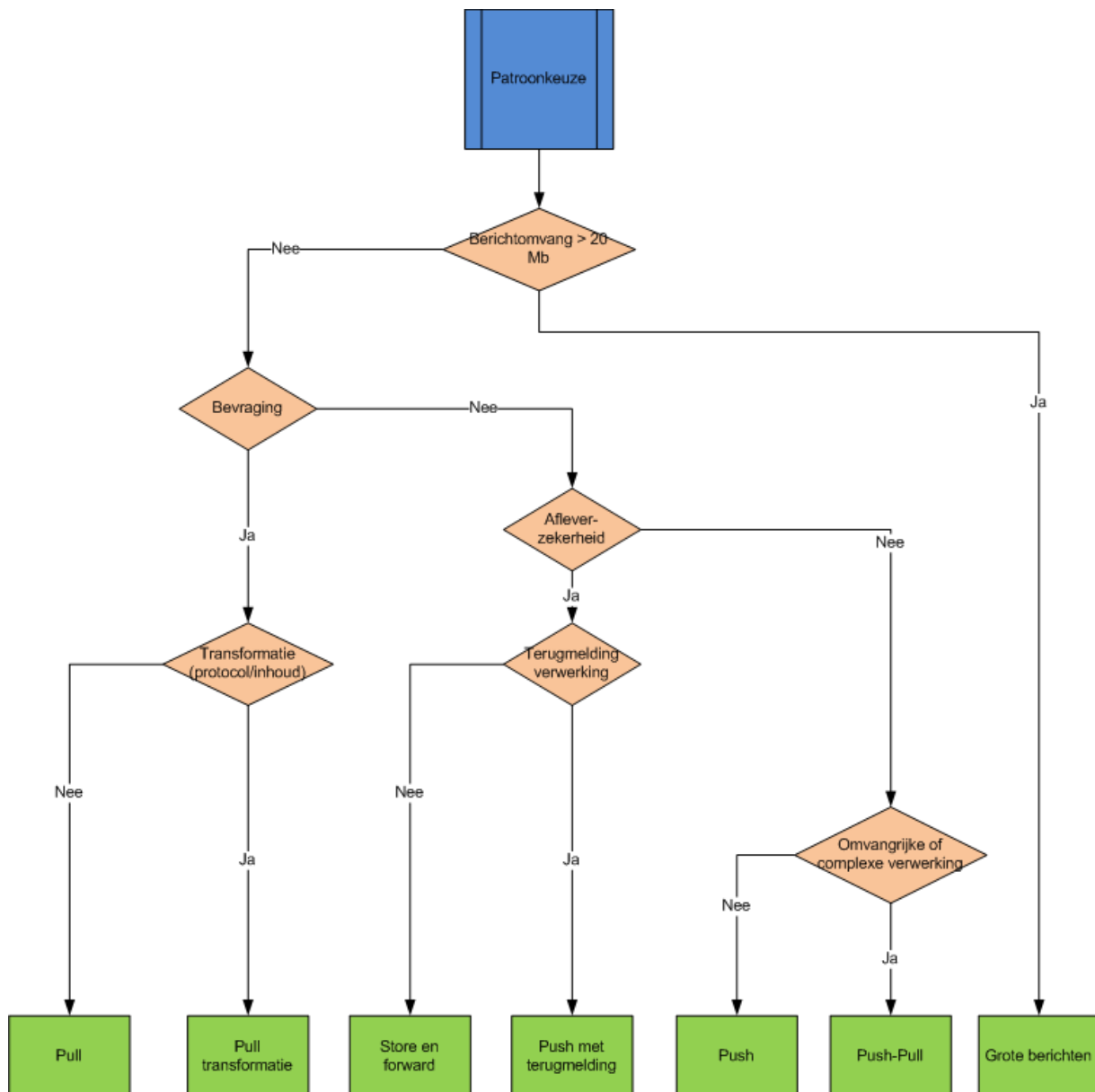
¹ Veelal naar ERP systemen en open standaarden

- Biztalk
- WCF
- Oracle
- Mule ESB

Voorbeeld

Biztalk kent een aantal endpoint varianten die ingezet worden bij het ontsluiten van bijvoorbeeld EBS of het verwerken van files in het berichtenverkeer.

In bovenstaande paragrafen zijn de beschikbare integratiepatronen beschreven. Houd er rekening mee dat deze integratievormen gecombineerd kunnen worden tot ketens van integratiepatronen om een complexe implementatie van integratie mogelijk te maken. De inzet van Biztalk om transacties te modelleren is een voorbeeld van het combineren van integratiepatronen. In onderstaande beslisboom een stappenplan voor het bepalen van het in te zetten patroon.



