edustandaard

Edukoppeling

M2M gegevensuitwisseling binnen het onderwijs

Architectuur

Edustandaard
Datum: 30 september 2026
Versie: 3.0
Status: concept

Inhoudsopgave

1. Status van dit document 3
1.1. Documenthistorie 3
1.2. Overzicht actuele documentatie en compliance 4
2. Inleiding 5
2.1. Doel van Edukoppeling 5
2.2. Doelgroep 5
2.3. Organisatorisch werkingsgebied Edukoppeling 6
2.4. Functioneel toepassingsgebied Edukoppeling 6
2.5. Positionering van Edukoppeling in het Edustandaard vijflagen model 6
2.6. Positionering van dit document 8
2.7. Leeswijzer 9
2.8. Relevante ontwikkelingen 9
3. Ontwerpprincipes 14
4. Bouwblokken 17
4.1. API 17
4.2. Resource Server 17
4.3. Protected resource 18
4.4. Authorization Server 18
4.5. Clientapplicatie 19
4.6. Clientregistratie 19
4.7. Contractregister 20
4.8. Consumer-intermediair 21
4.9. Publisher-intermediair 22
4.10. Machtiging 23
5. Ketensamenwerking 24
5.1. Inleiding 24
5.2. Grondslagenlaag 24
5.3. Organisatielaag 25
5.4. Informatielaag 27
5.5. Applicatielaag 27
6. Transactiepatronen 29
6.1. Inleiding 29
6.2. Kenmerken van een interactie 30
6.2.1. Richting (eenzijdig / tweezijdig) 30
6.2.2. Overdrachtswijze (push / pull) 31

6.2.3. Interactie (timing) 32
6.2.4. Koppeling (hecht / los) 33
6.2.5. Verwerking (synchroon / asynchroon) 34
6.3. Transactiepatronen 34
6.3.1. Bevraging 35
6.3.2. Melding 36
6.3.3. Melding + bevestiging 37
6.3.4. Bulk synchronisatie (dataset-overdracht) 38
6.4. Ontwerpkaders 39
7. API-architectuur 41
7.1. Ontwerpkaders 41
7.2. Toelichting API-architectuur 42
7.2.1. Technisch vertrouwensmodel 43
7.2.2. Standaarden 44
7.2.3. Gebruik van certificaten en identifiers in een ketentest 45
8. Bijlage A: Begrippen 47

1. Status van dit document

Dit document is de tweede conceptversie van de Edukoppeling-architectuur versie 3.0. Deze architectuur versie 3.0 wijkt sterk af van versie 2.0[1]. In versie 2.0 onderkenden we al RESTful API's, maar dat was nog sterk gericht op webservices en de Digikoppeling-indeling en -documentatie. Deze nieuwe architectuur sluit waar nodig inhoudelijk aan op Digikoppeling, maar de Edukoppeling-documenten zijn niet afhankelijk van specifieke Digikoppeling-documenten en -versies, conform het eerder hierover genomen besluit in de werkgroep Edukoppeling[2]. Verder ondersteunt deze nieuwe versie niet alleen gegevensdiensten die via RESTful API's worden ontsloten maar ook eventvoorzieningen die een belangrijk onderdeel vormen binnen Event-driven architectuur (EDA). We spreken dan ook in meest abstracte zin over API's die betrekking kunnen hebben op gegevensdiensten of eventvoorzieningen. De communicatie-aspecten rond RESTFul API's en EDA worden nader beschreven in het Profiel voor communicatie via APIs. Tevens biedt deze nieuwe architectuur ondersteuning aan token-based access op basis van het Edukoppeling OAuth-profiel.

