



GEMEENTE TILBURG

Doelarchitectuur Integratieplatform Gemeente Tilburg

Opdrachtgever	Afdelingshoofd Informatie Technologie
Auteur:	Ronald Coenen
Redacteur	Harco van Hees, Rob Klomp, Jan Mutsaers, Tim Schoonebeek, Mike Groels, Niels Spangenberg en Victor Akkersdijk.
Datum	10 juni 2020
Versie	1.0

1 Inhoudsopgave

2	Leeswijzer	3
3	Architectuurkaders	4
4	VNG Realisatie	10
5	Markontwikkeling: Ontkoppeling van applicaties en API's	13
6	Functionele Bouwblokken	15
7	Integratie richtlijnen, principes en patronen	26
8	Beveiliging	29
9	Infrastructuur	32
10	Organisatie	35
11	Kennis en vaardigheden	38
12	Roadmap	40
13	Bijlage Architectuur richtlijnen en principes	43
14	Bijlage II: Definities	52
15	Bijlage III: Bronnen	54

2 Leeswijzer

Dit document schetst de doelarchitectuur van het integratieplatform van de Gemeente Tilburg. In onderstaande leeswijzer is per logisch onderdeel de beoogde doelgroep opgenomen.

Hoofdstuk 3 en 4 beschrijven de uitgangspunten aan de hand van de strategische doelstellingen, bestaande architectuurkaders en standaarden alsmede de ontwikkeling binnen VNG Realisatie, Common Ground.
(Doelgroep: management, architecten)

Hoofdstuk 5 beschrijft de Tilburgse strategie voor integratie op basis van API's. In hoofdstuk 6 worden aan de hand van alle beschikbare bouwblokken de functies van het integratieplatform en relevantie voor de Gemeente Tilburg beschreven.
(Doelgroep: management, architecten, applicatiebeheerders)

Hoofdstuk 7 betreft een inhoudelijke beschrijving van integratiepatronen met verwijzingen naar de integratierichtlijnen en -principes in de bijlage.
(Doelgroep: architecten, applicatiebeheerders)

In hoofdstuk 8 wordt ingegaan op de wijze waarop het integratieplatform invulling kan geven aan de Baseline informatiebeveiliging overheid (BIO), gevolgd door de beschrijving van de benodigde infrastructuur.
(Doelgroep: architecten, applicatiebeheerders)

Hoofdstuk 10 en 11 schetsen de gevolgen van de inzet van een integratieplatform en API strategie voor de organisatie in termen van rollen en benodigde vaardigheden.
(Doelgroep: management, architecten, applicatiebeheerders)

Het document sluit af met een beschrijving van een roadmap voor de inzet van het integratieplatform die ruimte geeft het behalen van korte termijn doelstellingen, mogelijke vervanging van de ODS en MKS alsmede de doorontwikkeling naar de Tilburgse strategie voor integratie op basis van API's.
(Doelgroep: management, architecten)

3 Architectuurkaders

Architectuurkaders bepalen op hoger niveau de kaders waarbinnen de doelarchitectuur van het integratieplatform zal moeten passen. De kaders en uitgangspunten die van toepassing zijn voor de integratiearchitectuur van de Gemeente Tilburg worden in de volgende paragrafen toegelicht.

3.1 Business strategie

De Gemeente Tilburg is een gemeente van verbinding. Bij de Gemeente Tilburg werken meer dan 2000 mensen elke dag samen aan de stad van vandaag en de stad van morgen. Het realiseren van de stad van morgen is een grote uitdaging en de verantwoordelijkheid voor de vormgeving hiervan is verdeeld over 19 afdelingen.

In de huidige maatschappelijke ontwikkeling van het overheidslandschap is een steeds grotere onderlinge afhankelijkheid waar te nemen. Gemeenten maken deel uit van een sterk veranderende samenleving, waarin de steeds verdere digitalisering van de informatievoorziening een van de belangrijkste ontwikkelingen is. Dit leidt ertoe dat verschillende overheidsinstanties en ketenpartners samen moeten werken dan wel taken uitbesteden om een specifiek doel te realiseren. Ook streven Gemeenten naar intensievere samenwerking met andere Gemeenten, regio's en instanties. Verbinden en samenwerken vraagt in de regel om meer informatie-uitwisseling. De business strategie resulteert daarmee in meer informatie-uitwisseling met meer partijen.

3.2 IT-strategie

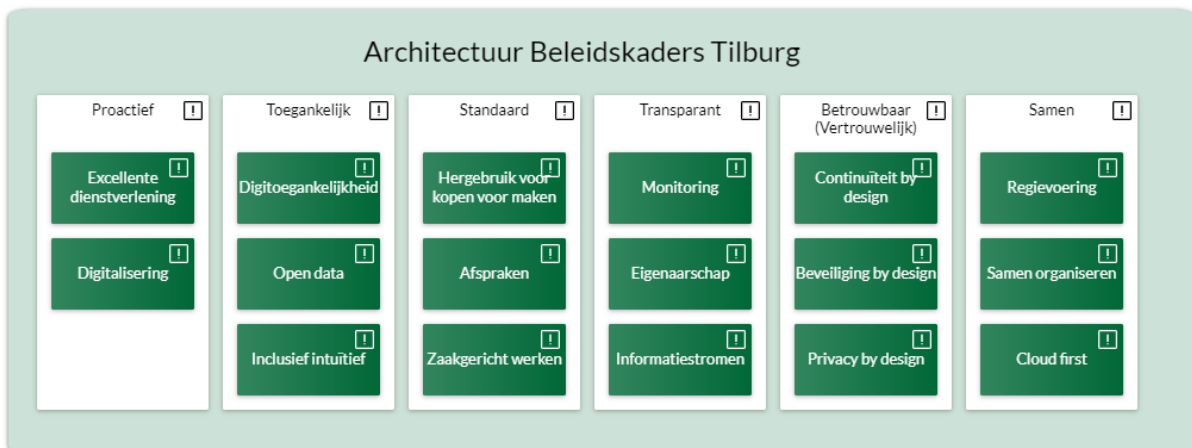
De strategie van de Gemeente Tilburg is de IT-voorzieningen in de vorm van diensten af te nemen van externe leveranciers, bij voorkeur enkele grote marktpartijen uit oogpunt van continuïteit. Voor innovatieve trajecten zal eerder gekeken worden naar kleinere marktpartijen.

Bij het selecteren van nieuwe oplossingen of diensten wordt een 'Cloud first' strategie gehanteerd. Dat betekent Cloud waar mogelijk rekening houdend met technische en juridische (AVG) beperkingen.

3.3 Architectuurprincipes Gemeente Tilburg

Via richtlijnen en gemeente brede principes stelt de Gemeente Tilburg kaders aan veranderingen in de organisatie met de focus op het gemeenschappelijke belang. Met deze principes worden alle belangen in en buiten de organisatie bewaakt. Dit betreft niet alleen IT-beleid, maar ook de richting die is gegeven via de concernopgaven. Daarnaast wordt toezicht gehouden op de afspraken die landelijk zijn gemaakt om samenwerken mogelijk te maken.

De principes zijn onderverdeeld in zes thema's.



Figuur 1: Architectuur beleidskaders Tilburg.

De thema's en daaronder beschreven architectuurprincipes zijn leidend voor de Data-Integratiearchitectuur voor de Gemeente Tilburg.

Veel van de rationale die beschreven worden in de principes hebben betrekking op het vindbaar, betrouwbaar, gecontroleerd en veilig benaderbaar zijn van gegevens. Andere principes behandelen logging en traceerbaarheid van informatie en uitwisseling waar mogelijk via de landelijke gemeente standaarden zoals API's. Het integratieplatform kan in deze principes een relevante rol spelen, mits op de juiste manier ingericht en ingezet. De integratiearchitectuur definieert hiervoor de specifieke kaders.

3.4 Uitgangspunten

- NORA en GEMMA zijn leidend wat betreft architectuurkaders en principes voor zover van toepassing voor de Gemeente Tilburg.
- De ontwikkeling van VNG Realisatie met Common Ground beschrijft een ontwikkeling van een IT-landschap op basis van RESTful API's en ontkoppeling van applicatielogica en data. De doelarchitectuur van het integratieplatform moet deze ontwikkeling ondersteunen.
- Het IT- en applicatielandschap zijn in beweging. Bestaande koppelvlakken met overheidsinstanties zijn gebaseerd op SOAP web services met StUF berichtenverkeer. Ketenpartners zoals de Belastingdienst, Kadaster en Kamer van Koophandel stappen over op RESTful API's voor de uitwisseling van gegevens.
- Veel leveranciers van applicaties bieden nieuwe versies van hun software alleen nog als SaaS oplossing aan. Interactie met andere applicaties en het uitwisselen van data is dan alleen mogelijk via API's.
- Er is een beweging richting Datagedreven werken, gedreven door de behoefte om de effectiviteit en efficiëntie van beleid en de uitvoering te vergroten door besluitvorming meer op basis van feiten en inzichten uit data te ondersteunen. Dit vraagt om ontsluiting van data, ook in grote hoeveelheden en uit de Cloud.
- De ondersteuning van toenemende samenwerking en informatie-uitwisseling stelt hogere eisen aan de flexibiliteit en beheersing van het applicatielandschap. Hiervoor is vanuit VNG Realisatie het Common Ground initiatief gestart. Het idee van Common Ground is dat data wordt losgekoppeld van werkprocessen en applicaties. En data worden bevraagd bij de bron, in plaats van ze veelvuldig te kopiëren en op te slaan. Om data op een efficiënte, gecontroleerde en veilige manier te ontsluiten zijn nieuwe technische oplossingen nodig op basis van marktconforme en overheidstandaarden. Hierbij

staat het gebruik van API's als ontkoppelvlak van data en applicaties centraal.

- Het integratieplatform wordt het standaard platform voor alle berichtenkoppelingen binnen de Gemeente Tilburg. In het bestaande IT-landschap van de Gemeente Tilburg zijn meerdere integratietechnologieën in gebruik. De belangrijkste hiervan en door de Gemeente Tilburg zelf beheerd is de Oracle Service Bus (OSB). Daarnaast zijn er enkele impliciet onderdeel van een applicatie en worden ze onderhouden door de leverancier van de applicatie, zoals JBoss binnen de Makelaarssuite onderhouden door PinkRoccade en OpenTunnel van JNet onderhouden door Exxelence. Elk van deze bestaande oplossingen vormt om meerdere redenen geen basis voor verdere doorontwikkeling tot het integratieplatform van de Gemeente Tilburg. Zo is de Makelaarssuite weliswaar aangeschaft voor het maken van StUF koppelingen, maar biedt de leverancier geen ondersteuning voor het koppelen van niet-StUF berichten met de Makelaarssuite. Het integratieplatform van de Gemeente Tilburg zal op grond hiervan de bestaande standaarden zoals StUF moeten ondersteunen.
- De Gemeente Tilburg moet regie voeren over leveranciers. Regievoering gaat over het in controle zijn. Juist als taken elders worden belegd is het belangrijk dat de gemeente hier de regie op blijft voeren, de gemeente blijft verantwoordelijk. Er wordt daarom geïnvesteerd in de ontwikkeling van competenties die essentieel zijn voor regievoering.
- De Gemeente Tilburg heeft gekozen voor de Microsoft E3 Azure-stack als infrastructuur voor het dataplatform. Het integratieplatform moet data op een betrouwbare, gecontroleerde en veilige manier kunnen ontsluiten naar en van het dataplatform.

3.4.1 Nederlandse Overheid Referentie Architectuur (NORA)

De Nederlandse Overheid Referentie Architectuur (NORA) is bedoeld als richting gevend en sturend instrument. Het bevat kaders en bestaande afspraken voor het inrichten van de informatiehuishouding van de Nederlandse overheid. De toepassing van API's kan worden beschrijven aan de hand van het NORA-vijflagenmodel: Een API geeft invulling aan de informatiebehoefte op het niveau van processen (in de organisatorische laag).

De NORA staat aan het hoofd van een familie van (referentie-)architecturen, waarbij de dochters de moeder verfijnen. Daarnaast kent NORA een aantal thema's die bruikbare architectuurkaders bieden.

3.4.2 Kennisplatform API's - API Strategie

Eén van deze thema's gaat over API's, welke als doel heeft kaders te beschrijven voor de implementatie van een API-strategie. Zie <https://docs.geostandaarden.nl/api/API-Strategie/>.

Binnen dit thema wordt gerefereerd aan de kaders van de Nederlandse API Strategie IIa, die de richtlijnen omvat voor het ontwerpen en gebruik van API's. Het GEONOVUM werkt aan deze toepasbare API-ontwerpregels en normatieve principes, te vinden op: <https://geonovum.github.io/API-Designrules/>. Deze API-ontwerpregels hebben specifiek betrekking op het ontwerpen van RESTful API's. Gegeven de volledigheid en toepasbaarheid van deze regels, zijn ze 1-op-1 opgenomen in de doelarchitectuur van het integratieplatform van de Gemeente Tilburg.

De API Strategie Algemeen beschrijft verder een aantal overwegingen die de basis vormen van de API-strategie van de Gemeente Tilburg. Deze zijn:

- **Gegevens bij de bron beheren.**
Voorkom het dupliceren en lokaal bewerken van centrale gegevensverzamelingen. Dit maakt processen efficiënter en voorkomt dat kopieën van gegevensverzamelingen uit de pas lopen.
- **Processen en gegevens losser koppelen.**
Vaste ketens maken steeds vaker plaats voor lossere koppelingen met een kortere levensduur, zoals in

de visie van Common Ground. Of in smart cities als Amsterdam waar een wisselende groep publieke en private organisaties dynamisch gegevens uitwisselt.

- **Afhankelijkheid van leveranciers verminderen.**
Voorkom dat integraties maatwerk vereisen in de applicatie of afhankelijk zijn van producten of kennis van een specifieke leverancier. Ontwerp API's leveranciersafhankelijk. Gebruik waar mogelijk open standaarden.
- **Wettelijke verplichting.**
Er komen steeds meer wettelijke verplichtingen om herbruikbare overheidsdata te publiceren. In de nieuwe wetgeving wordt het gebruik van API's expliciet verplicht voor bepaalde data. Bijvoorbeeld in de Europese richtlijn die in 2021 zal leiden tot een herziening van de Wet Hergebruik van Overheidsinformatie.

3.4.3 GEMEentelijk Model Architectuur (GEMMA) 2.0 Referentiearchitectuur

De GEMMA 2.0 Referentiearchitectuur is de landelijke referentiearchitectuur voor gemeenten. GEMMA is afgeleid van de NORA en opgezet als richtinggevend en sturend instrument voor ICT-ontwikkelingen binnen de Nederlandse Gemeenten.

3.4.3.1 Referentiecomponenten

Door VNG Realisatie zijn de bouwblokken van de gemeentelijke informatiearchitectuur in kaart gebracht. Deze bouwblokken, de referentiecomponenten, zijn modulair, zelfstandig inzetbaar en vervangbare delen van softwaresystemen. Referentiecomponenten zijn door VNG Realisatie beschreven voor zowel specifieke als generieke functionaliteit. De op de doelarchitectuur integratieplatform betrekking hebbende onderdeel is het Servicebuscomponent. Het integratieplatform van de Gemeente Tilburg moet de functies van het Servicebuscomponent ondersteunen.

3.4.4 Verplichte standaarden en stelsels

In het huidige applicatielandschap van de Gemeente Tilburg wordt volop gebruik gemaakt van bestaande standaarden en stelsels van de Nederlandse overheid. Basisregistraties worden lokaal ontsloten door kernapplicaties en de data gedeeld met verschillende afnemers.

Het integratieplatform van de Gemeente Tilburg moet de standaarden optimaal ondersteunen en optimale oplossingen bieden rondom de registraties. De specifieke kenmerken en eisen worden in de volgende paragrafen toegelicht.

3.4.4.1 StUF Berichtenstandaard

Een belangrijk component van de GEMMA is het Standaard Uitwissel Formaat (StUF). De StUF beschrijft de afspraken over de basisprincipes voor het uitwisselen van gegevens tussen applicaties in het gemeentelijke veld. De StUF is gebaseerd op de open standaarden als XML en SOAP Web Services (WS). Met gebruik van StUF kan het berichtenverkeer tot op zekere hoogte worden gestandaardiseerd. De StUF bevat zelf geen berichten maar beschrijft wel bouwstenen en richtlijnen waarmee berichtstandaarden kunnen worden samengesteld. Het gevolg is dat de StUF implementatie per leverancier verschilt.

De Gemeente Tilburg heeft diverse applicaties van verschillende leveranciers in gebruik die StUF berichtenverkeer ondersteunen. Doordat leveranciers van applicaties de StUF standaarden naar eigen inzicht hebben geïmplementeerd, kunnen deze applicaties niet zonder meer aan elkaar gekoppeld worden. Berichten moeten in bepaalde gevallen worden aangepast. Dit is een expliciete taak van het integratieplatform in plaats van hiervoor maatwerk in de applicaties te laten bouwen. Het integratieplatform moet de verwerking van StUF

berichten ondersteunen, om kunnen gaan met verschillende implementaties van de standaard en wijzigingen hierop.

3.4.4.2 Digikoppeling

Digikoppeling schrijft voor hoe overheidsorganisaties over een digitaal netwerk met elkaar communiceren. De Digikoppeling standaarden (Digikoppeling3.0 WUS en Digikoppeling 2.0 ebMS) beschrijven uitsluitend de communicatie op organisatieniveau. Binnen de organisatie kan de communicatie afkomstig zijn van of bestemd zijn voor allerlei verschillende systemen, maar over de routing binnen een organisatie zegt Digikoppeling niets.

Omdat het inrichten van een Digikoppeling adapter complex is, kiezen organisaties en ook gemeenten veelal voor één Digikoppeling adapter voor de hele organisatie. Bij de implementatie van Digikoppeling binnen gemeenten is gebleken dat het ontbreken van voorschriften voor de communicatie van systemen met een Digikoppeling adapter leidt tot allerlei ad hoc oplossingen die onderling niet op elkaar zijn afgestemd en ook tot het gebruiken van meerdere Digikoppeling adapters door een gemeente. Om die reden is het Koppelvlak Digikoppeling Adapter Intern ontwikkeld.

Het integratieplatform dient te voorzien in een centrale voorziening voor één aansluitpunt Digikoppeling.

3.4.4.3 Basisregistraties en stelselvoorzieningen

Een basisregistratie is een door de overheid officieel aangewezen registratie met gegevens die door alle overheidsinstellingen verplicht worden gebruikt bij de uitvoering van publiekrechtelijke taken. De Gemeente Tilburg maakt voor een groot deel van de werkzaamheden gebruik van basisregistraties zoals Basisregistraties Adressen en Gebouwen (BAG), Basisregistratie Kadaster (BRK) en Handelsregister van de Kamer van Koophandel (HR). Dit kan gaan om het toetsen van vergunningaanvragen, toekennen van subsidies, etc..

Het stelsel van basisregistraties bestaat uit 10 afzonderlijke basisregistraties die onderling met elkaar samenhangen. Om ervoor te zorgen dat er ook inderdaad een in samenhang werkend stelsel ontstaat, zijn afspraken gemaakt. De Stelselcatalogus is een online catalogus die medewerkers van de overheid inzicht geeft in welke gegevens het Stelsel van Basisregistraties bevat, wat ze betekenen en hoe ze met elkaar en met wetgeving verbonden zijn. De inhoud van het Stelselcatalogus kan als Linked Open Data gebruikt worden in werkprocessen.