Afbeelding: Beslisboom integratiepatronen

In onderstaande tabel wordt op basis van de beslisboom toegelicht welk integratieplatform beschikbaar is

	Biztalk	WCF	Oracle OSB	Mulesoft	Database View
Pull	X	X	X	X	X
Pull met transformatie	X	X	X	X	X
Store en forward	X		X	X	
Push met terugmelding	X	?	X	X	
Push	X	X	X	X	
Push pull	X		X	X	
Grote berichten	X				
File endpoint	X		X	X	
Database endpoint		X			X
Webservice endpoint		X	X	X	X
Mail endpoint	X		X	X	
B2B endpoint	X		X	X	

(Open) standaarden

Open standaarden zijn van groot belang voor het inzetten van een service laag. Door inzet van open standaarden wordt een groot aantal van de SOA principes op de juiste wijze geïmplementeerd.

Open standaarden zijn onder te verdelen in twee soorten te weten:

- **Berichtinhoud**, hierbij wordt een gestandaardiseerd informatie- of procesmodel geïmplementeerd. Heeft dus effect op met name de modeltransformatie die nodig zal zijn om van het applicatiespecifieke model om te zetten naar de open standaard.
- **Berichttransport**, denk hierbij vooral aan de verschillende protocollen op het vlak van transport en beveiliging. Heeft dus effect op het vlak van protocoltransformatie tussen de applicatie en de generieke voorzieningen

Voor beide aspecten zijn veel standaarden te noemen. Het is ondoenlijk om deze standaarden te noemen. Echter er zijn een aantal goede bronnen te noemen:

- **Forum standaardisatie** met de lijst voor pas toe of leg uit. Richt zich zowel op inhoud als op transport
- **Oasis** met een aantal belangrijke internationale standaarden. Richt zich zowel op inhoud als op transport
- **W3C** met een aantal webstandaarden. Richt zich voornamelijk op transport

De architecten van de organisatie dienen een lijst op te stellen waarmee aangegeven wordt welke standaarden de voorkeur verdienen. Deze lijst is een belangrijk selectie criterium voor applicaties. Helaas zijn er namelijk zoveel open standaarden beschikbaar waardoor er een selectie gemaakt moet worden omdat anders de inzet van open standaarden vanuit beheer perspectief weinig toegevoegde waarde heeft.

Lijst van open standaarden voor een overheidsorganisatie bestaat veelal uit de pas toe of leg uit lijst. Daarnaast worden een aantal open standaarden toegepast buiten deze lijst zoals CMIS voor document management.

Voorbeeld applicatie

Op basis van een voorbeeldapplicatie worden een aantal aspecten van de servicelaag behandeld. Voor dit voorbeeld nemen we SharePoint als uitgangspunt. Reden om hiervoor te kiezen is dat SharePoint een aantal voorzieningen biedt die kort toegelicht worden in de onderstaande paragrafen.

Webservices

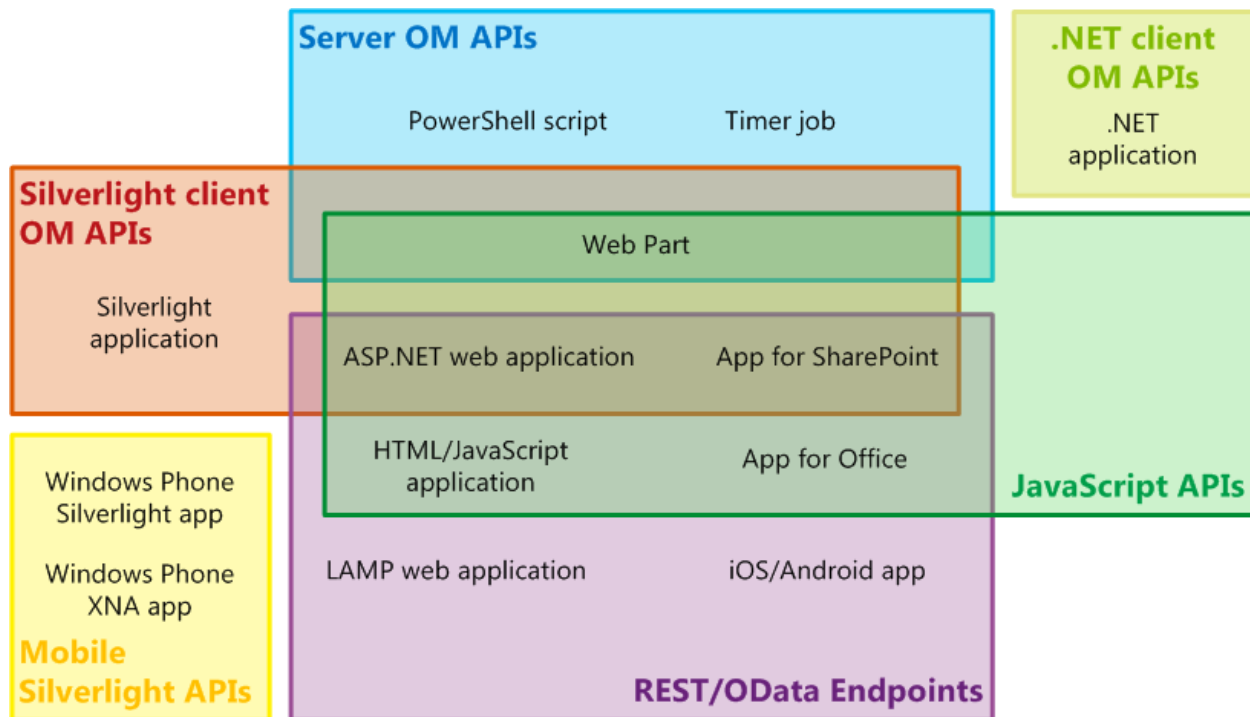
SharePoint heeft een interne structuur voor de opslag van diverse entiteiten zoals lijsten, documenten en gebruikersgegevens. Echter deze interne structuur wordt enkel ontsloten door een user interface en een webservicelaag. Via deze laatste laag is het mogelijk om aanpassingen te doen aan de opgeslagen entiteiten maar ook om de status van de verschillende entiteiten op te vragen. Zie voor achtergrond informatie bijvoorbeeld [http://msdn.microsoft.com/en-us/library/ms458094\(v=office.12\).aspx](http://msdn.microsoft.com/en-us/library/ms458094(v=office.12).aspx)