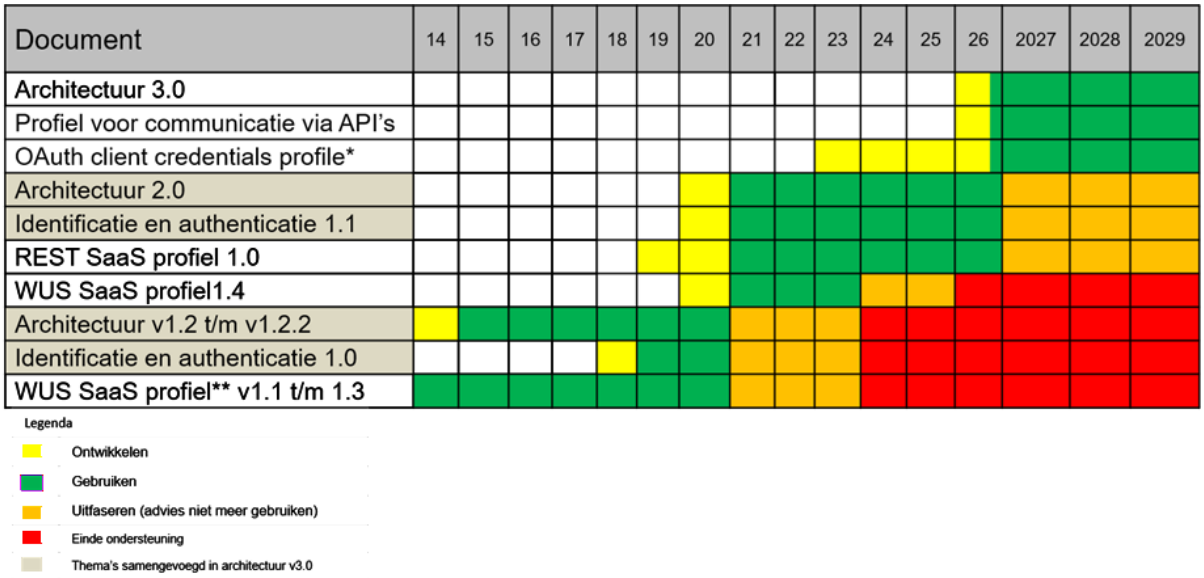
1.1. Documenthistorie

Versie	Status	Auteur	Datum	Opmerking

[1] https://www.edustandaard.nl/app/uploads/2021/10/2021-02-10-Edukoppeling-Architectuur-2.0-definitief.pdf
[2] Besluit 21 d.d. 4-11-2025 (zie besluitenlijst in de verslagen)

1.2.01	Vervallen	WG Edukoppeling	Maart 2015	Zie document release notes
1.2.93	Vervallen	WG Edukoppeling	Juni 2015	Zie document release notes
1.2.94	Vervallen	WG Edukoppeling	Juni 2015	Zie document release notes
1.2.1	Vervallen	WG Edukoppeling	Juli 2017	Zie document release notes
1.2.2	Vervallen	WG Edukoppeling	December 2018	Zie document release notes
2.0	Vastgesteld	WG Edukoppeling	Februari 2021	Zie document release notes
3.0	Concept	WG Edukoppeling	September 2026	Door werkgroep vastgestelde versie

1.2. Overzicht actuele documentatie en compliance



* Van 2023 t/m 2025 zijn OAuth best practices opgesteld ten behoeve van Edu-V en stonden aan de basis van het OAuth cient credentials profiel
** Voorheen Edukopeling Transactiestandaard

Figuur 1 - Overzicht Educkoppeling documenten en versies

2. Inleiding

2.1. Doel van Edukoppeling

Het doel van Edukoppeling is standaardisatie van de technische afspraken op het M2M-koppelvlak waardoor het ontwikkelen van ketensamenwerkingen[3] by design veilig en eenvoudiger wordt. De toepassing van Edukoppeling zorgt ervoor dat, op het niveau van de applicatielaag, gesloten data tijdens transport van de ene naar de andere organisatie niet ongeoorloofd kunnen worden ingezien of gemanipuleerd. De standaard gaat over de afhandeling van berichten (het transport) en niet over de inhoud van berichten.