Momenteel worden de diverse basisregistraties op verschillende en in sommige gevallen door meerdere applicaties ontsloten. Zo ontsluit de Makelaarsuite een aantal landelijke voorzieningen zoals BAG maar sluit de Squit20/20 applicatie van de Gemeente Tilburg zelf ook rechtstreeks aan op landelijke voorzieningen, zoals de LV-BAG. Hiermee wordt het niet alleen lastiger bij te houden waar welke gegevens worden geraadpleegd, maar kan het leiden tot hogere kosten in geval van doorbelasting van gegevens bevragingen.

De Gemeente Tilburg streeft er naar de basisregistraties via één toegangspunt voor de eigen organisatie te ontsluiten en beschikbaar te stellen. Het integratieplatform zorg draagt voor de optimalisatie door middel van caching, traceerbaarheid en logging van de bevragingen.

De stelselvoorzieningen ondersteunen de basisregistraties bij eenvoudige, uniforme, betrouwbare en efficiënte gegevensuitwisseling met hun afnemers. Zo biedt Digimelding een webservice voor het melden van onjuistheid van aangesloten basisregistraties. Gemeenten moeten op de hoogte zijn van gebeurtenissen die voor hun taken relevant zijn. Via Digilevering worden op een uniforme, betrouwbare en efficiënte manier gebeurtenisberichten verstuurd over gegevens in de basisregistraties. Op dit moment is het nog beperkt tot wijzigingen in het HR en in de BAG, maar andere basisregistraties volgen. Digilevering maakt gebruik van Digikoppeling.

Het integratieplatform zal binnen de Gemeente Tilburg als centraal ontvangstpunt en distributieplatform voor Digilevering ingezet moeten kunnen worden.

3.4.4.4 Kernregistraties

Naast landelijke basisregistraties heeft de Gemeente Tilburg ook eigen registraties – kernregistraties - zoals de Kernregistratie Topografie Tilburg. Deze bevat gegevens die voor uiteenlopende doeleinden geraadpleegd worden. Ook voor de kernregistraties moeten aspecten zoals datakwaliteit, beveiliging en privacy gegarandeerd worden.

De kernregistraties worden binnen de Gemeente Tilburg ontsloten via het integratieplatform, waarbij het platform zorg draagt voor de kwaliteit, beveiliging, traceerbaarheid en logging van de gegevensuitwisseling, bijvoorbeeld ten behoeve van de privacy wet- en regelgeving.

4 VNG Realisatie

VNG Realisatie werkt samen met gemeenten aan oplossingen om de gemeentelijke uitvoering te verbeteren. Dit alles vanuit de ambitie om lokale overheden optimaal van dienst te zijn, met zichtbare resultaten voor inwoners en ondernemers. VNG Realisatie doet dit op basis van het door gemeenten vastgelegde meerjarenplan Gezamenlijke Gemeentelijke Uitvoering (GGU) 2020-2024.

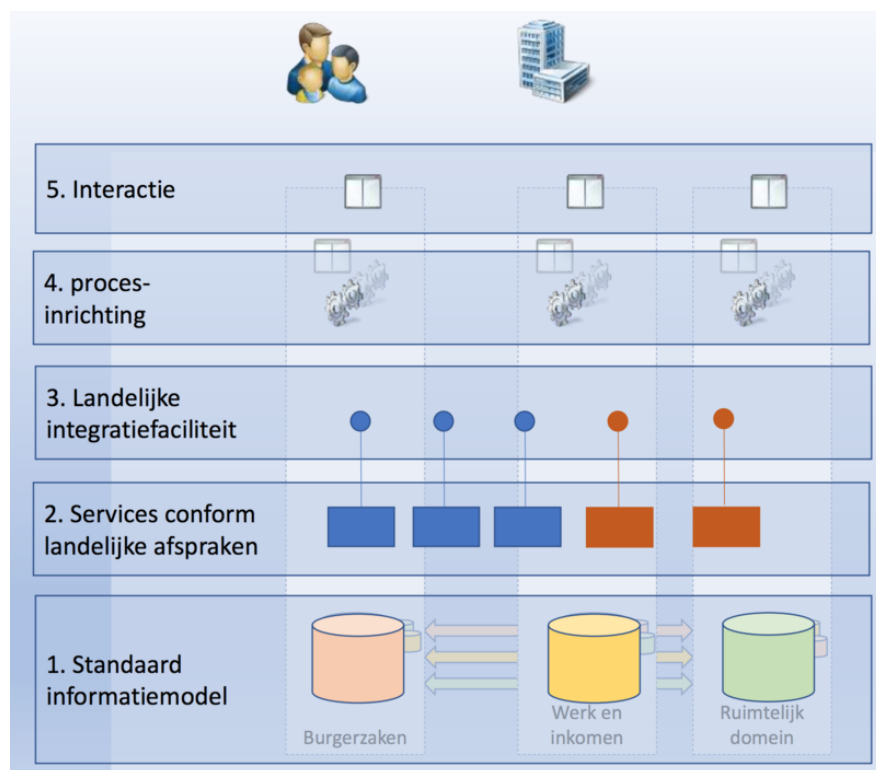
Vanuit VNG Realisatie onder andere het Common Ground initiatief gestart om een andere manier van omgaan met data en het beschikbaar stellen van databronnen vorm te geven. De Gemeente Tilburg is actief betrokken bij en investeert mee in de ontwikkeling van de Common Ground visie en bijbehorende componenten, onder andere via het Innovatie lab.

4.1 Common Ground

Common Ground staat voor een moderne, agile manier van ICT-systemen ontwerpen, bouwen en beheren. Het gaat uit van de gedachte van een Gemeenschappelijk Digitaal Platform en stelt gemeenten in staat sneller te innoveren en kosten te besparen.

Common Ground is het idee om naast de bestaande gemeentelijke ICT-infrastructuur een nieuwe, moderne ICT-infrastructuur te bouwen met gegevenslandschap voor de uitwisseling van gegevens binnen en tussen gemeenten. Met Common Ground kunnen gemeenten hun dienstverlening en bedrijfsvoering ingrijpend vernieuwen vanuit de basis: de gegevenslaag. Daarbij worden data losgekoppeld van werkprocessen en applicaties. En data worden bevraagd bij de bron, in plaats van ze veelvuldig te kopiëren en op te slaan.

Een van de implicaties is dat de architectuur een gelaagdheid kent waarin de gegevens strikt gescheiden worden van de interactie en de procesinrichting. Dit komt tot uiting in het 5-lagenmodel van Common Ground.



Figuur 2: Common Ground 5-lagenmodel

Dat heeft 4 belangrijke gevolgen voor gemeenten:

- Gegevens worden uniform gemaakt;
- Gegevens worden opgehaald met zogenoemde API's;
- Er wordt gewerkt met één gemeenschappelijke integratielaag;
- Data blijven in de bron.

Naast de uitgangspunten wordt binnen Common Ground ook gewerkt aan de realisatie van concreet bruikbare componenten, zoals een standaard informatiemodel met en API specificaties die voldoen aan landelijke afspraken.

4.1.1 Standaard informatiemodel

Voor de implementatie van laag 1 is men afhankelijk van hoe leveranciers hun applicaties hiervoor geschikt maken om te voldoen aan het standaard informatiemodel van Common Ground. Afhankelijk van de leverancier kan dit langer duren of bijvoorbeeld in geval van internationale leveranciers helemaal niet de doelstelling zijn. Voor gegevensbronnen en systemen waarbij dit het geval is moet in laag 2 een vertaling plaats vinden naar de landelijke afspraken.

4.1.1.1 Services conform landelijke afspraken

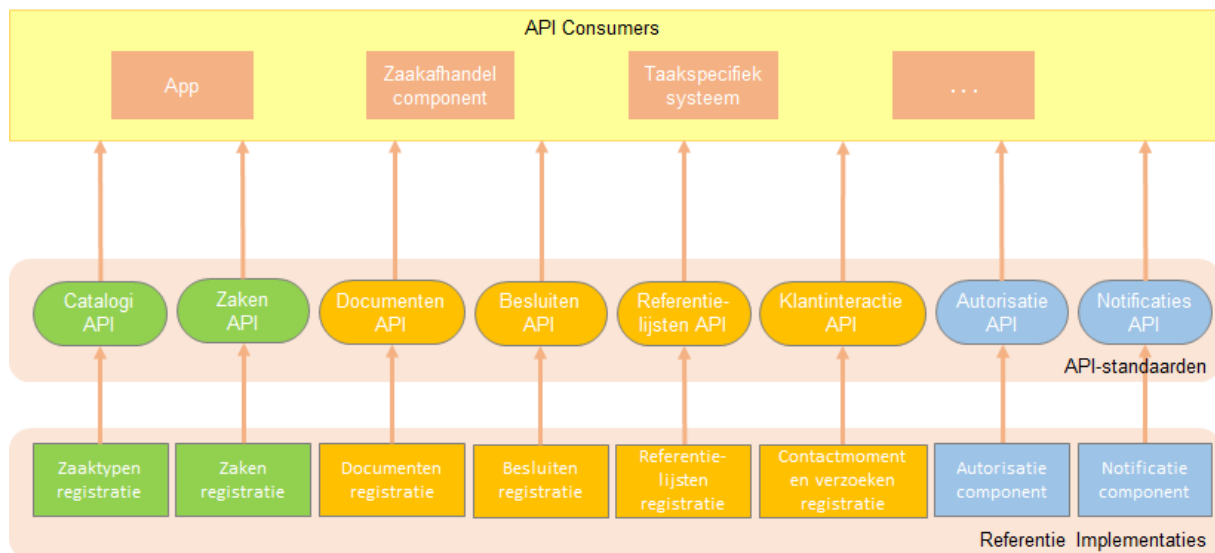
Op laag 2 worden de gegevensbronnen beschikbaar gesteld aan afnemers binnen en buiten de organisatie in de vorm van API's die voldoen aan de landelijke afspraken. Afhankelijk van de bronnen in laag 1 moet er in laag 2 transformatie plaats vinden van het protocol (JDBC naar HTTP), formaat (XML naar JSON) en/of data (mapping van dataveld naar de juiste structuur, naam en/of type).

Het integratieplatform moet deze transformaties tussen de databronnen op laag 1 en het standaard informatiemodel op laag 2 verzorgen.

4.1.1.1.1 API-standaarden voor Zaakgericht Werken

Binnen Common Ground lopen initiatieven om op API gebaseerde standaarden te ontwikkelen als opvolger voor bestaande berichtdefinities. Een eerste initiatief is Zaakgericht werken. De API-standaarden voor zaakgericht werken zijn de opvolger van de Zaak- en Documentservices (ZDS) en helpen bij het (ont)koppelen van zaaksystemen en de registers waar documenten en overige informatie in zijn opgeslagen.

Voor iedere API is er een OAS 3-specificatie, een referentie-implementatie en aanvullende documentatie beschikbaar. De OAS 3-specificaties met beschrijvingen zijn normatief.



Figuur 3: Common Ground API's voor Zaakgericht Werken

Op 18 juli 2019 is versie 1.0.0 RC1 (Release Candidate 1) van de API's voor Zaakgericht Werken gepresenteerd. Vanaf nu kunnen leveranciers en gemeenten de API's inbouwen en gaan de API-standaarden over in de beheerfase.

4.1.1.2 NLX

Laag 3 van de Common Ground architectuur wordt ingevuld met de NLX. NLX sluit aan bij de servicegerichte architectuur die GEMMA, NORA en EIF voorschrijven. NLX is een open source systeem om peer-to-peer gegevens uit te wisselen op basis van federatieve authenticatie, beveiligde connectiviteit en protocollen die faciliteren in een grootschalig API landschap met meerdere organisaties.

NLX biedt API gateway functionaliteiten en vult daarmee een deel van de capabilities van het integratieplatform in. Binnen de Gemeente Tilburg wordt de NLX onderzocht in het Innovatielab. Op dit moment is het nog niet geschikt voor productionele inzet.

Het integratieplatform zal moeten kunnen samenwerken met de NLX configuratie, ongeacht de architectuur daarvan.

Afhankelijk van de ontwikkeling van NLX en inzetbaarheid ervan als generieke API Management oplossing voor de Gemeente Tilburg, kan besloten worden de API Management capabilities van het integratieplatform op met NLX in te vullen.

4.1.1.3 Procesinrichting en interactie

In de procesinrichting en interactielagen kunnen processen en applicaties worden ingericht die gebruik maken van de onderliggende API's. Hierbij kan sprake zijn van maatwerk, eventueel in low-code platformen ontwikkeld.

Ook kunnen door middel van orkestratiefunctieiteit binnen een integratieplatform nieuwe API's worden samengesteld die rijkere functionaliteiten, uit meerdere basis API's bieden.

In het geval van workflow (BPM) processen, kunnen vanuit het proces de basis API's worden gebruikt voor het ophalen en/of opslaan van gegevens die betrokken zijn bij het betreffende proces.

5 Marktentwikkeling: Ontkoppeling van applicaties en API's

In lijn met de Common Ground architectuur, kennen moderne applicatie architecturen gebaseerd op API's een gelaagdheid waarin de gegevens gescheiden worden van de interactie en de procesinrichting.

Bij de Common Ground worden gegevens beschikbaar gesteld aan afnemers binnen en buiten de organisatie in de vorm van RESTful API's die voldoen aan de landelijke afspraken. Dit betekent dat de API specificaties (contracten) vast staan.

De Zaak(typen) registratiefunctie wordt doorgaans geïmplementeerd op basis van een commerciële oplossing, een Zaaksysteem. Niet alle leveranciers van Zaaksystemen zullen altijd hun oplossingen van API's gaan voorzien die exact voldoen aan de specificaties van VNG Realisatie. Afhankelijk van de leverancier kan dit langer duren en in het geval van internationale leveranciers helemaal niet de doelstelling zijn. Er is dus extra flexibiliteit nodig, namelijk ontkoppeling van de Zaak API en het Zaaksysteem.

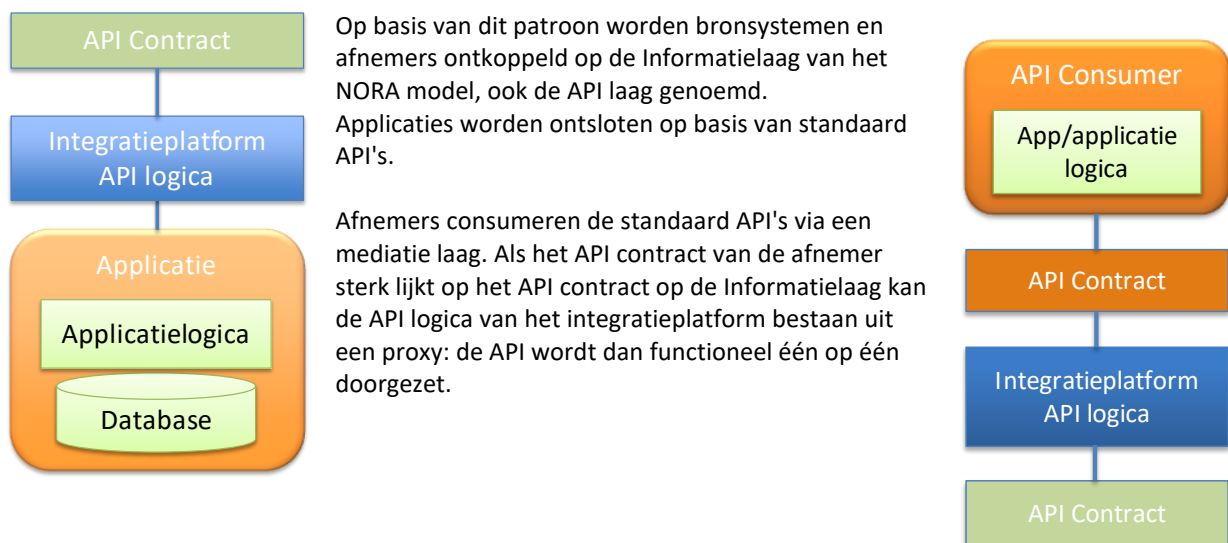
Voor gegevensbronnen en systemen waarbij dit het geval is, zal laag 2 de vertaling verzorgen naar de landelijke afspraken. Soms betekent dit vertaling naar Engelse en applicatieve dataformaten, omzetting van formaat, het afhandelen van specifieke applicatieve beveiliging (inloggen) of samenstelling van bestaande interfaces om alle benodigde data uit de bronapplicatie op te halen.

Hetzelfde geldt voor de consumerende kant van API's: ook daar kan het afhankelijk van het type applicatie nodig zijn de standaard API te voorzien van een extra laag om bruikbaar te maken voor de afnemende applicatie.



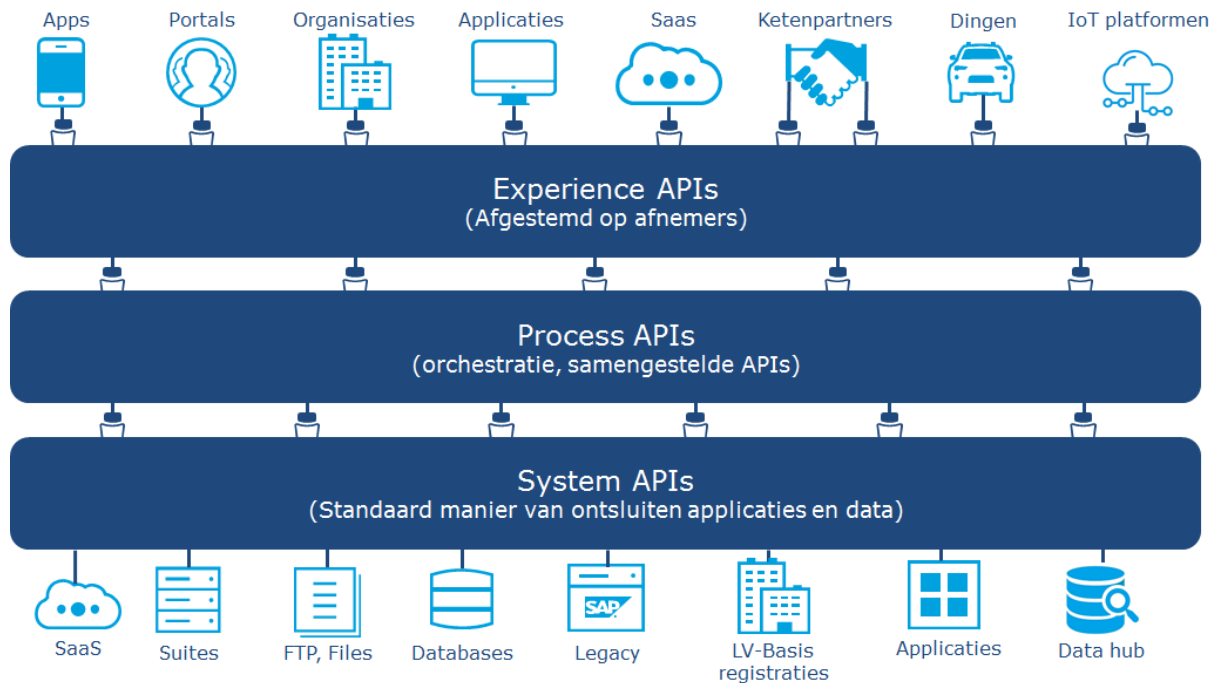
5.1 Tilburgse strategie voor integratie: API-led integratie

De benadering die hier invulling aan geeft wordt de API-Led benadering genoemd. Het patroon lijkt op het Enterprise Service Bus (ESB) patroon in termen van ontkoppeling. De technologie die hiervoor ingezet wordt, kan ook voor ESB oplossingen worden gebruikt, maar is geoptimaliseerd voor RESTful API toepassingen.