Door deze opzet vervult Sharepoint met name de rol van Serviceproducer. De inrichtingswijze geeft invulling aan de volgende eisen (zie hfdst 8)

- Gestandaardiseerde service contracten (8.1)
- Service ontkoppeling (8.2)
- Service abstractie(8.3)
- Service ontdekbaarheid (8.4)

Service API

De ontsluiting van SharePoint op basis van services is door de ontwikkelingen op het vlak van standaarden en devices niet meer geheel te ondersteunen op basis van SOAP. Daarom is er een schil ingericht van verschillende API's op basis waarvan de inhoud van SharePoint ontsloten wordt. Onderstaande afbeelding geeft een idee van de verschillende API's



Door deze opzet wordt voldaan aan de volgende eisen:

- Gestandaardiseerde service contracten (8.1)
- Service ont koppeling (8.2)
- Service abstractie (8.3)

Business connectivity services

SharePoint biedt een voorziening voor het ontsluiten van externe gegevensbronnen zodat deze beschikbaar gesteld kunnen worden binnen de portaal omgevingen. Dat beschikbaar stellen binnen de portalen vindt plaats door het eenvoudigweg configureren van interfaces.

De bovengenoemde interfaces bieden de mogelijkheid om op basis van verschillende technologieën en platformen externe gegevens te ontsluiten. Denk hierbij aan verschillende databases maar ook aan een aantal service technologieën. Voor meer informatie zie

<http://technet.microsoft.com/en-us/library/ee661740.aspx>.

Hiermee wordt invulling gegeven aan de volgende eisen:

- Service ont koppeling (8.2)
- Service abstractie (8.3)
- Service hergebruik, mits adequaat ingericht (8.4)
- Service combineerbaarheid (8.7)
- Service ontdekbaarheid, mits adequaat ingericht (8.8)

Lijst van eisen voor applicaties

Op basis van bovenstaande uitwerking kunnen voor de applicaties die door een organisatie aangeschaft worden een aantal criteria gesteld worden. Hierbij is de indeling op basis van de SOA principes gebruikt.

Gestandaardiseerde service contracten

De services zijn beschreven op een gestandaardiseerde manier en worden op een gestandaardiseerde manier ontsloten, bijvoorbeeld met behulp van XML, XSD en WSDL definities.

Service ontkoppeling

Services worden ontkoppeld van de implementatie en kunnen daardoor los van de onderliggende implementatie (door)ontwikkeld worden. Hiermee wordt de servicelaag onafhankelijk van de implementatie binnen de bron- en doelsystemen

Service abstractie

Met abstractie wordt bereikt dat enkel de service interface bekend is bij de afnemers, alle onderliggende aspecten, zoals bijvoorbeeld implementatie, dienen niet relevant te zijn voor de afnemers

Service hergebruik

Services dienen voor meerdere afnemers geschikt te zijn zodat ze worden hergebruikt. Bijvoorbeeld de inzet van open standaarden is een hulpmiddel om hergebruik van services te stimuleren

Service autonomie

Services dienen zo onafhankelijk mogelijk van andere componenten in het ICT landschap te kunnen functioneren

Service statusloosheid

Services moeten vanwege beperkt resourcegebruik en reductie van complexiteit binnen de implementatie status van aanroepen niet vasthouden. Dat stelt eisen aan de inrichting van de servicedefinitie en de berichtinhoud.