De aanleiding voor de introductie van Edukoppeling in het onderwijsdomein is een steeds groter wordende stroom van geautomatiseerde machine-to-machine uitwisselingen in het onderwijs. Dit wordt veroorzaakt door vernieuwingen in het onderwijs zelf, in wetgeving en in de beschikbare techniek. In toenemende mate lopen de processen over organisaties heen, tussen onderwijsorganisaties (zowel op bestuursniveau als op het niveau van onderwijsaanbieders, de “scholen”) onderling, tussen onderwijsorganisaties en overheidsorganisaties en tussen onderwijsorganisaties en private onderwijsgerelateerde organisaties. En vaak, als er iets nieuws komt, wordt er dan pas nagedacht over de benodigde wijze van koppelen. Als men niet oppast worden er evenveel verschillende soorten van koppelingen bedacht als er geautomatiseerde gegevensuitwisselingen zijn. Dat is nadelig, omdat hiervoor veel kennis nodig is, dit onnodig veel en kostbaar onderhoud vergt, en dit de interoperabiliteit en aanpasbaarheid hindert. Met Edukoppeling verandert dat. Edukoppeling zorgt voor technische interoperabiliteit[4] en is een meervoudig inzetbare wijze van koppelen. Het beheer is binnen Edustandaard publiek-privaat georganiseerd. Verder is Edukoppeling gebaseerd op internationale open standaarden en de onderwijsstandaarden van Edustandaard. Dit alles maakt dat partijen met een lage drempel kunnen deelnemen, wat gunstig is voor het onderwijs

Edukoppeling bestaat uit op zichzelf staande documenten en waar relevant wordt verwezen naar de (volwassen) internationale open industriestandaarden en onderwijsstandaarden (bijvoorbeeld Edustandaard UBV TLS[5]). Edukoppeling wordt onder Edustandaard beheer doorontwikkeld.

2.2. Doelgroep

Dit document is bedoeld voor ICT-specialisten die betrokken zijn bij het ontwerpen en ontwikkelen van systeem-naar-systeem (M2M) koppelingen voor API’s. Het gaat hier om werknemers (ontwikkelaars, architecten, projectmanagers, informatiemanagers etc.) werkzaam bij onderwijsgerelateerde organisaties, zowel in de publieke als private sector. De lezer van dit document willen wij vragen om zaken die ontbreken of onduidelijk zijn te melden bij de beheerorganisatie Edustandaard[6]. Edustandaard is een open platform waar partijen

[3] ROSA definieert een ketensamenwerking als een concrete invulling van een scenario, waarbij er sprake is van een vorm van gegevensuitwisseling (interactie) tussen twee of meer ketenpartners ter ondersteuning van het afhandelen een of meerdere ketenprocess(tapp)en.
4 Edukoppeling gaat over de envelop, niet de inhoud. Procesmatige en semantische interoperabiliteit zijn buiten scope.
[5] https://www.edustandaard.nl/standaard_afspraken/uniforme-beveiligingsvoorschriften/
[6] Reageren kan via info@edustandaard.nl.

binnen het onderwijsveld bij elkaar komen om afspraken te maken. Hier vindt tevens de doorontwikkeling van de standaard plaats. Hiertoe is een werkgroep Edukoppeling[7] ingericht.

2.3. Organisatorisch werkingsgebied Edukoppeling

Edukoppeling schrijft voor hoe onderwijsorganisaties, publieke uitvoeringsorganisaties, dienstverleners[8] en andere ketenpartners gegevensuitwisselingen opzetten. Edukoppeling heeft als scope alle werkingsgebieden vallend onder alle onderwijssectoren. Zie de werkingsgebieden in ROSA[9]. Hieronder vallen dus alle door de overheid erkende onderwijsorganisaties binnen de sectoren po, vo, bve en ho. Daar waar behoefte is aan technische afspraken op het M2M-koppelvlak, volgens het functioneel toepassingsgebied van Edukoppeling, zijn ketensamenwerkingen binnen deze sectoren verantwoordelijk voor de toepassing ervan. De toepassing buiten het organisatorisch werkingsgebied is toegestaan.