Binnen de Gemeente Tilburg kan een functioneel op API's gebaseerd integratielandschap ontstaan dat een koppelvlak is voor zowel interne toepassing als externe partijen waaronder burgers, ketenpartners maar ook de plaats waar IoT Platformen worden ontsloten.

In onderstaande afbeelding is hiervan een vereenvoudigde logische plaat als voorbeeld weergegeven.



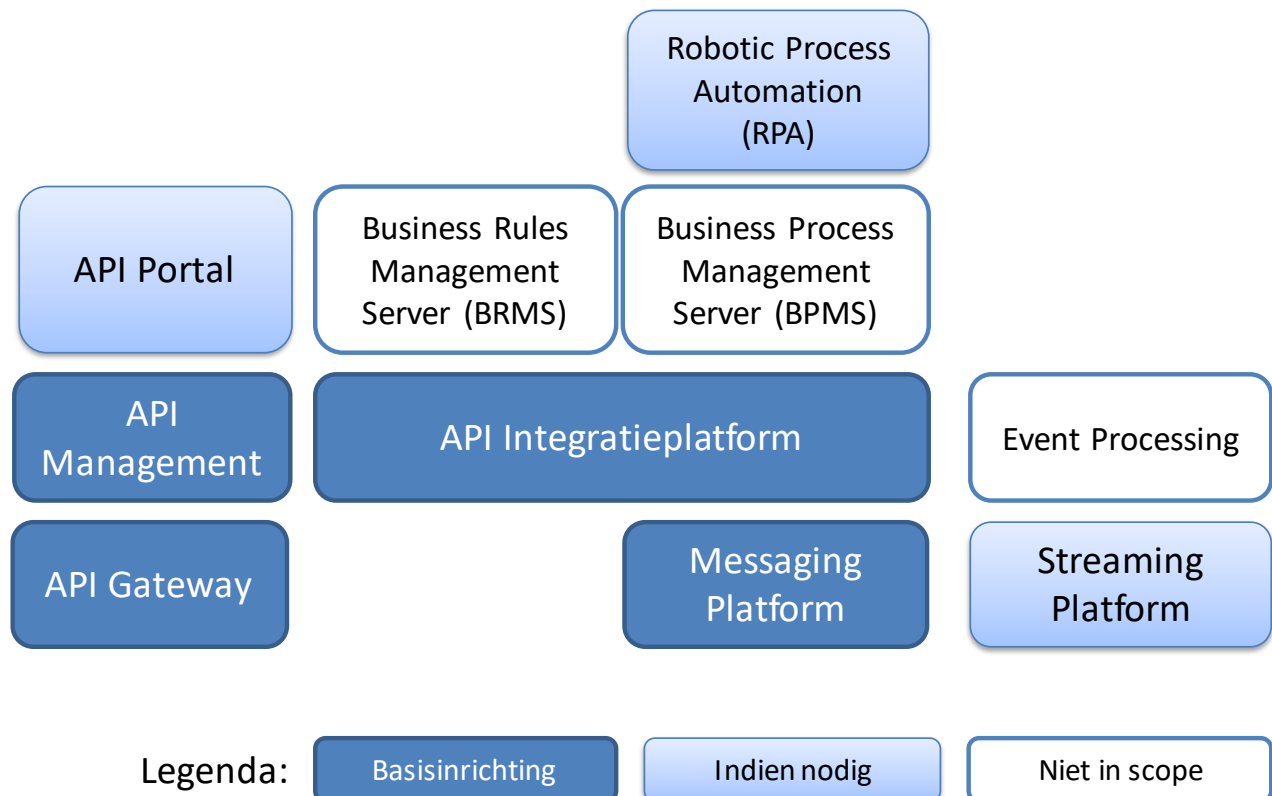
Figuur 4: API-led integratielandschap

In bovenstaande afbeelding vormen de 3 donkerblauwe blokken samen het API integratieplatform.

6 Functionele Bouwblokken

De doelarchitectuur Integratieplatform is een containerbegrip en laat zich omschrijven aan de hand van functionele bouwblokken met de bijbehorende capabilities.

In onderstaand overzicht zijn de verschillende functionele bouwblokken van de doelarchitectuur weergegeven. Deze worden in de volgende paragrafen toegelicht.



Figuur 5: Functionele bouwblokken doelarchitectuur

Niet alle functies van het integratieplatform zijn benodigd voor de realisatie van de integraties, maar kunnen onderdeel zijn van toekomstige uitbreidingen.

- De donderblauwe blokken beschrijven de bouwblokken die onderdeel zijn van de kernfunctionaliteit van het integratieplatform.
- De lichtblauwe blokken bevatten functionaliteit die niet nodig zijn voor de eerste integratietrajecten maar op een later moment nodig kunnen zijn.
- De witte blokken zijn functionele bouwblokken van een integratieplatform maar in de context van Tilburg niet in scope van voorziene integratiescenario's. De bouwblokken worden als onderdeel van de doelarchitectuur wel beschreven om deze te kunnen plaatsen in scenario's waarin deze functionaliteiten een rol spelen

6.1 API Integratieplatform

Het API integratieplatform, ook wel ESB of Servicebuscomponent (GEMMA) genoemd, bevat een specifieke set aan functionaliteiten om de verschillende integratiepatronen op betrouwbare en onderhoudbare wijze te kunnen realiseren.

Voor de Gemeente Tilburg is een kernfunctionaliteit om API's die voldoen aan de landelijke afspraken op te kunnen leveren, onafhankelijk van de onderliggende applicaties. Het kan de vertaling maken van de Zaakgericht Werken Documenten API naar het CMIS formaat waarmee Filenet communiceert. Het biedt de functionaliteit die nodig is om huidige in gebruik zijnde integraties te kunnen vervangen.

6.1.1 Capabilities

De capabilities zijn van een API integratieplatform zijn:

- *Orkestratie*
Een Orkestratie is het in vooraf gedefinieerde volgorde uitvoeren van verschillende acties. Wanneer het noodzakelijk is om data uit meerdere bronnen samen te stellen of meerdere losse acties achter elkaar uit te voeren om een benodigd resultaat te verkrijgen, worden deze in het integratieplatform op basis van Orkestraties geïmplementeerd.
Orkestraties zijn niet te gebruiken voor de implementatie van workflows omdat ze geen toestand vasthouden en geen gebruikersinteractie kennen.
Een open standaard voor de specificatie van Orkestraties gericht op de uitvoering van business processen is BPEL.
- *Routing*
Op basis van verschillende criteria kan het integratieplatform ontvangen berichten naar één of meer ontvangers doorsturen. De criteria kunnen betrekking hebben op de vorm of het type van het bericht maar ook op de inhoud ervan.
Een speciale vorm van routing is filtering, waarbij berichten op basis van criteria al dan niet doorgestuurd worden.
- *Transformatie*
Data moet van een bronformaat naar een doelformaat kunnen worden getransformeerd op basis van deterministische regels en bewerkingen. Transformaties worden gespecificeerd, niet geprogrammeerd. Bij voorkeur biedt het platform een ontwikkelomgeving voor het grafisch specificeren van de transformatie.
Naast transformatie van data, kan transformatie ook betrekking hebben op formaat (XML / JSON) of protocol (HTTP, JMS of FTP).
- *Adapters of connectoren*
Door middel van adapters of connectoren kunnen integraties worden gemaakt via specifieke protocollen, met applicaties of databronnen. Dit kunnen standaard adapters zijn of specifieke maatwerk adapters. Veel gebruikte adapters zijn die voor web services/API's, databases, file transfer en berichtenverkeer.
- *Logging en monitoring*
Het integratieplatform is in staat om het berichtenverkeer in het platform te kunnen loggen naar een te specificeren locatie. Het niveau van logging dient instelbaar te zijn. Bij logging van berichten met gevoelige gegevens (sleutels, persoonsgegevens, medische- en financiële gegevens, etc.) dient men rekening te houden met geldende wet- en regelgeving en hiervoor zo nodig aanvullende maatregelen

te nemen. Het platform voorziet ook in een oplossing om de status van processen en fouten in verwerking te bewaken en op basis van instelbare regels notificaties uit te sturen.

- **Security**

Het integratieplatform voorziet in mogelijkheden van beveiliging van het berichtenverkeer van en naar applicaties en systemen, waarbij gebruik gemaakt kan worden van certificaten voor verificatie van signatures of encryptie.

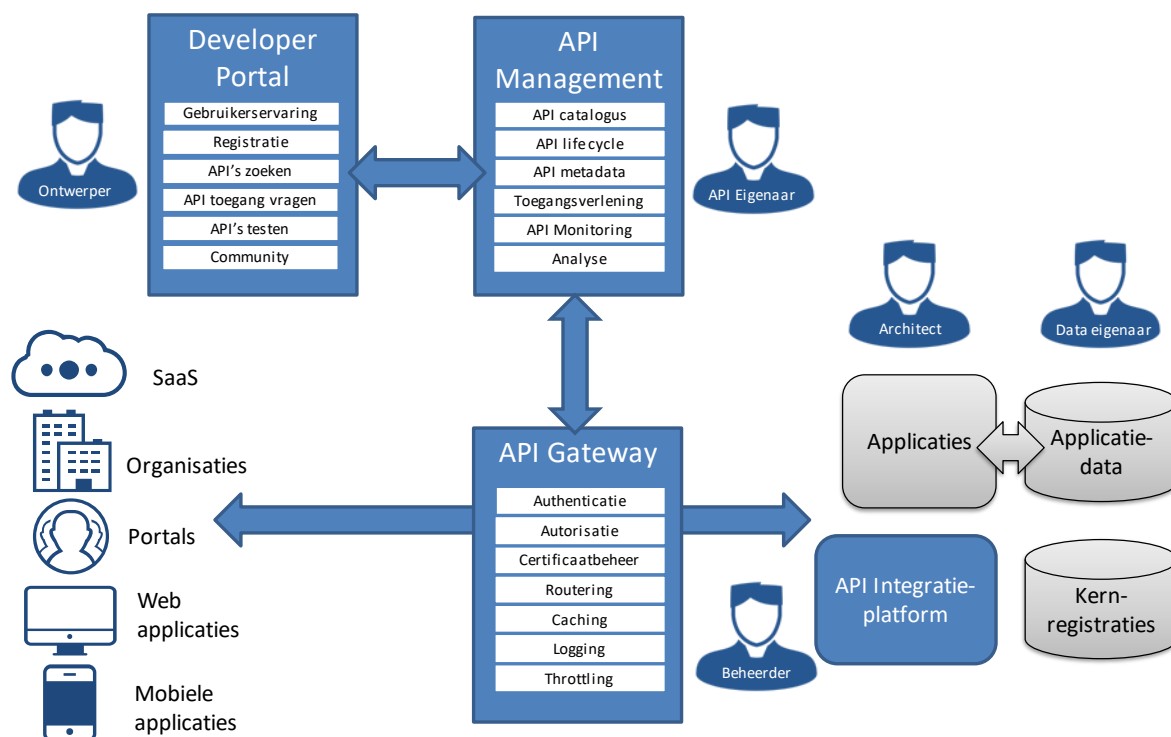
6.2 API Management

API's gaan een steeds groter deel van de integratiebehoefte van de Gemeente Tilburg vormen door de marktontwikkelingen, ver-SaaS-ing van het applicatielandschap en de Common Ground ontwikkeling.

Niet alleen beschikt bijvoorbeeld JDEdwards over een uitgebreide set aan RESTful API's, een SaaS toepassing als Salesforce¹ heeft bij enkel standaard integratiemogelijkheden op basis van RESTful API's.

Kortom, de Gemeente Tilburg gaat steeds meer te maken krijgen met API's en API's in de cloud. Dit zijn API's op gegevensbronnen zoals kernregistraties, API's voor applicaties en SaaS diensten en API's op het integratieplatform. API's bieden toegang tot data en applicatiefunctionaliteit en dat vraagt om beheer. Beheer van grote aantallen vraagt om goede ondersteuning in de vorm van tooling.

Omdat API's toegang geven tot databronnen binnen de Gemeente Tilburg, ook via toegang vanuit het internet, is het beveiligen van de API's één van de kerntaken van API beheer. De API Gateway is hiervoor het verplichte functionele bouwblok. De API Gateway regelt daarbij veilige, gecontroleerde toegang tot de API's van alle cloudapplicaties vanuit zowel de Tilburgse IT-omgeving als vanuit andere cloud applicaties



Figuur 6: API integratie architectuur

¹ Een PoC wordt momenteel uitgevoerd binnen de Gemeente Tilburg voor omnichannel.

Naast specifieke technische componenten ontstaan ook nieuwe taken en verantwoordelijkheden bij het beheer van de API's. Zie voor een beschrijving hiervan Organisatie op pagina 35.

6.2.1 API-Gateway

De belangrijkste taak van de API Gateway is het op een veilige en gecontroleerde manier doorsturen van het API verkeer. Functioneel gezien is de API Gateway transparant en fungeert het als proxy. Daarbij past het de regels (policies) toe op het API verkeer en houdt hiervan de statistieken bij.

6.2.1.1 Capabilities

Authenticatie

De Gemeente Tilburg moet op wettelijke grond alleen geauthentiseerde API aanroepen toestaan en het gebruik van anonieme aanroepen blokkeren. Dit onder andere op grond van de privacy wetgeving bij toegang tot persoonsgebonden informatie. Dit geldt voor zowel API aanroepen van buiten en van binnen de Gemeente Tilburg.

De authenticatie functie van de API-gateway is voor de Gemeente Tilburg onmisbaar om vast te kunnen stellen wat de identiteit is van de aanroeper van een API. De gateway maakt gebruik van een Identity Provider om de identiteit van een API afnemer te kunnen valideren. Dit kan een interne Identity Provider zijn, maar het advies is aan te sluiten op een centrale administratie op basis van Active Directory of LDAP.

- *Autorisatie*

De kerntaak van de API-gateway voor de Gemeente Tilburg is om basis van de vastgestelde identiteit van een API gebruiker, vast te stellen of deze toegang heeft tot de gevraagde API. Geautoriseerde aanroepen worden doorgelaten, niet geautoriseerde aanroepen krijgen een foutmelding dat de aanroep niet geautoriseerd is.

- *Certificaatbeheer*

Certificaten worden gebruikt voor het beveiligen van het API berichtenverkeer. Deze certificaten moeten beschikbaar zijn binnen de API Gateway en daarvoor beschikt deze over een beveiligde omgeving, de trust store.

Technisch Beheer van de Gemeente Tilburg zal actief beheer moeten voeren over de gebruikte certificaten en deze tijdig vervangen voordat ze verlopen of certificaten intrekken als deze gecompromitteerd zijn.

- *Throttling*

De API-eigenaren van de Gemeente Tilburg zullen moeten toezien op eerlijk gebruik van API's. Door middel van afspraken moet worden voorkomen dat één of meer API afnemers de volledige capaciteit van een API opsouperen, met alle mogelijke risico's voor de performance en stabiliteit van de achterliggende applicatie. De mogelijkheid van de throttling functie beschermt het applicatielandschap tegen overbelasting door een maximum aantal aanvragen per geïdentificeerde aanroeper per tijdseenheid toe te laten.

De richtlijn is om voor iedere API een default throttling waarde in te stellen en situaties waarop deze grens wordt bereikt te monitoren. Dit om een optimale dienstverlening naar de API afnemers te kunnen blijven bieden.

- *Routing*

Voor de Gemeente Tilburg is de meest fundamentele taak van de API Gateway het doorsturen van de

API-aanroep naar het endpoint van de achterliggende implementatie. Dit is een éénmalige activiteit bij het ontsluiten van een nieuwe (versie van) een API via de API-gateway.

- *Caching*
Caching verbetert de gebruikerservaring van toepassingen die gebruik maken van API's. Het verkort responsetijden van API's en vermindert de belasting van de onderliggende applicaties en bronsystemen. Een API Gateway biedt de mogelijkheid antwoordberichten in cache opslaan en gebruiken bij volgende uitvragen van dezelfde API.
API caching is alleen zinvol voor relatief statische data objecten die frequent worden opgevraagd. In het geval van de Gemeente Tilburg is dit van toepassing op de basisregistraties en kernregistraties, zoals de GEO kernregistratie.
- *Logging*
Per API aanroep kan informatie over de aanroep en afnemer worden gelogd, waaronder tijdstip, URI, parameters en berichtinhoud. Wat exact wordt gelogd is per API configureerbaar.
Bij logging van API aanroepen met gevoelige gegevens (sleutels, persoonsgegevens, medische- en financiële gegevens, etc.) dient rekening gehouden te worden met geldende wet- en regelgeving en hiervoor zo nodig aanvullende maatregelen te nemen.
De logregels worden weggeschreven in een bestand. De logbestanden kunnen door applicatiebeheerders van de Gemeente Tilburg gebruikt worden voor handmatige zoekacties of input zijn voor verdere geautomatiseerde verwerking.

6.2.2 API Management

API Management gaat over het beheren van de API's. Op basis van de registratie van API's kan de lifecycle van de API beheerd worden. Ook kunnen applicatie ontwikkelaars documentatie opvragen over welke API's er zijn, wat deze kunnen doen en hoe deze aan te roepen.

6.2.2.1 Capabilities

- *Toegangsverlening*
Het verlenen van toegang door het geven van toestemming is een kernactiviteit van API management. Hiermee bepaalt de Gemeente Tilburg wie of welke applicatie toegang krijgt tot welke API – en daarmee de achterliggende data.
- *Monitoring*
Onder de monitoring functie vallen bewaking van de performance, beschikbaarheid, succesvolle uitvoering van de API's. Daarnaast wordt de frequentie van het gebruik van API's bijgehouden. Op basis van ingestelde regels kunnen notificaties worden verstuurd, bijvoorbeeld naar een centraal monitoring systeem en/of mailbox.
Binnen de Gemeente Tilburg kunnen de applicatiebeheerders worden geïnformeerd die zo nodig herstelacties kunnen uitvoeren.
- *Analytics*
Het API Management platform houdt statistieken bij van het API verkeer, per API en/of per afnemer. Deze statistieken kunnen gebruikt worden om inzicht te geven in het gebruik en gedrag van de API's. Op basis hiervan is het ook mogelijk het gebruik van API's door te belasten.
- *API catalogus*
Beschikbare API's kunnen als producten worden geregistreerd in de API catalogus. Hierbij wordt gebruik gemaakt van de gegevens in de API specificatie, zoals documentatie, voorbeeld data en

mogelijke statuscodes.