Service combineerbaarheid

Services dienen samengesteld kunnen worden zodat er complexere services ontstaan die aansluiten bij de behoefte van de aanvrager

Service ontdekbaarheid

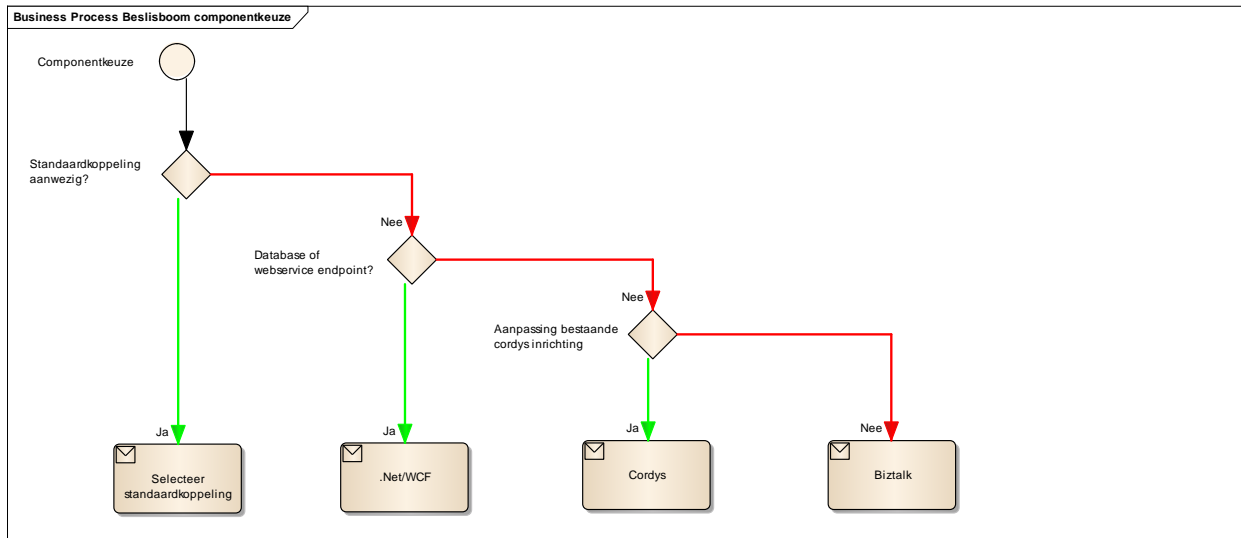
Services dienen gevonden te kunnen dit wordt enerzijds bereikt door het opstellen van metadata van de service maar ook door het inzetten van service registers.

Werkwijze

Bij het uitvragen van een leverancier, bijvoorbeeld bij een aanbesteding kan de volgende werkwijze toegepast worden.

- Specificeer de koppelingen tussen het nieuwe informatiesysteem en de bestaande applicaties en bepaald daarbij welke rol het nieuwe systeem vervult

- Vraag aan de leverancier welke integratiepatronen ondersteund worden door het pakket
- Vraag aan de leverancier welke open standaarden, denk hierbij aan zowel de transport als de inhoudelijke open standaarden, op welke wijze geïmplementeerd zijn.
- Laat de leverancier per koppeling onderstaande beslisboom invullen
- Vanuit de lijst van requirements worden de relevante onderdelen ter beantwoording aan de leverancier voorgelegd



Lijst van eisen/requirements

	Consumer	Producer
Toepassing legt geen beperkingen op aan de inzet van een DNS naam ipv een IP adres ten behoeve van inrichtingsonafhankelijkheid	*	*
Toepassing staat het inzetten van transactiepatronen toe (compensating transaction/Atomic transaction)	*	*
Toepassing ondersteunt de belangrijkste transportprotocollen zoals SOAP/XML/WSDL, REST, JSON, Mobile	*	*
Toepassing heeft mogelijkheden voor het nemen van beveiligingsmaatregelen zoals inzet van certificaten, sessies en handtekeningen etc	*	*
Toepassing kan in zowel asynchrone als synchrone communicatie deelnemen	*	*
Toepassing kan interne datamodelinrichting relateren aan webservice ontsluiting	*	

Toepassing is in staat om tenminste in de pull en de push integratiepatronen deel te nemen	*	
Kan interne datastructuur zonder replicatie gebruik maken van generieke gegevens, dus alleen sleutel opslag	*	
Is de aanwezige servicelaag gebaseerd op een clientside of serverside inrichting of beiden	*	*
Op welke endpoint of adapters kan de toepassing aansluiten (zie 6.4.8)		*
Op welke wijze kan een op services gebaseerde terugmeldfaciliteit ingezet worden		*