2.4. Functioneel toepassingsgebied Edukoppeling

Het functioneel toepassingsgebied van Edukoppeling is de geautomatiseerde uitwisseling van gesloten gegevens[10] tussen informatiesystemen van onderwijsorganisaties en dienstverleners (onderling, met bedrijven of met de overheid). Deze uitwisseling betreft M2M point-to-point verbinding voor uitwisseling tussen een confidential client en een gesloten API.

Edukoppeling gaat over de afhandeling van de gegevensuitwisseling (het transport) en niet over de inhoud van die uitwisseling ('berichten'). Het is een functioneel technische standaard, maar zal ook aansluiting moeten vinden op kaders van andere architectuurlagen. Hoe Edukoppeling aansluit op bredere afspraken is aan ketensamenwerkingen[11] waarin de standaard als onderdeel van de afspraak of het afsprakenstelsel wordt gevat.

Edukoppeling regelt de volgende ketenfuncties: identificatie, authenticatie, autorisatie en routering, op de 'uitwisselingslaag'. Dit om de zorgen dat gesloten data tijdens transport van de ene naar de andere organisatie, niet ongeoorloofd worden ingezien of gemanipuleerd.

2.5. Positionering van Edukoppeling in het Edustandaard vijflagen model

Het Edustandaard 5-lagenmodel[12] onderkent de volgende lagen:

1. Grondslagenlaag: borgt de juridische basis en beleidskaders waarbinnen gegevensuitwisseling is toegestaan;

---

[7] Voor meer info over de Edukoppeling werkgroep, zie https://www.edustandaard.nl/standaard_werkgroepen/werkgroep-edukoppeling/

[8] Dienstverlener: ROSA Begrippenkader: *Een entiteit die een dienst levert aan een andere entiteit.* Entiteit is doorgaans een organisatie maar kan ook een natuurlijk persoon zijn. De definitie is gebaseerd op het NORA Begrippenkader – Basisbegrippen.

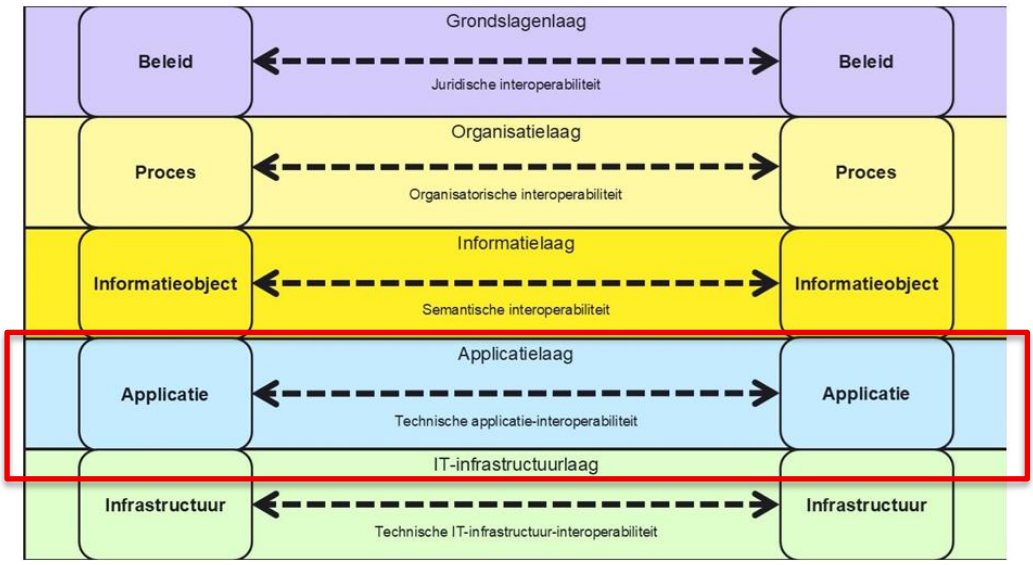
[9] https://rosa.wikixl.nl/index.php/Werkingsgebieden