Binnen de Gemeente Tilburg kunnen analisten en solution architecten van deze catalogus gebruik maken om vast te stellen welke herbruikbare componenten beschikbaar zijn voor nieuwe oplossingen. In de context van NLX zijn hierin ook de beschikbare API's van andere organisaties te vinden.

- **API lifecycle**
Vanuit de regievoering op het API landschap, biedt API lifecycle de middelen aan API eigenaren om de status van API's te bewaken en kwaliteit te waarborgen. Zo kan worden bepaald of een API al in productie is genomen en daarmee beschikbaar voor mogelijke afnemers in de API catalogus.
- **API meta-data**
Het API management platform houdt bij welke API's door de organisatie aangeboden en afgenomen worden. Dit betreft zowel de publieke API's als afgeschermd API's;

6.2.3 Developer portal

Het developer portal vormt een soort van etalage voor API's en ondersteunt ontwikkelaars van applicaties en integraties bij het gebruik van de API's.

Voor de Gemeente Tilburg is dit de functionaliteit om gebruik van API's te organiseren; toegangsverzoeken kunnen geautomatiseerd worden afgehandeld of via een workflow aan de applicatiebeheerders worden voorgelegd.

Om nieuwe functionaliteiten sneller te kunnen ontwikkelen en het risico op wijzigingen achteraf te beperken streeft de Gemeente Tilburg naar een API first aanpak. Ter ondersteuning van de API-first aanpak moet het mogelijk zijn om API specificaties te kunnen testen zonder dat de achterliggende functionaliteit al is geïmplementeerd.

6.2.3.1 Capabilities

De voornaamste capabilities van een API developer portal zijn:

- *Zoeken in API catalogus*
De API portal biedt de mogelijkheid op basis van verschillende criteria in de API catalogus te zoeken, rekening houdend met de autorisatie op basis van de classificatie van de API's.
- *Toegang tot API aanvragen*
Geregistreerde ontwikkelaars kunnen via de portal eenvoudig toegang aanvragen tot API's, waarbij eventueel via een geautomatiseerd proces toegang kan worden verleend en de bijbehorende API keys kunnen worden uitgegeven.
- *API's testen*
De API portal biedt mogelijkheden om API's uit te proberen met beschikbare voorbeeld data (mocking). Hiermee kan de ontwikkelaar o.a. controleren of de API technisch voldoet aan de verwachtingen. Ook API sleutels kunnen worden gevalideerd.

6.3 Event handling

Event handling heeft betrekking op het reageren op gebeurtenissen, zogenaamde events. Een gebeurtenis kan in principe alles zijn en een oorsprong hebben binnen of buiten de organisatie. Voorbeelden zijn een nieuwe bewoner die zich inschrijft of wordt ingeschreven bij een geboorte, een bedrijf dat zich vestigt, een

vuilcontainer die vol is wat wordt waargenomen door een sensor, een ontvangen betaling of juist het uitblijven ervan bij verlopen van de betaaltermijn, een zaak in het zaaksysteem die wordt geopend, bijgewerkt of afgerond.

Events hebben de vorm van digitale berichten. Deze berichten hebben als kenmerk dat ze asynchroon zijn en door meerdere afnemers afgenomen kunnen worden. Asynchroon betekent dat het zenden en verwerken van een bericht niet op hetzelfde moment hoeft plaats te vinden, maar dat kan wel; zender en ontvanger zijn bij asynchrone communicatie van elkaar ontkoppeld in de tijd.

Voor de verwerking van deze berichten zijn specifieke componenten in het integratieplatform voorzien.

Dit zijn:

- Messaging component
- Streaming component

Zowel een messaging component als streaming component bieden een dienst voor het verzorgen van asynchroon berichtenverkeer. De verschillende kenmerken en capabilities worden in de volgende paragrafen beschreven.

6.3.1 Messaging

De messaging component van een integratieplatform, ook wel message broker geheten, is verantwoordelijk voor de uitwisseling van berichten tussen verschillende systemen en applicaties. De systemen en applicaties kunnen verschillen in leverancier, technologie (van mainframe tot serverless) en functionaliteit. Het messaging component biedt een generieke dienst aan applicaties, zodat de applicatieontwikkelaar zich niet meer druk hoeft te maken over platform- en protocolspecifieke aspecten van het uitwisselen van berichten.

Een messaging component kan onderdeel zijn van een Event Driven Architectuur. Voorbeeld van een implementatie hiervan is de verwerking van Digikoppeling notificatie berichten (Digilevering).

Een wereldwijd veel gebruikte standaard van een messaging component is de Java Message Service (JMS). Deze beschrijft met name de interface (API) maar niet de implementatie zelf. Als gevolg daarvan zijn er meerdere implementaties van de JMS beschikbaar, waarvan verschillende Open Source projecten en diverse commerciële oplossingen.

De Gemeente Tilburg zal, zodra deze vereist is ter ondersteuning van een Event handling patroon, een messaging component selecteren die voldoet aan de JMS standaard, de vereiste capabilities biedt en wordt ondersteund door de leverancier van het integratieplatform.

6.3.1.1 Capabilities

De volgende capabilities zijn onderdeel van een messaging component:

- Asynchroon berichtenverkeer
- Point-to-point berichtenverkeer op basis van queues
- Publish Subscribe berichtenverkeer op basis van topics
- Gepersisteerde berichtenopslag
- Gegarandeerde aflevering

Queues voorzien in een point-to-point kanaal, waarbij één of meer zenders berichten naar één afnemer kunnen sturen, welke de berichten na elkaar verwerkt. Een kenmerk van asynchroon berichtenverkeer via een queue is dat zender en ontvanger niet gelijktijdig actief hoeven te zijn. Naast deze eigenschap wordt een messaging component vanwege de snelle reactietijd veel gebruikt in near real-time integratiescenario's, waarbij aspecten als gegarandeerde aflevering, loadbalancing of het opvangen van piekverkeer een rol spelen.

Topics bieden de mogelijkheid voor afnemers zich te abonneren als ontvanger van berichten. Topics bieden standaard geen voorzieningen voor het vasthouden van berichten - als een geabonneerde afnemer niet beschikbaar is, zal deze een op dat topic afgeleverde bericht niet ontvangen.

Ten behoeve van gegarandeerde aflevering is een mogelijkheid beschikbaar om berichten te persisteren (opslaan op schijf), zodat gegarandeerd kan worden dat berichten niet verloren gaan bij herstart van de messaging component. Deze mogelijkheid bestaat ook voor topics waarbij afnemers zich als 'durable subscribers' registreren. In dat geval gedraagt het topic zich op dit aspect als een queue.

Een messaging component is stateless, in die zin dat het de volgorde van berichten niet kan garanderen en ook niet opnieuw kan afspelen voor geselecteerde ontvangers.

6.3.2 Streaming

Door de verregaande digitalisering komt steeds meer data van slimme apparaten en sensoren beschikbaar. Deze data heeft de vorm van berichtenstromen, een continue stroom vanuit elke bron. Een voorbeeld hiervan is sensordata van een IoT apparaat, bijvoorbeeld het gewicht van een afvalcontainer of registratie van verkeerstromen naar, in of vanuit de stad door slimme camera's. Deze data is van belang voor het dataplatform ten behoeve van het datagedreven werken.

Berichtenstromen kennen context, zoals een volgorde op basis van tijdstip. Soms moet verwerking plaats vinden over een set van berichten heen - bijvoorbeeld een specifiek tijdvak om daarvan een gemiddelde te berekenen. Om dit te kunnen moeten berichtstromen voor een gedefinieerde periode worden opgeslagen.

Voor het verwerken van berichtenstromen is een streaming component in het integratieplatform voorzien. Naast het optimaal verwerken van berichtenstromen maakt een streaming component het mogelijk in de berichtenstromen te zoeken of voor een specifieke afnemer deze opnieuw af te spelen.

De meest concrete behoefte binnen de Gemeente Tilburg voor een streaming functie ligt bij het dataplatform. Een streaming oplossing op basis van een cloud-native technologie heeft daarbij de voorkeur vanuit schaalbaarheid en optimale integratie met het dataplatform.

6.3.2.1 Capabilities

De volgende capabilities worden geboden door een streaming component:

- Berichtenstromen
- Context van berichten
- Gepersisteerde berichtenopslag voor onbepaalde termijn
- Zoeken in berichtenstromen
- Opnieuw afspelen van berichtstromen

6.3.3 Event Processing

Event Processing is erop gericht patronen in data van uiteenlopende vorm en bronnen te herkennen en daarop geautomatiseerd acties te ondernemen. De klassieke vorm van Event Processing staat bekend als Complex Event Processing (zie kader).

In de context van streaming data is Event Stream Processing (ESP) de logische opvolger. ESP is het real-time verwerken van streaming data uit één of meerdere bronnen. ESP is gericht op een continue stroom in plaats van individuele events. ESP wordt op grond van deze eigenschap veel gebruikt in IoT edge oplossingen (verwerking van de datastroom dicht bij de sensor), maar ook voor andere event processing doeleinden.

Voor de Gemeente Tilburg is inzet van Event Stream Processing afhankelijk van de beoogde use cases, relevant voor het initiatief van het datagedreven werken en inzet op het dataplatform. Een cloud-native oplossing heeft daarbij de voorkeur vanuit de optimale integratie met het dataplatform.

Complex Event Processing (CEP)

is het combineren van verschillende losse gebeurtenissen (events) tot concrete acties. Een event kan zijn het resultaat van een handeling zijn, het wijzigen van een toestand of het overschrijden van een grenswaarde. Bij CEP kunnen events in verschillende volgorde optreden en zelfs volledig onafhankelijk van elkaar zijn. De specifieke relatie tussen de events waarop moet worden geacteerd wordt in de CEP oplossing gedefinieerd. Het is bedoeld om te reageren op situaties op het moment dat ze optreden en niet op analyse achteraf. Het near-realtime reageren op events maakt CEP een event-gedreven component. CEP wordt veel gebruikt in toepassingen om fraude te detecteren en daarmee te voorkomen. Door de opkomst van sensoren en streaming data, is er minder focus op CEP oplossingen.

6.3.3.1 Capabilities

De belangrijkste capabilities van een ESP oplossing zijn:

- Event patroon detectie.
- Event filtering.
- Event aggregatie en transformatie.
- Integreren met event en streaming bronnen.
- Ontwikkelomgeving voor het configureren van event patronen.
- Logging van events en acties hierop.
- Debugging mogelijkheden.

6.4 Business Process Management

Business Process Management(BPM) is een methode om processen binnen een organisatie te beschrijven, ontwerpen, analyseren, optimaliseren, uit te voeren en te bewaken. Processen die zich daartoe lenen kunnen worden uitgevoerd of ondersteund in een IT oplossing. Een mogelijke implementatie van zo'n IT oplossing is een Business Process Management Server (BPMS). BPM processen kunnen via API's worden aangeroepen en BPM processen kunnen API's aanroepen voor de uitvoering van specifieke taken. Een BPMS is Stateful daar waar een API integratieplatform per definitie stateless is; ook een orkestratie van meerder API's is een stateless proces.

Binnen de Gemeente Tilburg wordt een BPMS op beperkte schaal ingezet voor de uitvoering van het Identity Provisioning proces. Er vindt momenteel geen doorontwikkeling plaats van deze oplossing. Enerzijds vanwege de ervaren complexiteit van de gekozen oplossing en het ontbreken van kennis hiervan binnen de organisatie, anderzijds omdat de meeste processen gekoppeld worden aan een zaak en daarmee binnen het zaakstelsel worden afgehandeld. Ook kunnen processen binnen taakspecifieke applicaties worden uitgevoerd.

Op grond van bovenstaande wordt een BPMS niet ingericht als kernfunctionaliteit van het integratieplatform binnen de Gemeente Tilburg. Het component is ter referentie opgenomen. Dit om expliciet uit te sluiten dat Stateful BPM en workflow processen op het API-integratieplatform ingericht gaan worden.

6.4.1 Capabilities

Een BPMS oplossing bevat minimaal de volgende capabilities:

- Grafische modeleringsomgeving.
- Modelering op basis van BPMN 2.0.
- Uitvoering van in BPMN gespecificeerde processen.
- Interactie met interfaces op basis van API's.
- Logging van de uitvoering van processen.
- Analyse van uitvoering van processen

6.5 Robotic Process Automation

Robotic Process Automation (RPA) is een integratiestijl waarbij de UI van een applicatie het koppelvlak voor de integratie is. Het is een toepassing van software (bots) waarbij de primaire doelstelling is het automatiseren van repeterende handmatige handelingen. Daarbij beschikt het over ingebouwde beslislogica voor de uitvoering van processen. De onderliggende applicaties hoeven hiervoor niet te worden aangepast.

In de context van het integratieplatform is de belangrijkste toepassing van RPA het ontsluiten van applicaties die geen andere betrouwbare mogelijkheid tot integratie (zoals API's) bieden, waarbij ook data in de applicatie moet kunnen worden opgeslagen. RPA kan bijdragen de levensduur van een applicatie verlengen door API's hierop mogelijk te maken om de onderliggende logica en data te ontsluiten.

Anders dan bij API's en webservices kennen (grafische) gebruikersinterfaces geen contracten of formele specificaties. Dat betekent dat op een andere manier moet worden bewaakt dat de invoer en uitvoer van het interface correct is en niet als gevolg van een versiewijziging is aangepast. Dit vraagt om passend beheer van de RPA integraties. Naar verwachting worden de RPA oplossingen op dit vlak verder geoptimaliseerd en slimmer om het beheer te minimaliseren.

Doordat de integratie via de interface laag loopt van een applicatie, heeft dit gevolgen voor de belasting en performance van het systeem. Schaalbaarheid van de applicatie is een van de succesfactoren van RPA oplossingen. De API's die RPA beschikbaar stelt moeten worden beveiligd door de API Gateway waarbij throttling de ontsloten applicatie moet beschermen tegen een te hoog aantal aanroepen.

6.5.1 Capabilities

Een RPA oplossing beschikt minimaal over de volgende capabilities:

- Intuïtieve omgeving voor het programmeren van de bots.
- Schermopname mogelijkheden ten behoeve van programmeren van de bots.
- Optische karakter herkenning (OCR) voor tekstherkenning in bitmaps.
- Debugging mogelijkheden.
- Role based security mogelijkheden.
- Beslisregels ter ondersteuning van procesuitvoering en foutafhandeling.
- API's voor het aanroepen van RPA processen/scherminteracties.
- Analytics voor het analyseren van het gebruik en de performance.
- Bots version control

6.6 Business Rules Management

Het GEMMA Gegevenslandschap heeft het component Business Rules Management opgenomen in de laag van de Procesinrichting.

Business rules of bedrijfsregels zijn organisatie specifieke beslisregels. Deze regels worden veelvuldig (vaak impliciet) toegepast in applicaties voor de uitvoering of ondersteuning van processen. Voorbeelden van dergelijke regels zijn het vaststellen van de hoogte van toelages, het verlenen van een vergunning, etc.. De hoogte van de toelages, de criteria voor het verlenen van een vergunning wijzigen gedurende de tijd, meestal door veranderende wet- en regelgeving.

Business rules management gaat over het centraal vastleggen, beheren en beschikbaar stellen van bedrijfsregels. Dit betekent dat alle logica voor de toepassing van een bedrijfsregels op één plaats is vastgelegd en beschikbaar is voor alle applicaties die dit nodig hebben. Op deze manier kunnen de regels worden aangepast als de situatie daarom vraagt, zonder dat de applicaties zelf hoeven te worden aangepast. De bedrijfsregels zijn hiermee ontkoppeld van de applicatie logica.

Decision Model and Notation

Een standaard voor het modeleren van beslisregels is opgesteld door Object Management Group (OMG), vastgelegd in de Decision Model and Notation (DMN) Publication. DMN is ontworpen om proceslogica in beslis-service te kunnen uitvoeren. Het heeft een relatie met BPMN. Het biedt een standaard notatievorm voor beslistabellen, welke de meest voorkomende vorm van beslisregels in een BRMS vormen.

Een business rules management system (BRMS) is een softwaresysteem waarin beslisregels zoals bedrijfsregels kunnen worden ontworpen, centraal opgeslagen en in een runtime-omgeving uitgevoerd. De beslisregels worden door middel van API's aangeroepen.

Voor de Gemeente Tilburg is de inzet van een BRMS op dit moment niet relevant omdat er geen vorm wordt gegeven aan de procesinrichting zoals beschreven in het Gemma Gegevenslandschap en Common Ground, voor zover dit niet al in de bestaande applicaties is opgenomen.

Op korte termijn wordt een dergelijke behoefte ook niet voorzien, maar is het component ter referentie opgenomen. Dit om expliciet uit te sluiten dat bedrijfsregels in de API integratiecomponent worden geïmplementeerd.

6.6.1 Capabilities

Een BRMS oplossing biedt minimaal de capabilities:

- Grafische ontwikkelomgeving voor het ontwerpen van beslisregels op basis van DMN.
- Middelen om beslisregels te valideren.
- Simulatiemogelijkheden om nieuwe of gewijzigde beslisregels te testen.
- Gecontroleerde uitrol naar productie van nieuwe en gewijzigde beslisregels.
- Logging van uitvoering van beslisregels.
- Beslisregels via API's aan te roepen.

7 Integratie richtlijnen, principes en patronen

Integratieprincipes en richtlijnen dragen bij aan de realisatie en uitbouw van een effectief, onderhoudbaar en kostenefficiënt integratielandschap. Ze geven sturing aan de keuze voor de in een specifiek scenario toe te passen integratiepatroon bij nieuwe toepassingen en/of vervanging van bestaande applicaties. Ontwerpen die als onderdeel van een Project Start Architectuur of in een ontwerpdocument worden opgesteld, dienen te worden getoetst aan deze principes en afwijkingen te worden onderbouwd.

In Bijlage 1 zijn de voor de Gemeente Tilburg van toepassing zijnde integratierichtlijnen en -principes opgenomen.

7.1 Integratiepatronen

De volgende integratiestijlen dienen op grond van de hierboven genoemde uitgangspunten door het integratieplatform te worden ondersteund:

- Request - reply
- Publish -subscribe
- Event notificatie
- File transfer

7.1.1 Request-reply

Het Request-reply patroon is synchroon van aard en bestaat uit een combinatie van een verzoek met antwoordbericht waarbij de aanvrager blijft wachten op het resultaat van zijn verzoek. Het vereist daarbij dat beide partijen actief zijn op het moment van het versturen van de berichten. Het is een point-to-point patroon: er zijn in een interactie slechts één zender en ontvanger actief.

Request-reply is het basispatroon van SOAP webservices en RESTful API's. De communicatie vindt plaats op basis van het HTTP(S) protocol. Er is hierbij sprake van een actieve sessie. Hoewel er theoretische oneindig grote berichten via het request-reply patroon kunnen worden verstuurd, dwingen technische limieten van o.a. het integratieplatform om de berichten niet te groot te laten worden. Voor de uitwisseling van grote bestanden geniet het file transfer patroon of het Claim Check patroon de voorkeur.

7.1.2 Publish-subscribe

Het publish subscribe patroon is asynchroon patroon. Zenders en ontvangers zijn niet van elkaars bestaan op de hoogte. Ze kunnen ook in de tijd van elkaar ontkoppeld zijn, waarbij er sprake is van een berichtenwachtrij (queue). Binnen dit patroon bestaan er twee typen:

1. Point-to-point - queue
2. One-to-many - topic

Ad1: Een bericht kan slechts éénmaal van een queue worden afgelezen. Iedere queue kent dus slechts één afnemer. Een queue kan meerdere zenders hebben, waarmee een many-to-one patroon ontstaat.

Ad2: Topics hebben als kenmerk dat ze 0, 1 of meer afnemers (abonnees) kunnen hebben. Eenzelfde bericht kan daarmee meerdere keren worden ontvangen. Met topics kunnen berichten gelijktijdig naar meerdere verschillende afnemers worden gestuurd. Een topic kan ook meerdere zenders hebben, waarmee een many-to-many patroon ontstaat.

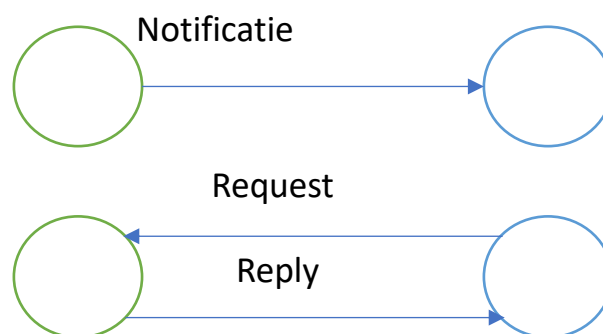
De communicatie onder het publish-subscribe patroon wordt verzorgd door een message broker. Publish subscribe patronen worden vooral gebruikt in integraties waarbij een update in één systeem moet worden gedeeld met andere systemen. In de traditionele EAI integratie implementaties worden hele dataobjecten met

andere afnemers gedeeld. Dit patroon kan bij grote dataobjecten resulteren in onvoorspelbare performance en risico op verlies van datakwaliteit.

In een API landschap wordt het event notificatie patroon hiervoor toegepast, waarbij alleen de minimale informatie over de wijziging (het event) wordt gedeeld. Het ontvangende systeem kan op basis van het event zelf beslissen of het de rest van de informatie via een API wil opvragen.

7.1.3 Event Notificatie

Event notificatie is een combinatie van het publish-subscribe en request-reply-patroon. Een event wordt via een message broker of streaming server gedistribueerd naar alle afnemers die hierin geïnteresseerd zijn. Het bevat de minimaal benodigde hoeveelheid informatie over het dataobject waar het event betrekking op heeft én om via een API call het hele dataobject uit het bronsysteem op te halen.



Figuur 7: Event notificatie patroon

Bij event notificatie is het niet relevant of events elkaar inhalen, want bij het ophalen van de betreffende data objecten wordt altijd de meest actuele versie opgehaald. Een implicatie van het Event Notificatie patroon is dat iedere afnemer voor zichzelf door middel van een API call de gewijzigde data moet ophalen. Slimme API caching voorkomt dat het bronsysteem wordt overladen door dezelfde API calls.

7.1.4 File Transfer

File transfer of bestandsuitwisseling is het patroon waarbij een bestand wordt uitgewisseld tussen zender en ontvanger. Dit bestand kan intern gestructureerd zijn op basis van XML of JSON, maar ook een kolommenindeling met vaste lengtes of afgescheiden door speciale tekens (bijvoorbeeld CSV). Bestandsuitwisseling is een passend integratiepatroon bij batch georiënteerde processen zoals database replicatie en ETL processen.

Een betrouwbaar en veel gebruikt transport mechanisme voor bestandsuitwisseling is SFTP of FTPS. Bestanden kunnen ook als attachment bij een HTTPS bericht worden verstuurd. Dit protocol voorziet niet expliciet in bestandverwerking zoals FTP dit wel doet, er is extra logica bij de zender en ontvanger nodig om het betrouwbaar over te sturen en er uiteindelijk weer een bestand van te maken.

7.2 Canoniek Datamodel

Een meerwaarde van het systematisch integreren van applicaties is het realiseren van ontkoppeling. Een integratieplatform zorgt ervoor dat de gegevens van zowel de zendende als de ontvangende applicaties vertaald worden naar een applicatie-onafhankelijk model. Hierdoor hoeven applicaties geen kennis te hebben van of rekening te houden met elkaars datamodel. En doordat applicaties geen kennis hebben van elkaars gegevensmodel hoeft bij vervanging van een applicatie, andere applicaties niet te worden aangepast.

Een veel gebruikt patroon daarbij is ontkoppeling via een Canonical of Common Data Model (CDM). Een CDM standaardiseert de definities van gegevens die worden uitgewisseld binnen een organisatie. Het is gebaseerd op organisatie-brede gegevensdefinities (glossary). Het staat los van het datamodel van een specifieke (kern)applicatie. Het belangrijkste doel van organisatie-brede gegevensmodel is het geven van een eenduidige betekenis aan de gegevens die worden gebruikt binnen het bedrijf. Het kan daarom worden gezien als een belangrijk onderdeel van de Enterprise Architectuur van een organisatie.

De ontwikkeling van een CDM begint klein en met alleen die objecten die nodig zijn voor de eerste projecten en/of koppelvlakken. Het zelf definiëren van een CDM vergt veel inspanning, daarom wordt zo veel mogelijk gebruik gemaakt van standaarden die (delen van) een dergelijk model definiëren. De Basisregistraties, StUF en Common Ground bieden hiervoor bouwblokken. Alleen de bruikbare delen die van toepassing zijn voor de eigen organisatie worden overgenomen.

Uitbreidingen op het CDM worden met behulp van domein experts op basis van een aantal eenvoudig te begrijpen ontwerpprincipes gemaakt. Wijzingen op CDM objecten dienen zo veel mogelijk te worden uitgesloten omdat die grote impact kunnen hebben op bestaande interfaces.

Het CDM referentie model moet zijn opgebouwd rondom de volgende principes:

- **Begrijpelijk.** Het CDM geeft de gebruikelijke taal binnen de Gemeente Tilburg weer. Het voegt geen nuttelose abstractie toe. Het is modulair opgezet in eenvoudig leesbare overzichten.
- **Onafhankelijk.** Het CDM is platform/applicatie onafhankelijk. Dat wil zeggen dat de namen en definities begrepen kunnen worden zonder enige kennis van de technische systemen waartussen de gegevens worden uitgewisseld.. Het geeft eerder de integratiebehoefte tussen de applicaties weer.
- **Stabiele kern.** De kern entiteiten en relaties zijn zodanig algemeen dat hierin geen wijziging verwacht hoeft te worden.
- **Uitbreidbaar.** Het CDM is zo ontworpen dat het gemakkelijk uit te breiden is naar aanleiding van toekomstige ontwikkelingen.
- **Commutatieve opzet.** Het CDM is zo opgezet dat er geen specifiek startpunt is voor een bewerking.
- **Scheiding van verantwoordelijkheden.** Het CDM is zo opgezet dat er niet meerdere personen verantwoordelijk (eigenaar) kunnen zijn voor (delen van) een entiteit.

Objecten die worden beschreven in het CDM worden gebruikt in de diverse interface definities zoals XSD, WSDL of OAS. (OpenAPI Specification, ook bekend als Swagger). Aanvullend kunnen de definities ook worden gebruikt bij beschrijving van data entiteiten binnen een data warehouse omgeving.

8 Beveiliging

Één van de kenmerken van een integratieplatform is dat het toegang heeft tot veel applicaties en databronnen en dat het data in verschillende vormen tussen aanbieders en afnemers - zowel binnen als buiten de organisatie - transporteert. Dit stelt specifieke eisen aan de beveiliging ervan.

Informatiebeveiliging is het proces van vaststellen van de vereiste betrouwbaarheid van informatiesystemen in termen van beschikbaarheid, integriteit en beschikbaarheid (B.I.V.) en het treffen, onderhouden en controleren van een samenhangend pakket van bijbehorende maatregelen.

Vanaf 2020 geldt voor de gehele Nederlandse overheid één uniform normenkader voor informatiebeveiliging, de Baseline Informatiebeveiliging Overheid (BIO). De BIO is gericht op de Informatiebeveiliging binnen de gehele gemeente. Het beschrijft vooral de maatregelen en handreikingen om de maatregelen in te vullen. De basis voor de inrichting van het beveiligingsbeleid is NEN-ISO/IEC 27001:2017+A11:2020 nl. De NEN-ISO/IEC 27002 standaard is een “best practice” van beveiligingsmaatregelen om informatiebeveiligingsrisico’s aan te pakken met betrekking tot vertrouwelijkheid, integriteit en beschikbaarheid van de informatievoorziening. Voor het opstellen van de beveiligingsaspecten van het integratieplatform zijn de afgeleide principes voor beveiliging van de NORA als kaders gehanteerd.

8.1 Beschikbaarheid

Op basis van de eisen die voortvloeien uit het plan voor bedrijfscontinuïteit wordt bepaald welke delen van het integratieplatform hoog beschikbaar moeten zijn uitgevoerd om single-points-of-failure te vermijden.

8.1.1 On-premise

Binnen de Gemeente Tilburg is een dergelijk plan, met uitzondering van de voorziening voor de uitwijk van de IT infrastructuur naar de colocation, niet beschikbaar.

De aanbeveling is om een dergelijk plan op te stellen en de daarbij passende maatregelen te nemen.

Bij on-premise componenten zal - afhankelijk van de eisen – als mitigerende maatregel gekozen worden voor:

1. Enkelvoudige Active-Passive configuratie waarbij de virtuele servers na een calamiteit op de remote locatie worden opgestart.
2. Active-Active configuratie waarbij alle verwerking standaard verdeeld wordt over meervoudig uitgevoerde componenten. De dimensionering van de componenten is zodanig dat in geval van uitval van een willekeurig component, de resterende configuratie de volledige werklast kan blijven verwerken.

Een combinatie van 1 en 2 kan worden gebruikt om zowel aan de beschikbaarheid (active-active) - en bedrijfscontinuïteit (uitwijk naar colocation) eisen te voldoen.

Ten behoeve van bedrijfscontinuïteit kan een Active-Active configuratie over meerdere geografische locaties worden ingericht, waarbij de uitwijksituatie impliciet gevalideerd wordt op correctheid en volledigheid.

8.1.2 Cloud

Voor alle cloud componenten is de desbetreffende leverancier verantwoordelijk om op passende maatregelen te nemen om aan de gestelde bedrijfscontinuïteitseisen van de Gemeente Tilburg te voldoen.

Het platform moet mogelijkheden bieden om de correcte werking ervan te kunnen bewaken. Het beschikt over monitoring functies ter bewaking van het overschrijden van ingestelde drempelwaarden, specifiek van kritische omgevingsparameters die kunnen leiden tot instabiliteit of uitval van een omgeving.

8.2 Integriteit

De componenten voor berichtverwerking zijn zodanig ontworpen en ingericht, dat foutsituaties worden voorkomen, herkend en zo mogelijk hersteld zonder afbreuk te doen aan de integriteit van de verwerking.

- In geval van API's worden de fouten terug gemeld aan de aanvrager die verantwoordelijk is voor de verdere afhandeling van de foutsituatie.
- In het geval van notificatieberichten is de leverancier verantwoordelijk voor het bewaken van niet verwerkte berichten op queues en error queues. Onder herstelacties valt ook het opnieuw aanbieden van berichten voor verwerking.

Het integratieplatform bevat mogelijkheden om berichten te kunnen valideren op correctheid. Hiervoor wordt syntactische validatie van berichten uitgevoerd op basis van de berichtspecificaties, zoals API specificaties (OAS), web service definities (WSDL) of XML schema's. Validatie dient zo dicht mogelijk aan de randen van het integratieplatform plaats te vinden. Validatie is een functie van een API gateway maar kan ook in de API integratiecomponent plaatsvinden.

Berichten die niet succesvol kunnen worden gevalideerd, worden niet verder verwerkt maar via daartoe ingerichte foutverwerkingslogica afgehandeld.

8.3 Vertrouwelijkheid

Het integratieplatform moet op de juiste wijze met vertrouwelijke data omgaan. Het dient er ook voor te zorgen dat geen ongeautoriseerde toegang tot het platform en datastromen mogelijk is.

Er wordt geen data opgeslagen in het platform, anders dan berichten die tijdelijk bewaard worden op een queue en log bestanden. Logbestanden bevatten nooit vertrouwelijke data.

8.3.1 Toegang

Door middel van de authenticatie functie stelt het integratieplatform de identiteit van de gebruiker of ander subject vast dat toegang wil hebben tot het platform. Hierbij wordt onderscheid gemaakt tussen gebruikers van de API's en integraties en beheerders van het platform.

Eindgebruikers hebben nooit direct toegang tot het integratieplatform in de vorm van aanloggen e.d..

Autorisaties zijn ingesteld op basis van welke rechten in welke gebruikersgroepen zijn ondergebracht. Alle gebruikers die toegang hebben tot het platform, of gebruik maken van integraties en/of API's op het platform zijn uniek herleidbaar tot één natuurlijke persoon, organisatie of IT-voorziening.

Toegang tot het platform mag voor personen enkel op basis van gedefinieerde rollen plaatsvinden en door toegangsbeveiligingsmiddelen te worden afgedwongen. Om de mogelijkheden van misbruik te beperken, hebben beheerders van het platform niet méér rechten dan zij voor hun werk nodig hebben (geen standaard administrator of root rechten). Verleende toegangsrechten zijn inzichtelijk en beheersbaar in een daarvoor ingerichte omgeving.

Logbestanden moeten beschikbaar zijn om te kunnen worden geanalyseerd door de leverancier of de Gemeente Tilburg om afwijkingen in de werking van het platform en daarop uitgerolde componenten vast te stellen en beveiligingsincidenten te detecteren.

8.3.1.1 API toegang

Aanroepen van API's op het platform dienen enkel voor geauthentiseerde en geautoriseerde aanvragers te zijn toegestaan. Alle API aanvragen worden gelogd. Over het volledige API verkeer kunnen analyses worden uitgevoerd waarvan grafische weergaven in een dashboard kunnen worden weergegeven.

8.3.1.2 OAUTH 2.0

De standaard OAuth (Open Authorization -actuele versie 2.0) is een open standaard voor autorisatie. OAuth 2.0 maakt gebruik van tokens, in plaats van de vertrouwelijke gegevens als een gebruikersnaam of wachtwoord. Met OAuth 2.0 kan een API aanvrager toegang krijgen tot specifieke gegevens, die opgeslagen zijn in een bronsysteem, zonder de gebruikersnaam en wachtwoord in de aanroep mee te geven. OAuth 2.0 is een autorisatiestandaard voor met name applicaties die gegevens uitwisselen met behulp van API's. Elk token geeft slechts toegang tot specifieke gegevens voor een bepaalde duur. Zo kan ingesteld worden dat een bepaald programma slechts een uur toegang heeft tot de gegevens. Hierna moet opnieuw een token worden gevraagd.

Het gebruik van OAuth 2.0 is niet verplicht maar wordt geadviseerd voor applicaties waarbij eindgebruikers toestemming geven (impliciet of expliciet) aan bijvoorbeeld een app of web applicatie om namens hem/haar via een RESTful API toegang te krijgen tot specifieke data (resource).

Voor gebruik van OAuth 2.0 is een authenticatie server vereist, die de authenticatie van de API aanvrager verzorgt. Het beheren van de identiteiten en uitgeven van de tokens heeft een sterke raakvlak met het domein Identity en Access Management (IAM). Een IAM oplossing is randvoorwaardelijk voor het gebruik van OAUTH 2.0.

8.3.2 Berichtenverkeer

In lijn met de Technische standaarden informatiebeveiliging van de NORA wordt al het berichtenverkeer tussen systemen op beveiligde wijze ingericht. Dit betekent dat alle API's en web services op basis van HTTPS worden aangeroepen, zowel intern als extern. Hierbij kan onderscheid gemaakt worden tussen intern en extern verkeer wat betreft het certificaat dat gebruikt wordt bij de verbinding.

8.3.3 Zonering

De technische infrastructuur is in zones ingedeeld om isolatie van onderdelen hiervan mogelijk te maken. Alle componenten van productieomgeving van het integratieplatform zijn afgescheiden van de ontwikkel, test en acceptatieomgevingen.

8.3.4 Cloud Computing

Daar waar de Gemeente Tilburg gebruik maakt van de clouddiensten van externe leveranciers, dienen deze aan de vertrouwelijkheidscriteria te voldoen zoals vastgelegd in het document 'Cloud Computing Eisen en Wensen InformatieBeveiliging (v1.59)'.

Cloud computing is hierin gedefinieerd als 'de situatie waarin toepassingen ter beschikking worden gesteld door gebruik van internettechnologie.' Cloud Computing raakt alle maatregelen die de Gemeente Tilburg zelf moeten nemen in de eigen infrastructuur. Deze maatregelen worden vervolgens gedeeld door de Cloud-leverancier en door de Gemeente Tilburg uitgevoerd. Indien de Gemeente Tilburg gebruikt maakt van Cloud-diensten dan blijft de gemeente Tilburg verantwoordelijk voor de juiste beveiliging van haar gegevens en waarborging van de privacy.

Voor verdere details van de eisen en inhoud wordt verwezen naar het document 'Cloud Computing Eisen en Wensen InformatieBeveiliging'

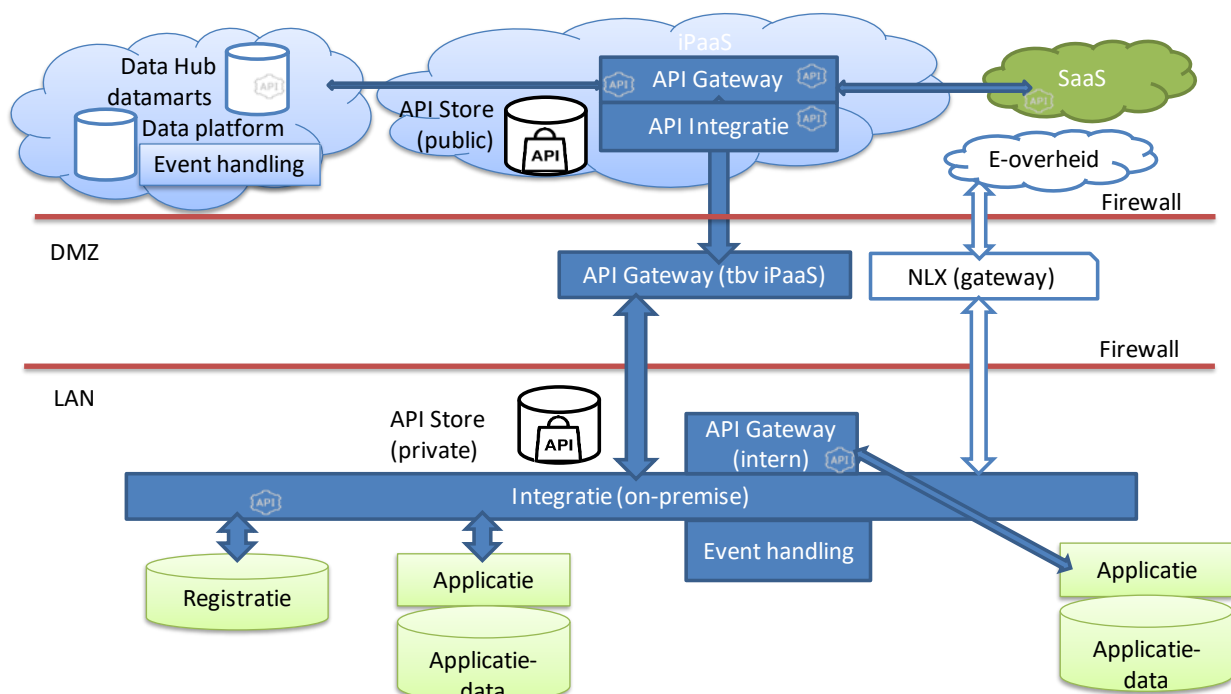
9 Infrastructuur

Het integratieplatform is voorzien op een hybride infrastructuur, met componenten in de cloud en on-premise. Het cloud component van het integratieplatform wordt *integration Platform as a Service (iPaaS)* genoemd. De totale oplossing wordt in de markt betiteld als een *Hybrid Integration Platform (HIP)*.