[10] Het gaat om gegevens waarvoor je geautoriseerd moet worden voor toegang. De gegevens zijn niet voor publiek hergebruik beschikbaar. Dit kan om verschillende redenen zijn, zie https://data.overheid.nl/gesloten-datasets

[11] Ketensamenwerkingen zijn bijvoorbeeld OKE, Edu-V, ROD ec. (zie ook: https://rosa.wikixl.nl/index.php/Begrip:27a6accf-472d-4415-bc5b-1e9de17bf288)

[12] AMIGO-methodiek-1.1.0-1.pdf en https://rosa.wikixl.nl/index.php/Interoperabiliteit_en_het_Edustandaard_lagenmodel#Opbouw_van_het_lagenmodel

2. Organisatorische laag: ketensamenwerking afspraken over wie welke rol heeft, welke gegevensdiensten, interfaces en interactiepatronen er zijn en welke gegevens onder welke condities uitgewisseld worden;
3. Informatielaag: semantiek, waaronder gegevensdefinities, informatiemodellen en de gebruikte identifiers voor rechtspersonen en natuurlijke personen;
4. Applicatielaag: API's en hun beveiligingsprofielen, berichtspecificaties, payload beveiliging, interactiepatronen en foutafhandeling;
5. IT-infrastructuurlaag: transportprotocollen en technische beveiligingsmechanismen zoals TLS.



Figuur 2 - Edustandaard 5-lagenmodel

Edukoppeling richt zich primair op de technische applicatie-interoperabiliteit. De focus ligt op de logistieke aspecten van gegevensuitwisseling: hoe gegevens, API-request-responses en events tussen ketenpartners worden getransporteerd, hoe ketenpartners elkaar en hun voorzieningen identificeren en authenticeren, hoe autorisatie plaatsvindt en hoe de betrouwbaarheid en beveiliging van de uitwisseling worden geborgd. Edukoppeling levert hiermee ondersteuning aan de applicatielaag binnen het Edustandaard vijflagenmodel. Binnen de organisatielaag worden afspraken gemaakt over de samenwerking tussen organisaties waarmee vastgesteld wordt of er organisatorische interoperabiliteit mogelijk is. De proceskarakteristieken bepalen in belangrijke mate welke interactiepatronen passend zijn, bijvoorbeeld een synchrone bevraging of een eventgedreven (event driven) notificatie. Binnen de informatielaag worden afspraken gemaakt over welke gegevens moeten worden uitgewisseld en onder welke voorwaarden dit plaatsvindt. Deze laag zorgt voor semantische interoperabiliteit, zowel verzender als ontvanger kennen dezelfde betekenis toe aan gegevens en gebeurtenissen. Binnen API-gebaseerde gegevensuitwisseling komt dit onder meer tot uitdrukking in OpenAPI- en AsyncAPI-specificaties, JSON-schema's en afspraken over CloudEvents-typen.

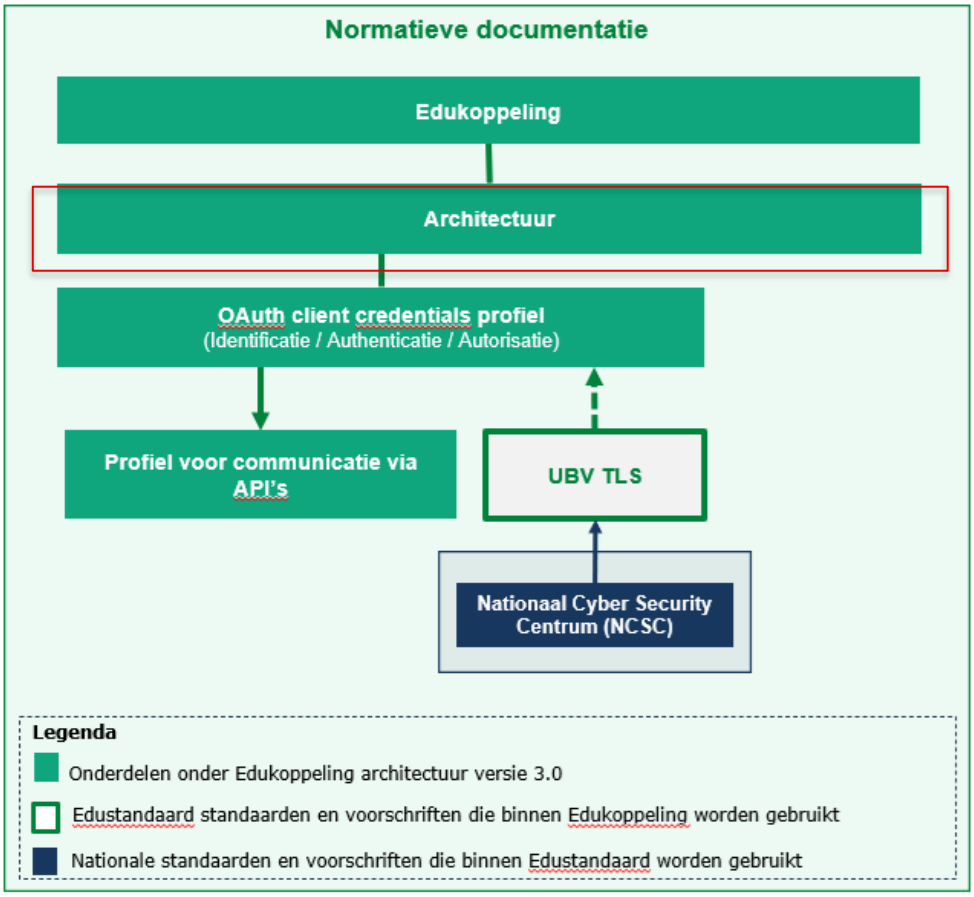
Procesafspraken bepalen waarom gegevens worden uitgewisseld, semantische afspraken beschrijven wat wordt uitgewisseld, en Edukoppeling beschrijft hoe deze uitwisseling technisch wordt gerealiseerd. Edukoppeling wordt samen met de afspraken van de overige

architectuurlagen[13] opgenomen in het afsprakenstelsel van de betreffende ketensamenwerking[14].

2.6. Positionering van dit document

Edukoppeling bestaat uit verschillende documenten en is in beheer bij Edustandaard. Dit Architectuurdocument is één van de normatieve documenten binnen de Edukoppeling-afspraak. In Figuur 3 - Positionering van Architectuur binnen Edukoppeling is een schematische weergave opgenomen.

Dit architectuurdocument geeft context bij de toepassing van de nieuwe Edukoppeling profielen. Het biedt geen ondersteuning aan het voormalige WUS SaaS-profiel[15]. Voor het REST SaaS-profiel[16] geldt dat deze onderdeel is van architectuur versie 2.0. Ook dit profiel wordt niet meegenomen in deze nieuwe architectuur. De toepassing van WUS-SaaS en REST-SaaS valt dus onder de architectuurversie 2.0. We verwachten dat de komende jaren versie 2.0 en bijbehorende profielen minder en minder toegepast zullen worden. Zie Overzicht actuele documentatie en compliance.