Voor de koppeling van de iPaaS met de on-premise componenten wordt gebruik gemaakt van een beveiligde verbinding via een gateway in de DMZ.

Zowel de iPaaS als de on-premise componenten bestaan uit een API integratiecomponent voor de functionele integratie en gateway ten behoeve van de policy enforcement. Uitgangspunt is dat iedere API een beveiligde API is, met andere woorden altijd via een gateway wordt aangeroepen.

Voor de volledigheid is de NLX van Common Ground ook in de plaat opgenomen, die de verbinding legt met andere Gemeenten, ketenpartners en overheden. De NLX Gateway kan ook in de cloud worden ingericht.



Figuur 8: Technische bouwblokken doelarchitectuur

De kerncomponenten van het integratieplatform, in donkerblauw weergegeven, zijn kritisch in de zin dat ze een *single point of failure* kunnen vormen. Voor het realiseren van integraties die kritisch op het aspect beschikbaarheid is het van belang deze componenten hoog beschikbaar (active-active) uit te voeren.

Zo lang er geen strikte SLA's of beschikbaarheidseisen van toepassing zijn op het integratieplatform, wordt om reden van kostenbeheersing in de on-premise omgeving van de Gemeente Tilburg volstaan met enkelvoudig uitgevoerde componenten.

9.1 On premise

De interne IT infrastructuur van de Gemeente Tilburg is ondergebracht in 2 datacentra:

- Gemeente Tilburg DC SK2
- Tilburg University

SK2 is het primaire DC en in geval van calamiteiten kan worden uitgeweken naar de faciliteit op het universiteitsterrein. Hierbij worden alle servers in de uitwijkconfiguratie opgestart. Dit betekent dat alle servers en services primair in het DC in SK2 beschikbaar zijn.

9.1.1 Server platform

Het serverplatform wordt aangeboden middels virtualisatie:

- Windows servers op VMware vSphere platform

Besturingssysteem voor Windows systemen:

- Microsoft Windows Server 2016

Active Directory:

- MS Windows Server 2016

9.1.2 Netwerk

Het netwerk van de Gemeente Tilburg is op basis van IPV4, in de toekomst zal dit op basis van IPV6 zijn. Gebruik van TLS 1.2 is de standaard voor encryptie, waarbij geen gebruik mag worden gemaakt van self signed certificaten

9.1.2.1 LAN

De LAN omgeving van de Gemeente Tilburg bestaat uit meerdere segmenten. De meeste backend servers zitten in één enkel segment.

De roadmap van het netwerk is als volgt:

1. Verbeteren netwerk segmentering
2. Microsegmentatie toepassen op basis van functionaliteiten.
3. Doorgroeien naar een *Zero Trust Model* met *Identity Access Management (IAM)* en *Privileged Access Management (PAM)* toevoegingen

Microsegmentatie verdeelt het datacenter in kleinere, beter beschermde zones zodat de impact van een gecompromitteerd endpoint beperkt blijft. Kenmerk van een integratieplatform is dat het koppervlakken heeft met veel andere systemen binnen het IT netwerk en bij toepassing van microsegmentatie de autorisaties hiervoor geconfigureerd moeten worden.

9.1.2.2 DMZ

Het DMZ heeft als doel een veiligheidszone te bieden voor het interne IT landschap die externe koppelingen hebben. De API gateway wordt in de DMZ geïmplementeerd voor de beveiliging van het API verkeer van buiten naar binnen.

9.1.2.3 Firewall

Firewalls worden gebruikt voor de beveiliging van de DMZ en LAN voor ongeautoriseerde toegangspogingen naar de beveiligde omgeving.

Per endpoint integratie die door een firewall loopt zal de bijbehorende poort hiervoor open moeten staan.

9.1.2.4 Loadbalancing

Zodra de beschikbaarheidseisen voor de on-premise componenten van het integratieplatform vragen om een hoog-beschikbaarheid configuratie, is een extra netwerkcomponent vereist om het API verkeer goed te kunnen verdelen.

Voor het HTTPS transport is een loadbalancer onderdeel van de configuratie om de meervoudige inrichting te kunnen gebruiken in een hoog-beschikbaarheid configuratie.

N.B.: Bij message queue (JMS) afnemers wordt de loadbalancing door de message broker zelf uitgevoerd.

9.2 Gemeentelijke Gemeenschappelijke Infrastructuur (GGI)

Gemeenten hebben VNG en VNG Realisatie in het kader van Samen Organiseren opdracht gegeven te werken aan meer collectiviteit op ICT-voorzieningen. De GGI is een ICT- infrastructuur van en voor gemeenten en gemeentelijke samenwerkingsverbanden.

Het realiseren van een GGI schept de voorwaarden om op een goede, veilige en flexibele wijze de overstap te maken naar de cloud. De GGI creëert een veilige, samenhangende digitale infrastructuur waardoor samenwerken tussen gemeenten en andere overheden beter, veiliger en gemakkelijker wordt. De GGI vormt de technische basis waarin toegewerkt wordt naar een gemeentelijke community cloud, die is aangesloten op de landelijke Generieke Digitale Infrastructuur (GDI).

Binnen de GGI ligt een relatie met de ontwikkeling van Common Ground.

De volgende GGI-projecten zijn reeds gestart:

- GGI-Netwerk: Een beveiligd landelijk datanetwerk van en voor gemeenten.
- GGI-Afspraken: Dit zijn de voorwaarden die gelden voor het ontsluiten van commerciële cloud-omgevingen op het GGI-Netwerk.
- GGI-Veilig: zorgt voor de actieve beveiliging van het dataverkeer over het GGI-Netwerk en over de eigen bedrijfsnetwerken van gemeenten.

Als gevolg van de marktontwikkeling zal de Gemeente Tilburg steeds meer gebruik gaan maken van Cloud diensten en SaaS oplossingen. Ook het integratieplatform is voor een deel voorzien in de cloud om deze ontwikkeling optimaal te ondersteunen. Het zal op grond daarvan moeten kunnen aansluiten op de GGI en voldoen aan de daarvoor geldende afspraken .

10 Organisatie

Met de introductie van het integratieplatform binnen de Gemeente Tilburg, ontstaan er nieuwe rollen in de organisatie en krijgen bestaande rollen er taken, bevoegdheden en verantwoordelijkheden bij. In paragraaf 6.2 API op bladzijde 17 wordt al een aantal rollen genoemd. In dit hoofdstuk worden deze rollen nader toegelicht.

10.1 Ontwerp

In de context van het ontwerpen van nieuwe integraties en API's worden de volgende rollen onderkend:

- API-eigenaar
- Solution architect / integratie ontwerper

Deze rollen worden hieronder toegelicht.

10.1.1 API eigenaar

Een nieuwe rol voor de Gemeente Tilburg in de context van API integratie is de API-eigenaar. Net als bij de applicatiebeheerders en data-eigenaren zal dat er meer dan één zijn. De API eigenaar is verantwoordelijk voor het ontwerp, documentatie en voorbeelddata van een API alsmede de hele levenscyclus nadat de API in productie is genomen. De API eigenaar draagt zorg voor de API's wat betreft vindbaarheid, bruikbaarheid, documentatie, kwaliteit, etc.. Gedurende het gebruik van de API zal de API eigenaar terugkoppeling ophalen bij de stakeholders.

Per API per afnemer zal met de API eigenaar een afspraak moeten worden gemaakt over het gebruik van de API, zoals het gebruik per tijdeenheid voor de instelling van de throttling functie op de API Gateway.

10.1.2 Solution architect / ontwerper

Voor nieuwe integraties is de solution architect / ontwerper verantwoordelijk is voor het opstellen van de procesbeschrijving, identificatie van de onderliggende dataobjecten en daarmee betrokken applicaties, systemen en databronnen.

De integratierichtlijnen en principes dragen bij aan een flexibele, onderhoudbare en kostenefficiënte oplossing. In de ontwerprol zit ook de afstemming met leveranciers van aan te sluiten applicaties, SaaS diensten en/of databronnen.

De solution architect / ontwerper levert een PSA en functioneel detailontwerp op grond waarvan een ontwikkelaar een technisch ontwerp en implementatie kan opleveren.

10.2 Beheer

De implementatie van het integratieplatform heeft gevolgen voor de bestaande beheerrollen binnen de Gemeente Tilburg, te weten:

- Applicatiebeheer
- Technisch beheer

Per rol wordt in de volgende paragrafen de hieraan gekoppelde taken beschreven.

10.2.1 Applicatiebeheer

Applicatiebeheer heeft inzicht in de afhankelijkheden van integratie en API's met applicaties en SaaS diensten binnen de Gemeente Tilburg om in geval van verstoringen op het integratieplatform de impact ervan vast te kunnen stellen en de juiste applicatiebeheerder hierover te informeren.

Het operationeel beheer van het Integratieplatform en de daarop geïmplementeerde integraties en API's is belegd bij de leverancier. Dit is onderdeel van de afgenomen dienst (*Connection as a Service*).

De leverancier zal geconstateerde verstoringen in het integratieplatform doorgeven aan de betreffende applicatiebeheerders van de Gemeente Tilburg. Applicatiebeheer zal op grond van de aard van de verstoring verdere acties ondernemen, zoals het loggen en managen van het incident en communicatie richting de beheerders en gebruikers van applicaties die direct impact ondervinden als gevolg van de verstoring.

Functionele fouten in integraties kunnen doorgaans niet zonder meer door de leverancier van het integratieplatform worden opgelost omdat ze directe raakvlakken hebben met aangesloten applicaties. Applicatiebeheer van de Gemeente Tilburg is verantwoordelijk voor het beleggen van functionele fouten die optreden op het integratieplatform bij de applicatiebeheerder van de applicatie die de mogelijke veroorzaker is van de functionele fout.

Verder zijn er aanvullende taken voor de applicatiebeheerders van het integratieplatform. Deze worden in de volgende paragrafen toegelicht.

10.2.1.1 Toegangsbeheer

Het beheren van toegang tot het gebruik van API's is een verantwoordelijkheid van applicatiebeheer. Aanvragen voor het gebruik van API's kunnen via de API portal worden ingediend in het geval een dergelijke functie is ingericht. In ander gevallen komen aanvragen via andere kanalen bij de applicatiebeheerder terecht. Ongeoorloofd gebruik van API's kan leiden tot datalekken. Applicatiebeheer moet er op toezien dat autorisaties ook worden ingetrokken zodra de afnemer geen recht op gebruik van de API meer heeft, zoals in het geval dat een medewerker uit dienst treedt.

10.2.1.2 API monitoring

De API Gateway biedt de mogelijkheid het gebruik van API's te monitoren. Alerts kunnen worden ingesteld op foutsituaties, zoals timeouts of een vastgesteld aantal foutmeldingen op een API binnen een bepaald tijdbestek. Hiermee worden de beschikbaarheid en werking van de gemanagede API's en onderliggende integraties bewaakt.

Om beter en sneller inzicht te krijgen in de status van API's moeten applicatiebeheerders van de Gemeente Tilburg toegang hebben tot de API monitoring dashboard.

10.2.1.3 Messaging/streaming

In het geval er in het integratieplatform gebruik wordt gemaakt van messaging of streaming componenten, waarbij onder andere sprake is van het opslaan of persisteren van berichten, zal aanvullend beheer ingeregeld moeten worden voor het bewaken van de gebruikte resources.

Resources kunnen zijn: queues, durable subscribers en message stores.

In geval van aanhoudende technische fouten of functionele fouten waarbij berichten op een error queue of dead letter queue (DLQ) terecht komen, hierop actie worden ondernemen. Dit kan zijn het opnieuw laten aanbieden van berichten na herstel van de foutsituatie, in nauw overleg met de applicatiebeheerders van de betrokken applicaties.

Als het vollopen van een messaging resource onopgemerkt blijft dat dit, afhankelijk van de configuratie en implementatie, leiden tot berichtverlies, niet meer verwerken van berichten of het betreffende component doen falen.

10.2.2 Technisch Beheer

Alle infrastructuur, security en netwerkvoorzieningen die nodig zijn voor de correcte werking van de on-premise componenten van het integratieplatform vallen onder de verantwoordelijkheid van Technisch Beheer (TB). Tot de beheertaken horen ook het bewaken van de aanwezigheid van en zo nodig toevoegen van voldoende resources, zoals CPU, geheugen of disk opslagruimte. Ook dient de geldigheid van certificaten gecontroleerd te worden en deze vervangen als ze dreigen te verlopen.

Voor het technisch beheer van het platform is geen inhoudelijke kennis van het integratieplatform noodzakelijk, anders dan inzicht in de onderlinge verbindingen en security maatregelen.

10.3 Leveranciersmanagement

In de context dat het integratieplatform als dienst wordt afgenomen van een leverancier, zijn de volgende rollen voorzien voor de afstemming op de verschillende vlakken:

- Contractmanager
- Portfoliomanager
- Integratiearchitect

10.3.1 Contractmanager

De contractmanager is verantwoordelijk voor het beheer van het contract met de leverancier en bewaken van de gemaakte afspraken en kosten.

10.3.2 Portfoliomanager

De portfoliomanager is belast met het bijhouden van de lopende projecten en activiteiten die nieuwe componenten op het integratieplatform gerealiseerd willen hebben. De Portfoliomanager is degene binnen de organisatie die alle aanvragen voor werkzaamheden op het Integratieplatform bundelt en fungeert als *single point of contact* voor werkopdrachten, planning en prioritetstelling richting de leverancier van het integratieplatform.

De portfoliomanager bepaalt – in overleg met de stakeholders binnen de Gemeente Tilburg – de onderlinge prioriteit en moment van oplevering van (deel)componenten. De portfoliomanager bewaakt hierbij de doorlooptijd en bouwkosten ten opzichte van de afgegeven inschatting.

10.3.3 Integratiearchitect

De integratiearchitect bewaakt de architectuorkaders van het integratieplatform bij de realisatie van nieuwe integratiecomponenten. Hieronder vallen de niet-functionele aspecten als robuustheid, herbruikbaarheid en onderhoudbaarheid maar ook kwaliteitscontrole van de gerealiseerde componenten aan de hand van de principes en richtlijnen. Code review kan onderdeel zijn van deze kwaliteitscontrole.

De integratiearchitect voert hierover regelmatig overleg met de leverancier van het integratieplatform. Waar nodig worden de bestaande architectuorkaders uitgebreid of aangepast.

11 Kennis en vaardigheden

Naast het beheersen en kunnen toepassen van de integratieprincipes, zijn de volgende kennisgebieden relevant voor de architecten binnen de Gemeente Tilburg in het context van het integratieplatform.

11.1 Solution architectuur

De Gemeente Tilburg werkt onder architectuur. Onderdeel van projecten kan zijn het opstellen van een Project Start Architectuur (PSA) door een Solution Architect. De PSA is het projectdocument waarin de beoogde oplossing beschreven wordt vanuit de business context, waaronder business processen, data en applicaties.

Archimate is de taal die wordt gebruikt om de samenhang van de componenten in het landschap te beschrijven, gekoppeld aan de business laag en infrastructuur. Deze beschrijving wordt opgesteld in de EA tool BlueDolphin. Kennis van de Archimate taal en van BlueDolphin zijn vereist.

11.2 Datamodellering

De PSA bevat ook een beschrijving van de bij de oplossing betrokken dataobjecten. Dataobjecten die onderdeel zijn van een integratieontwerp vormen de basis van de API specificaties. Meerdere dataobjecten samen vormen een datamodel. Een beschrijving van een datamodel kan op basis van Archimate, een Entity Relation Diagram (ERD) of UML Class Diagram. Dit zijn internationale standaarden met onderscheid op kenmerken en niveau van complexiteit.

De UML Class Diagram notatie wordt het meest gebruikt in de context van applicatie integratie en API's. Het advies is deze notatiestandaard ook binnen de Gemeente Tilburg te gebruiken voor het ontwerpen van datamodellen voor integraties en API's.

11.3 Procesbeschrijving

Solution architecten en/of ontwerpers zijn binnen de Gemeente Tilburg belast met het opstellen van de ontwerpen van de te realiseren integratieprocessen op basis van de use cases en daaraan gekoppelde functionele en niet-functionele eisen.

Voor de uitwerking van de ontwerpen van de integraties en integratie flows, is een beschrijving van de interactie tussen de systemen nodig. Hiervoor lenen zich de UML Activity Diagrams en UML Sequence diagrams:

- UML Activity Diagrams beschrijven op basis van Swimlanes de functionele interactie tussen de betrokken componenten, waaronder de applicaties en integratieplatform.
- UML Sequence Diagrams beschrijven de technische interactie tussen de componenten in een integratieketen, waarbij berichten worden benoemd en onderscheid wordt gemaakt tussen synchrone en asynchrone interacties.

11.4 API Specificaties

Bij het werken op basis van een API-first aanpak, moeten eerst alle interfaces zijn beschreven, voordat er kan worden gebouwd. In geval van nieuw te ontwerpen RESTful API's worden de API specificaties in OAS 3.0 beschreven. In het geval van Open API specificaties van Common Ground zijn deze beschikbaar en klaar voor gebruik. Voor applicatie-specifieke API's kan de leverancier gevraagd worden de specificaties hiervan aan te leveren in OAS formaat.

Hoewel het opstellen van een API specificatie niet een taak is die bij de Gemeente Tilburg ligt, maar gebruik zal worden gemaakt van standaard API's van Common Ground en door leveranciers van applicaties aangeleverde API's, is het advies minimaal in staat te zijn een API specificatie op basis van OAS te kunnen lezen en begrijpen.

11.5 Code review

In het geval dat code review onderdeel is van deze kwaliteitscontrole van de opgeleverde componenten, betekent dit dat er binnen de Gemeente Tilburg voldoende platformkennis aanwezig moet zijn om dit uit te kunnen voeren. De benodigde vaardigheid is code te kunnen lezen, niet zelf produceren.

11.6 Applicatiebeheer

Functioneel beheer van het integratieplatform valt onder Applicatiebeheer (AB). Kennis van de gebruikte beheertools, zoals de API monitoring dashboard, is hierbij noodzakelijk.

12 Roadmap

12.1 Platform- en leverancierskeuze

De Gemeente Tilburg heeft als pilot gekozen voor Dell Boomi en Broadcom Layer7 als technische bouwblokken van het Integratieplatform. Deze producten zijn marktleiders in hun segment, zoals gedefinieerd door de onderzoeksbureaus Forrester en Gartner.

De producten ondersteunen het gebruik van Open Standaarden en er zijn meerdere lokale marktpartijen die diensten leveren op basis van deze producten. De belangrijkste motivatie hiervoor is de continuïteit zo veel mogelijk te waarborgen en het risico op vendor-lockin te beperken.

Enable-U biedt het integratieplatform als een dienst aan, onder de naam Connection as a Service (CaaS). Enable-U maakt ten behoeve van hun clouddiensten gebruik van de cloud infrastructuur in de Leaseweb datacenters in Amsterdam. Hierin worden de API gateway van Layer7 en runtimes van Dell Boomi gehost.

De Boomi development en management omgeving is onderdeel van de multi-tenant cloud oplossing van Dell Boomi. In deze omgeving wordt geen data van de Gemeente Tilburg verwerkt of opgeslagen. Voor de antwoorden op de 'Cloud Computing Eisen en Wensen InformatieBeveiliging' wordt verwezen naar het betreffende door Enable-U beantwoorde exemplaar. De Gemeente Tilburg is en blijft eigenaar van de op dit platform geïmplementeerde integraties en API's (broncode).

De als onderdeel van de overeenkomst opgenomen exit-strategie schept de kaders om over te kunnen stappen naar een andere dienstverlener, op basis van het Dell Boomi / Broadcom Layer7 platform. Dit beperkt de vendor-lockin van Enable-U als dienstverlener.

De CaaS dienst die Enable-U op hun integratieplatform levert, waarbij de leverancier verantwoordelijk is voor het hele traject van technisch ontwerp tot en met beheer van de integraties, sluit zonder aanvullende afspraken uit dat de Gemeente Tilburg en/of een andere marktpartij zelf integraties kunnen gaan realiseren en uitrollen op het Integratieplatform. Het advies is hierover aanvullende afspraken met Enable-U te maken om hierbij een vorm van marktwerking open te houden.

Binnen de initiële periode van één jaar, ingaande 1 maart 2020, zal de samenwerking met Enable-U en het integratieplatform worden geëvalueerd. Indien daar aanleiding toe bestaat kan de samenwerking binnen een jaar worden opgezegd en treedt de exit strategie in werking. De exacte inhoud staat beschreven in het contract dat is opgesteld voor de dienstverlening van Enable-U.

12.2 Vervolgtrajecten

Het succes van het integratieplatform is sterk afhankelijk van het succes van de vervolgtrajecten die er gebruik van gaan maken. Selectie van de juiste kandidaten is randvoorwaardelijk.

Vervolgtrajecten kunnen worden ingedeeld in verschillende niveaus van volwassenheid in de context van de doelarchitectuur.

De volgorde die is gehanteerd is dat per niveau de ervaring met een oplossing afneemt en de onzekerheid over de oplossingsrichting groter is. Zo zijn de API's van Common Ground nog volop in beweging, maar zijn de StUF koppelingen vrijwel uitgekristalliseerd.

Per niveau horen ook andere mate van kennis en uitvoering van regiefunctie op de realisatie van deze trajecten.

De verschillende niveaus worden in de volgende paragrafen toegelicht.



12.2.1 Onder controle brengen van koppelingen

Nieuwe koppelingen worden bij voorkeur op basis van API's als dit standaard API's zijn of met geringe inspanning te realiseren. Maar als deze nog niet beschikbaar zijn, kunnen koppelingen op basis van de StUF standaard via het integratieplatform worden gelegd, waarbij beveiliging, monitoring en beheer standaard geregeld zijn. Bestaande maatwerk koppelingen kunnen naar het nieuwe platform worden gemigreerd als dit de kwaliteit verbetert, beheerlast beperkt en/of kosten (licenties) bespaart.

Consolidatie van bestaande integratieoplossingen levert na realisatie een besparing op licentiekosten, beheerlast en infrastructuur. Hiervoor kunnen gerichte business cases worden opgesteld en gerealiseerd.

Kandidaten voor consolidatie binnen Gemeente Tilburg zijn:

- Oracle Service Bus (OSB)
- Koppelvlakken Sociaal Domein (Centric Message Handlers)
- OpenTunnel van JNet ten behoeve van Exxellence
- Makelaarssuite

De leverancier Enable U levert een aantal standaard koppelingen waarmee eenvoudig bestaande koppelingen vervangen kunnen worden. De huidige lijst van koppelingen is te vinden op <https://www.enable-u.nl/connections-as-a-service/>

Daarbij kunnen voor de Gemeente Tilburg additionele integraties worden gebouwd waarmee elke van de bestaande integraties kunnen worden vervangen. De hiermee gemoeide kosten zijn afhankelijk van hoeveel ontwerpinspanning de Gemeente Tilburg zelf levert of dit over laat aan de leverancier(s), het aantal betrokken systemen, aantal interfaces en complexiteit van de transformaties.

12.2.2 Aansluiten van SaaS diensten

Nieuwe SaaS diensten en applicaties die door de leverancier als SaaS oplossing naar de Cloud gemigreerd worden, moeten op beveiligde en gecontroleerde wijze ontsloten worden. Concrete voorbeelden hiervan zijn Squit20/20 van Roxit en Beaufort van Visma|Raet.

Kenmerk van SaaS applicaties is dat de koppelvlakken enkel via API's worden aangeboden. Alleen al voor een veilige en bewaakte integratie van de SaaS applicatie met de rest van het IT landschap van de Gemeente Tilburg is de API Gateway in de Cloud het aangewezen koppelvlak. Deze manier van integreren bespaart bijvoorbeeld op het aanleggen en onderhouden van nieuwe VPN's en kunnen data-lekken worden voorkomen. Waar mogelijk worden herbruikbare componenten in de vorm van RESTful API's opgeleverd.

Aandachtspunt bij Cloud migraties is de koppeling met de datawarehouse en datalake. Een rechtstreekse koppeling op de database van de applicatie is in dat geval niet meer mogelijk. Soms biedt de leverancier van de applicatie maatwerk aan door regelmatig een bestand aan te maken die kan worden gedownload. Afhankelijk van de kosten en omvang van de data kan voor een dergelijke oplossing worden gekozen.

Is dit alternatief er niet, zal gebruik gemaakt moeten worden van de aangeboden API's. Indien mogelijk maakt de SaaS oplossing gebruik van het event notificatie patroon om gebeurtenissen over dataobjecten te publiceren, waarna het dataobject met een API call kan worden opgehaald. Het integratieplatform kan deze functie vervullen.

In alle andere gevallen zal door het BI platform door middel van API calls de relevante data opgehaald moeten worden, waarbij rekening moet worden gehouden met omvang van de data en tijdstip van ophalen. Dit om het online gebruik niet in de weg te zitten met verminderde performance.

De keuze voor de oplossingsrichting is onderdeel van de startarchitectuur of integratieontwerp.

12.2.3 Voorsorteren op Common Ground

De 5-lagen architectuur van Common Ground is gebaseerd op een RESTful API landschap. Bij nieuwe aansluitingen op applicaties en databronnen worden API's als herbruikbare gegevensdiensten (laag-2 van Common Ground) gebouwd in lijn met de doelarchitectuur. Complexere functies kunnen op basis van API's gemaakt worden in de procesinrichtingslaag (laag 4). Hiermee worden bouwstenen voor hergebruik en flexibilisering van het IT landschap gerealiseerd.

Zodra de NLX als productiecomponent in het IT landschap van de Gemeente Tilburg beschikbaar is, kunnen API's van de gegevensdiensten die voldoen aan de landelijke afspraken hierop voor andere Gemeenten en ketenpartners worden ontsloten.

Vanuit de regie rol zal de Gemeente Tilburg ook moeten nazien op oplevering van herbruikbare componenten door de leverancier van het integratieplatform. Dit is één van de acceptatiecriteria van integratiecomponenten.

13 Bijlage Architectuur richtlijnen en principes

De kwaliteit van een integratieoplossing wordt bepaald door de inrichting van de koppelingen en daaruit volgend de beheersbaarheid en onderhoudbaarheid. Toepassing van richtlijnen en principes dragen bij aan het realiseren van deze kwaliteitsdoelstellingen.

13.1 Integratierichtlijnen

De integratierichtlijnen sturen de keuzes die moeten gemaakt bij het ontwerpen van een integratieoplossing en moeten altijd worden toegepast bij het aansluiten van nieuwe applicaties of diensten en het ontwerpen van nieuwe integraties.

13.1.1 Standaardmanier van ontsluiting en koppeling

In een onderhoudbaar applicatielandschap geldt als uitgangspunt dat alle applicaties en bronnen op een standaardmanier ontsloten en gekoppeld worden.

13.1.1.1 Koppelingstype

Het Koppelingstype beschrijft hoe twee of meer systemen zijn verbonden om een integratie te maken. Een Koppelingstype kan er een zijn van:

- Integratieplatform inclusief API Management en API-integratie/ESB functies
- Extract Transform en Load (ETL) processen
- Rechtstreekse integraties
- Handmatige integraties

De uitleg over het gebruik van bovenstaande koppelingstypen is in de paragrafen hieronder opgenomen.

13.1.1.2 Rationale

- Realiseren van een uniforme manier van koppelen van applicaties, diensten en bronnen in het IT-landschap van de Gemeente Tilburg.
- Optimale procesondersteuning en informatie-uitwisseling tussen toepassingen
- Reduceren van de beheerlast en onderhoudskosten van het applicatielandschap door maatwerk terug te dringen en het aantal verschillende koppelingstypen beperkt te houden..

13.1.1.3 Implicaties

Bij het selecteren van een koppelingstype voor nieuwe integraties met een bestaande applicatie moet het koppelingstype van de bestaande integraties van die applicatie worden heroverwogen. Als een bestaand koppelingstype geen deel uitmaakt van de integratiearchitectuur of als een ander koppelingstype de voorkeur heeft dan het huidige type, moet een migratie in gang worden gezet om de bestaande integraties te migreren naar het gewenste koppelingstype.

Andere koppeling-typen dan beschreven in de integratiearchitectuur mogen geen deel uitmaken van het IT landschap van de Gemeente Tilburg en worden beschouwd als *technical debt*.

13.1.1.4 Integratie en API Management platform

Het integratieplatform platform bevat Enterprise Service Bus (ESB) en API functionaliteiten. Dit platform ondersteunt een API-architectuur (Request-Reply) en Event Driven Architecture (Publish-Subscribe). Het gebruik van het API-integratieplatform voor koppelingen is de standaardfunctie bij het ontwerpen van integraties.

Het gebruik van het API-integratieplatform is van toepassing wanneer één of meer van volgende criteria van toepassing zijn:

- Het proces moet gegevens in bron- en doelsystemen kunnen ophalen, toevoegen of wijzigen.
- Er is potentieel hergebruik van (een deel van) de integratie; er zijn meerdere consumenten van de integratie of het kan deel uitmaken van meerdere integratie scenario's.
- Voor de afnemer wordt in de integratie de implementatie van de brontoepassing(en)/systeem(en) geabstraheerd.
- De integratie vereist datatransformatie(s) en/of orkestratie van meerdere API's.
- De integratie heeft near real-time eisen en is kortlopend.
- Het bronsysteem, applicatie of SaaS stelt interfaces beschikbaar, waaronder REST, SOAP, JDBC, enz.
- Aan alle vereisten op het gebied van beveiliging, privacy en (API en service) governance wordt voldaan door de standaard voorzieningen van het API-integratieplatform.
- De integratie vereist geautomatiseerde foutafhandeling, logging, monitoring en/of auditing voorzieningen.

Bij keuze voor het integratieplatform moet expliciet aan al deze eisen worden voldaan:

- Het proces wordt gestart door een API/service -gebruiker of door de zender van een eventbericht.
- Het proces op het integratieplatform kent geen toestand en slaat geen status of gegevens op.
- Voor een specifiek bericht of event is er geen correlatie of interactie met eerdere en toekomstige gebeurtenissen.
- De integratie bevat geen gebruikersinteractie.
- De integratie heeft geen betrekking op bulk import en export processen zoals ETL.

13.1.1.5 Extract, Transfer & Load (ETL)

ETL is de passende koppelmethode in scenario's waarbij alle volgende criteria van toepassing zijn:

- De integratie heeft slechts één afnemer. De afnemer heeft het kenmerk van een Data Warehouse (DHW) of Data Store.
- Iedere integratie is rechtstreeks gekoppeld.
- De integratie kent geen near realtime eisen en wordt uitgevoerd in een batch-type verwerking.
- Elke integratie wordt gestart door de afnemer.
- De integratie heeft betrekking op bulk data export. Er wordt geen data gewijzigd of toegevoegd in het bronsysteem.
- De integratie kent enkel standaard en relatief eenvoudige transformaties.
- De datakwaliteit, beveiliging, metadata management en governance aspecten zijn de verantwoordelijkheid van en worden afgedekt door de afnemer.
- Er is geen behoefte aan aanvullende logging en traceerbaarheid voorzieningen.

13.1.1.6 Rechtstreekse koppelingen

Rechtstreekse koppelingen zijn toegestaan als alle volgende condities gelden:

- Er is slechts één afnemer van de uitgewisselde data en mogelijke andere afnemers worden binnen 5 jaar niet verwacht.

- Het bronsysteem start het proces voor de data-aanlevering (export).
- Het uitwisselingsformaat is in de vorm van bestanden, zoals een CSV, XML of Excel export file.
- Data hoeft niet aangepast of getransformeerd te worden als onderdeel van de handmatige acties, transformatie vindt plaats in het bron- of doelsysteem.
- De integratie gebruikt een beveiligde bestandsuitwisseling oplossing (SFTP/FTPS) of een standaard integratiemethode van het bron- of doelsysteem die door de betreffende leverancier wordt ondersteund.
- Er worden slechts minimale aanpassingen op de integratie verwacht.
- De integratie heeft geen standaard foutafhandeling, logging of monitoring voorzieningen nodig.

13.1.1.7 Handmatige koppelingen

Handmatige koppelingen zijn een geldige keuze als alle onderstaande criteria van toepassing zijn:

- De integratie wordt incidenteel gebruikt.
- Het proces wordt gestart door menselijke interactie.
- De databronnen bestaan uit handmatig getriggerde bestand export, fysieke media of e-mail.
- Het uitwisselingsformaat is in de vorm van bestanden
- Data hoeft niet aangepast of getransformeerd te worden als onderdeel van de handmatige acties, transformatie vindt plaats in het bron- of doelsysteem.
- Het doelsysteem biedt een gebruikersvriendelijke upload voorziening.
- De integratie vereist geen standaard beveiligingsmaatregelen, foutafhandeling, logging of monitoring.

13.1.2 Enable-U platform voor het Integratieplatform inclusief API Management en ESB-functies

Het Enable-U integratieplatform is het enige platform binnen het IT landschap van de Gemeente Tilburg waarop integraties van het type Integratieplatform inclusief API Management en ESB-functies worden gerealiseerd.

13.1.2.1 Rationale

Integratie met behulp van het Integratieplatform inclusief API Management en ESB-functies biedt voorzieningen ter ondersteuning van het gebruik van een CDM voor berichtuitwisseling, potentiële hergebruikopties, monitoring en beheer. Het gebruik van één platform maakt het mogelijk functies en mogelijkheden zoals ontwikkeling, testen, implementatie, bewerkingen, foutafhandeling en logging te standaardiseren.

De functie van het Integratieplatform inclusief API Management en ESB-functies is het uitvoeren van:

- Transport – het afleveren van berichten op verschillende locaties (end points).
- Transformatie - protocol vertaling tussen meerdere formaten zoals HTTPS, JDBC, JMS en SFTP en transformatie van gegevens van één formaat in een ander, waaronder het CDM.
- Orkestratie - integreert twee of meer API's om gegevens te synchroniseren of API-functies te combineren in samengestelde structuren die zinvoller zijn voor de API-gebruikers. Uitvoeren van transactie management waar noodzakelijk.
- Policy enforcement – Beveiligen van API's door authenticatie/autorisatie en flow control op basis van policies, al dan niet gebaseerd op SLA's.
- Monitoring – monitoren van het gedrag van API's en herstellen van foutsituaties.
- Life cycle management – op gecontroleerde manier beheren van meerdere versies van API's.

13.1.2.2 Implicaties

Alle koppelingen van het type Integratieplatform inclusief API Management en ESB-functies zullen worden geïmplementeerd op het Enable-U integratieplatform van de Gemeente Tilburg. Er bestaan andere tools en platformen in het IT landschap van de Gemeente Tilburg die API- en applicatie integratiemogelijkheden bieden. Alleen het Enable-U platform moet hiervoor worden gebruikt om redenen van toekomstvastheid, efficiency, beheerafspraken, kostenmodel en kennisopbouw binnen de Gemeente Tilburg.

13.1.3 Toepassen API ontwerpregels

Ontwerp RESTful API's conform de richtlijnen van de API strategie voor de Nederlandse overheid, API Designrules (Nederlandse API Strategie IIa). <https://docs.geostandaarden.nl/api/API-Designrules/>

13.1.3.1 Rationale

Bij het ontwerpen van een API moeten de nodige semantische en technische keuzes worden gemaakt. De API ontwerpregels van Geonovum zijn opgesteld om bij te dragen aan de harmonisatie van afspraken rondom het ontwerpen van API's binnen de Nederlandse overheden. Dit om bij te dragen aan de kwaliteit, herkenbaarheid en bruikbaarheid van API's. Het toepassen van richtlijnen reduceert de tijd benodigd voor het maken van ontwerpkeuzes en het afwegen van alternatieven.

13.1.3.2 Implicaties

Nieuw te ontwerpen API's moeten voldoen aan de ontwerprichtlijnen van de Nederlandse API Strategie IIa. . API ontwikkelaars moeten bekend zijn met deze ontwerprichtlijnen en deze toe kunnen passen bij het ontwerpen van API's. Validatie van de API ontwerpregels is onderdeel van het formele acceptatieproces van nieuwe API's en integraties.

13.2 Integratie principes

De volgende integratieprincipes bieden handvaten voor de Gemeente Tilburg bij de ontwerpkeuzes die moeten gemaakt bij het ontwerpen van nieuwe integraties.

Principe 1. Ontkoppeling

Statement API /service aanbieders en gebruikers zijn van elkaar ontkoppeld.

Rationale Een API aanbieder en -afnemer(s) kunnen van elkaar worden ontkoppeld in:

Aspect	Methode van ontkoppeling
Tijd	Message Queueing
Volume	Message throttling / load balancing
Bestemming	Routing
Plaats	Adapter als single entry / exit point
Protocol	Transporttransformatie: HTTP -> JMS, JMS->JDBC, JMS->HTTP, etc.
Formaat	Berichttransformatie van applicatief naar CDM, JSON naar XML, etc. en vice versa.

Implicaties API /service aanbieders en gebruikers worden onafhankelijk van elkaar ontworpen en geïmplementeerd, waarbij een API service gebruiker enkel kennis heeft van de interface van de API.

Eigenschappen Efficiency, Onderhoudbaarheid

Principe 2. Implementatie- onafhankelijk

Statement Een API of service is onafhankelijk van de onderliggende implementatie

Rationale Een API of service interface definieert de toegang tot de functionaliteit die het beschikbaar stelt, waarbij het niet uit maakt op welke wijze de functionaliteit achter de API of service is geïmplementeerd zo lang aan de interface specificatie (API of service contract) wordt voldaan.

Implicaties API of service gebruikers hebben slechts kennis van het interface en geen weet van op welke wijze en in welke technologie de functionaliteit achter de API of service is geïmplementeerd.
Implementaties van service functionaliteit moeten minimaal voldoen aan de specificatie van het API of service contract.

Eigenschappen Efficiency, Onderhoudbaarheid

Principe 3. Stateless API's

Statement API's worden stateless uitgevoerd.

Rationale Een stateless API is een API die niets van de procescontext vast houdt nadat hij is aangeroepen en de aanroep heeft afgerond. De belangrijkste reden is de mogelijkheid van schaalbaarheid: er hoeft geen toestand te worden bijgehouden en gerepliceerd tussen instanties. Daarmee is herstel van een foutsituatie van een API eenvoudiger.

Implicaties Een eventuele toestand wordt buiten de API gepersisteerd, bijvoorbeeld in een database. Een volgend bericht is weer een nieuw bericht met een compleet nieuwe API aanroep.

Een API aanroep is niet afhankelijk van eerdere aanroepen. Een integratieplatform bevat geen functionaliteit voor duplicaatdetectie, berichtsortering, berichtcorrelatie en herverzending. Het integratieplatform neemt zelf niet het initiatief om berichten te verzenden, bijvoorbeeld op een bepaald moment in tijd. Er worden geen resources (geheugen, proces) vastgehouden op het moment dat de API aanroep is afgerond en waarbij eventueel een antwoordbericht verstuurd is.

Eigenschappen Onderhoudbaarheid, performance, separation of concerns

Principe 4. Geen business logica in het integratieplatform

Statement Business regels en -logica zijn gescheiden van andere programmeerlogica.

Rationale Business regels en -logica worden bepaald door interne processen en wet- en regelgeving. De Lifecycle van business regels en logica is afwijkend van lifecycles van de IT systemen. Business regels moeten tijdig en los van programmacode en infrastructurele objecten kunnen worden aangepast als de situatie daarom vraagt. Controle op correctheid en geldigheid van business regels kunnen eenvoudiger en efficiënter plaats vinden.

Implicaties Applicatieve systemen worden ontworpen met onderhoudbaarheid van de business regels en -logica als eis.
Business regels worden centraal ondergebracht (bijvoorbeeld in een Stored Procedure of Business Rules Management systeem) en uitgevraagd wanneer nodig in het proces en/of API implementatie.

Eigenschappen Onderhoudbaarheid, auditability, efficiency, separation of concerns

Principe 5. Herbruikbare componenten

Statement Oplossingen worden ontworpen door samenstelling van herbruikbare componenten. Nieuwe componenten worden ontworpen en ingezet met de mogelijkheid van hergebruik.

Rationale Herbruikbare componenten reduceren dubbele IT functies in het IT landschap. Hergebruik van functionaliteit zorgt voor minder te ontwikkelen componenten en besparing op beheerkosten. API's zijn primair ontwikkeld om hergebruik mogelijk te maken. Door te denken vanuit een API als product ontstaat zicht op gemeenschappelijke functionaliteit.

Implicaties Nieuwe applicatieve systemen worden ontworpen op basis van API principes: Applicaties stellen herbruikbare functionaliteit beschikbaar als API's, waarbij hergebruik van de API's door meerdere afnemers als uitgangspunt wordt gehanteerd. Vastlegging van het gebruik en coördinatie van aanpassingen van een API worden centraal belegd.
Applicaties gebruiken reeds beschikbare functionaliteit in andere applicaties door deze als API aan te roepen.
Er worden minder nieuwe componenten ontwikkeld, wat leidt tot lagere beheerkosten. Het ontwikkelen van herbruikbare componenten kost meer tijd dan voor één specifieke toepassing ontwikkelde componenten.

Eigenschappen Onderhoudbaarheid, efficiency

Principe 6. Uitbreidbare interfaces

Statement	Interfaces van herbruikbare componenten, API's en services worden uitbreidbaar ontworpen.
Rationale	Interfaces worden ontworpen om oudere versies te kunnen blijven ondersteunen. Vereenvoudiging invoering nieuwe versies van interfaces. Besparing op onderhoud en beheer van afnemers van herbruikbare componenten, services en API's.
Implicaties	Nieuwe versies van interfaces kunnen worden geïmplementeerd zonder dat afnemers van de huidige versie van de interface hoeven te worden aangepast. Dit bespaart op development-, test- en deploymentactiviteiten. Het ontwikkelen van uitbreidbare interfaces kost meer tijd dan niet-uitbreidbare interfaces. De compatibiliteit van een nieuwe versie van een interface ten opzichte van de bestaande versie moet met testen worden aangetoond.
Eigenschappen	Compatibiliteit, Uitbreidbaarheid, Onderhoudbaarheid

Principe 7. *forward en backward compatible API's*

Statement	Minor releases van API's zijn <i>forward</i> en <i>backward</i> compatible.
Rationale	Forward en backward compatibiliteit van een API specificatie draagt bij aan minder API major versies, eenvoudigere upgrade scenario's, minder risico en lagere kosten. Als het API contract backward compatible is ontworpen kan een minor release upgrade van dat API contract worden uitgevoerd zonder dat de API afnemers hierop hoeven te worden aangepast. Bij een forward compatible API contract hoeft het API contract niet op hetzelfde moment te worden aangepast op het moment dat een API gebruiker een minor release upgrade ondergaat.
Implicaties	Forward en backward compatibiliteit is onderdeel van de non-functionele eisen van een API specificatie en dient design-time vastgesteld te zijn. De API implementatie dient in de forward en backward compatibiliteit te voorzien in overeenstemming met de API specificatie.
Eigenschappen	Aanpasbaarheid, Flexibiliteit, Onderhoudbaarheid

Principe 8. Geen gebruikersinteractie in een API

Statement	Een API bevat geen gebruikersinteractie
Rationale	Een API kan volledig zonder menselijke tussenkomst (gebruikersinteractie) worden uitgevoerd. API's kennen een relatief korte doorlooptijd, bij voorkeur in de orde van een paar seconden.
Implicaties	In geval gebruikersinteractie in een proces noodzakelijk is voor de uitvoering ervan, wordt dit proces geclassificeerd als een workflow of Business Process Management (BPM) proces. Deze horen te worden ingericht in een BPM systeem, Zaaksysteem of een andere taakspecifieke applicatie, waarbij het concept van takenlijsten of virtuele werkbakjes is ingericht. Workflows of BPM processen kunnen als onderdeel van de uitvoering van het proces API's of services aanroepen.

Eigenschappen Efficiency, onderhoudbaarheid

Principe 9. API Eigenaarschap

Statement API's hebben exact één functionele eigenaar

Rationale API eigenaarschap legt per API vast wie er voor verantwoordelijk is en wie over wijzigingen besluit. Functioneel eigenaarschap is van belang om zinvolle en onderbouwde besluiten te kunnen nemen over het gebruik en benodigde aanpassingen van API's, deze af te stemmen met de API afnemers en daarmee de kwaliteit van de API's te kunnen garanderen. .

Implicaties API's hebben een functioneel eigenaar die verantwoordelijk is voor de API specificatie, contract, SLA en life cycle inclusief versiebeheer. De eigenaar heeft een duidelijk belang in de API en heeft budget (op teamniveau) om deze te (laten) onderhouden. Veelal is er een direct verband tussen de functioneel eigenaar van de applicatie/dataobject/resource waar de API betrekking op heeft en de eigenaar van de API. Wijzigingen in de API specificatie (interface) worden niet doorgevoerd zonder expliciete goedkeuring van de API eigenaar.

Eigenschappen Onderhoudbaarheid

Principe 10. Één waarheid

Statement Van elk dataobject is er exact één systeem de authentieke bron

Rationale Elk dataobject is minimaal één keer opgeslagen in een applicatie of database. Bij voorkeur wordt ieder object slechts éénmaal opgeslagen zodat er geen kopieën met afwijkende waarden kunnen ontstaan. In het geval er meerdere kopieën van een dataobject in het applicatielandschap bestaan, is vastgelegd welk systeem leidend is. Dit systeem representeert de waarheid en is daarmee de authentieke bron.

Implicaties Voor ieder Informatieobject is één unieke bron. Van ieder dataobject moet worden vastgelegd welk systeem de authentieke bron ervan is. Een informatieobject wordt vanuit één bron hergebruikt. Services op die dataobjecten mogen ook alleen van die authentieke bron gebruik maken om geen nieuwe realiteiten te laten ontstaan. Systemen die kopieën van dataobjecten hebben opgeslagen voor interne verwerking mogen geen services op deze objecten voor afnemers beschikbaar stellen.

Eigenschappen Efficiency, onderhoudbaarheid, single point of truth

Principe 11. Near real-time gegevensuitwisseling

Statement Gegevens worden in near real-time uitgewisseld en applicatieve componenten worden ontworpen voor (near) real-time verwerking van data en functies.

Rationale Gebruikers verwachten betrouwbare, actuele gegevens in hun werkprocessen. Gebruikerservaring vanuit websites en apps vereist (near) real-time verwerking van opdrachten en leveren van informatie. Besluiten die worden gemaakt op basis van verouderde gegevens, zoals die in batch processen worden aangeleverd, hebben een lagere betrouwbaarheid en kunnen tot fouten of inconsistenties leiden.

Implicaties	Wijzigingen in gegevens worden direct in het bronsysteem verwerkt en de informatie hierover zo snel mogelijk doorgestuurd naar andere IT systemen die een kopie van de gegevens bevatten. Applicaties beschikken bij voorkeur over mechanismen om wijzigingen in de data als event te kunnen publiceren. Batch processen worden zoveel mogelijk voorkomen.
Eigenschappen	Data kwaliteit, Bruikbaarheid, Efficiency

Principe 12. Standaard berichtheader

Statement	Berichten op het integratieplatform hanteren een standaard berichtheader										
Rationale	Een berichtheader wordt gebruikt voor het vastleggen van niet-functionele gegevens van een bericht. In een berichtheader zijn velden opgenomen die belangrijk zijn voor beveiliging, traceerbaarheid en beheer van integraties. Door deze berichtheader consistent te hanteren kunnen generieke mechanismen worden ingezet voor de verwerking van de informatie in de headers van alle berichten. Hiermee kan worden aangetoond dat aan belangrijke eisen wordt voldaan. Door bijvoorbeeld een standaard veld te gebruiken waarin een correlatielD van een event staat kunnen alle berichten in een keten met het zelfde correlatielD gevolgd worden, waarmee end-to-end traceerbaarheid mogelijk wordt.										
Implicaties	Alle berichten hanteren de standaard header zoals beschreven in de (op te stellen) ontwikkelstandaarden en -richtlijnen. AP afnemers leveren zoveel mogelijk zelf de informatie aan die noodzakelijk is voor het vullen van de berichtheader. Indien de afnemer daar zelf niet toe in staat zijn dan wordt deze informatie ingevuld door het integratieplatform. Bij het loggen van berichten op het integratieplatform wordt minimaal de berichtheader opgeslagen. De standaard header bevat geen geheime en/of gevoelige data, zoals wachtwoorden, persoonsgegevens, financiële gegevens, etc.. Het correleren van logrecords is geen kernfunctie van het integratieplatform maar van daartoe ingerichte oplossingen of platformen (CEP, ELK, Splunk,...).										
Eigenschappen	Functionaliteit, Betrouwbaarheid, Traceerbaarheid, Logging										
Voorbeeld	Voorbeeld van header attributen: <table> <tr> <td><code><MessageTimeStamp></code></td><td><i>Tijdstip van het aanmaken van het bericht.</i></td></tr> <tr> <td><code><MessagelD></code></td><td><i>Unieke identifier van het bericht</i></td></tr> <tr> <td><code><ExternalReferencelD></code></td><td><i>External reference nr</i></td></tr> <tr> <td><code><CorrelationId></code></td><td><i>Correlation ID, referentie van bij één flow horende berichten.</i></td></tr> <tr> <td><code><Domain></code></td><td><i>Business domein of functioneel domein van een API</i></td></tr> </table>	<code><MessageTimeStamp></code>	<i>Tijdstip van het aanmaken van het bericht.</i>	<code><MessagelD></code>	<i>Unieke identifier van het bericht</i>	<code><ExternalReferencelD></code>	<i>External reference nr</i>	<code><CorrelationId></code>	<i>Correlation ID, referentie van bij één flow horende berichten.</i>	<code><Domain></code>	<i>Business domein of functioneel domein van een API</i>
<code><MessageTimeStamp></code>	<i>Tijdstip van het aanmaken van het bericht.</i>										
<code><MessagelD></code>	<i>Unieke identifier van het bericht</i>										
<code><ExternalReferencelD></code>	<i>External reference nr</i>										
<code><CorrelationId></code>	<i>Correlation ID, referentie van bij één flow horende berichten.</i>										
<code><Domain></code>	<i>Business domein of functioneel domein van een API</i>										

14 Bijlage II: Definities

Term	Omschrijving
API	Een application programming interface (API) is een verzameling definities op basis waarvan een computerprogramma kan communiceren met een ander programma of onderdeel. Een API definieert de toegang tot de functionaliteit die erachter schuil gaat. De buitenwereld kent geen details van de functionaliteit of implementatie, maar kan dankzij de API die functionaliteit wel gebruiken.
Asynchroon	Asynchroon heeft betrekking op berichtverwerking en houdt in dat het zenden en ontvangen van een bericht niet op hetzelfde moment hoeft plaats te vinden; zender en ontvanger zijn bij asynchrone communicatie van elkaar ontkoppeld in de tijd.
BPM	Business Process management is een procesmethodiek met als doel het optimaliseren van bedrijfsprocessen. Het deel van de BPM processen die kunnen worden geautomatiseerd vallen onder het domein procesintegratie.
CDM	Canonical Data Model or Common Data Model; een ontwerppatroon dat wordt gebruikt om te communiceren tussen verschillende gegevensindelingen.
CEP	Complex Event Processing, een methode om op het zich voordoen van uiteenlopende events te reageren door deze voortdurend te bewaken en verbanden te leggen.
ESB	Enterprise Service Bus, een standaard integratietechnologie dat wordt ingezet voor applicatie integratie doeleinden op basis van messaging en web services.
Event	Een gebeurtenis, binnen of buiten de organisatie, dat is vastgelegd in een digitale beschrijving ervan, op grond waarvan vervolgacties kunnen worden gestart.
Interface	Een koppelvlak via welk met een systeem of applicatie kan worden gecommuniceerd.
OAS	Open API Specificatie, oorspronkelijk bekend als Swagger. Een specificatie voor de beschrijven van RESTful interfaces op basis van YAML.
FTP	File Transfer Protocol, een standaard internet protocol dat wordt gebruikt voor het uitwisselen van bestanden.
HTTP(S)	The Hypertext Transfer Protocol (Secure) is een applicatieprotocol voor gedistribueerde, collaboratieve, hypermedia-informatiesystemen.
JDBC	Java Database Connectivity, een Java-gebaseerde technologie voor databasetoegang.
JMS	Java Message Service, een Java-bericht-georiënteerde middleware specificatie voor het verzenden van berichten tussen twee of meer clients asynchroon.
JSON	JavaScript Object Notation, een lichtgewicht gegevensuitwisselingsformaat. Deze presentatie is zowel machineleesbaar als leesbaar voor de mens.
REST	Representational State Transfer, een architectuurstijl die een beperkte set van beperkingen definieert voor het ophalen en/of bewerken van data objecten.
SOA	Service Oriented Architecture, een software architectuurstijl waarbij applicatiefuncties kunnen worden aangeroepen als webservices.
SOAP	Simple Object Access Protocol, een specificatie van het berichtenprotocol voor het uitwisselen van gestructureerde informatie met webservices.
Stateful	Manier van verwerking waarbij een aanvraag wordt afgehandeld op basis van bestaande status, ook wel toestand geheten, die na uitvoering van de verwerking wordt bijgewerkt.
Stateless	Manier van verwerking waarbij iedere aanvraag volledig moet worden afgehandeld op basis van de informatie die in de aanvraag is verstrekt, zonder context van eerdere en toekomstige aanvragen.
Synchroon	Kenmerk van berichtverwerking met als kenmerk dat het zenden en ontvangen van een bericht op hetzelfde moment plaats vindt, veelal in de context van een vraag-antwoord patroon waarbij de zender actief wacht op een antwoordbericht van de ontvanger.
technical debt	Een 'schuld' die opgebouwd wordt bij software ontwikkeling projecten door voor de korte termijn te kiezen voor een eenvoudige (beperkte) oplossing in plaats van het gebruik van een standaard aanpak die tot beter onderhoudbare oplossingen leidt.
WSDL	Web Service Description Language; een XML-gebaseerde interfacedefinitietaal die wordt gebruikt voor het beschrijven van de functionaliteit die door een webservice wordt aangeboden.

XML	Extensible Mark-up Language; een standaard van het World Wide Web Consortium waarmee men gestructureerde gegevens kan weergeven in de vorm van platte tekst. Deze presentatie is zowel machineleesbaar als leesbaar voor de mens.
YAML	YAML Ain't Mark-up Language, een door de mens leesbare gegevensserialisatietaal.

15 Bijlage III:Bronnen

Titel	Auteur	Datum	Omschrijving
API Strategie Algemeen	Lancelot Schellevis, Forum Standaardisatie Han Zuidweg, Forum Standaardisatie Friso Penninga, Geonovum Matthias Snoei, Swis Jasper Roes, Het Kadaster Peter Haasnoot, Logius	17 1 2020	Nederlandse API Strategie I Geonovum Handreiking Versie ter vaststelling 17 januari 2020
BIO	IBD / VNG Realisatie	2019	Versie 1.03 Baseline Informatiebeveiliging Overheid (BIO)
Decision Model and Notation™ Publication	OMG	Jan 2019	Decision Model and Notation™ Version 1.2 https://www.omg.org/spec/DMN/1.2/ PDF
GEMMA	VNG Realisatie/ VNG	Actueel	GEMMA Online
Technische Infrastructuur standaarden gemeente Tilburg	Perry van Amelsvoort / Erik Manders /Charles Dickens, team Systeembeheer.	25-4-2019	Versie: 3.0
The Forrester Wave™: Strategic iPaaS And Hybrid Integration Platforms, Q1 2019	Forrester analist Henry Peyret	3-1-2019	Rapport dat op basis van 25 criteria iPaaS en HIP leveranciers onderzoekt, analyseert en op basis hiervan een score toekent.
Magic Quadrant for Enterprise Integration Platform as a Service	Gartner analisten Eric Thoo, Massimo Pezzini, Keith Guttridge, Bindi Bhullar	23-4-2019	Rapport dat Enterprise iPaaS oplossingen van verschillende leveranciers onderling afweegt op basis van meerdere criteria
Magic Quadrant for Full Life Cycle API Management	Gartner analisten Paolo Malinverno, Mark O'Neill, Aashish Gupta, Kimihiro Iijima	9-10-2019	Rapport dat API Management oplossingen van verschillende leveranciers onderling afweegt op basis van meerdere criteria