Figuur 3 - Positionering van Architectuur binnen Edukoppeling

[13] Zie Positionering van Edukoppeling in het Edustandaard vijflagen model
[14] Ketensamenwerkingen zijn bijvoorbeeld OKE, Edu-V, ROD ec. (zie ook: https://rosa.wikixl.nl/index.php/Begrip:27a6accf-472d-4415-bc5b-1e9de17bf288)
[15] Dit profiel is niet meer in beheer (status 'Einde ondersteuning')
[16] Routerings- en mandaatgegevens worden hierbij meegegeven in een querystring

2.7. Leeswijzer

Dit document is opgebouwd volgens de logische structuur van gegevensuitwisseling binnen een ketensamenwerking. Als eerste worden een aantal architectuurprincipes benoemd waarmee we aansluiten op ROSA en de architectuur kaderen (hoofdstuk 3). Het volgende hoofdstuk geeft de relevantie van een ketensamenwerking weer (hoofdstuk 4). Daarna worden de kernfuncties van de gegevensuitwisseling toegelicht bij transactiepatronen (hoofdstuk 5) en het laatste hoofdstuk beschrijft de API-architectuur waar deze versie op toegespitst is.

2.8. Relevante ontwikkelingen

Deze nieuwe versie van de architectuur beschrijft hoe binnen een ketensamenwerking op een gestandaardiseerde manier synchroon of asynchroon machine-to-machine gesloten data kan worden uitgewisseld via API's. Met de toepassing van de OAuth client credentials grant kunnen we toegang nu baseren op basis van tokens met grofmazige autorisaties op basis van scopes. Ook kan in deze flow metadata[17] meegestuurd worden waarmee extra controles kunnen worden uitgevoerd bij de Authorisation Server en/of Resource Server. Deze architectuur staat op zichzelf, maar is deels geïnspireerd door verschillende (lopende) initiatieven bij de overheid, zoals het Kennisplatform API's, Digikoppeling en verschillende internationale open standaarden. Deze blijven we nauwlettend volgen en worden hieronder nader toegelicht.

- **Kennisplatform API's**
  Geonovum[18] is samen met Bureau Forum Standaardisatie, Kamer van Koophandel, VNG Realisatie, Logius en het Kadaster in 2018 het Kennisplatform API's begonnen. Het Kennisplatform API's wil API's beter bij de vraag aan laten sluiten, kennis over het toepassen van API's uitwisselen en de aanpak bij verschillende organisaties op elkaar afstemmen en waar nodig standaardiseren. In werkgroepen die bestaan uit deelnemers uit de publieke en private sector zijn verschillende resultaten behaald. De API-strategie van het Kennisplatform omvat algemene onderdelen, zoals een beschrijving van architectuur en gebruikerswensen. Daarnaast is er een normatief deel[19] dat op de lijst met verplichte standaarden van de Nederlandse overheid staat (Forum Standaardisatie) met o.a. de volgende standaarden NL GOV Assurance profile for OAuth 2.0 (NL GOV OAuth profiel), Open API Specification (OAS) en API Design Rules (ADR).

  Voor Edukoppeling is met name het NL GOV OAuth profiel v1.1.0 relevant. De OAuth best practices[20] die zijn opgesteld ten behoeve van het Edu-V groeifondsprogramma is gebaseerd op deze versie van het NL GOV OAuth profiel. Dit NL GOV OAuth profiel is op zijn beurt weer gebaseerd op het OpenID iGOV OAuth profiel (Draft 03). Beide profielen hebben een brede scope en ondersteunen

[17] Vergelijkbaar met de ondersteuning van de edu-to en edu-from parameters bij het WUS-SaaS en REST-SaaS profiel van architectuurversie 2.0.
[18] Kennisplatform API's | developer.overheid.nl
[19] Zie github voor een compleet overzicht van onderdelen van de API-strategie
[20] Zie de voorlopige uitwerking die in 2024 is gepubliceerd (Edukoppeling - Edukoppeling - juni 2024 - Edustandaard).

verschillende use cases[21] en toepassing van zowel confidential als public clients. Op dit moment ontwikkelt OpenID een nieuwe versie van het OpenID iGOV OAuth profiel (Draft 09). Deze gaat in de basis uit van enkel confidential clients[22]. De nieuwe versie van het NL GOV OAuth profiel (Draft March 09, 2026[23]), die nog in ontwikkeling is, is vooralsnog gebaseerd op de vorige versie (Draft 03) van het iGOV OAuth profiel. Voor Edukoppeling is (nu) met name de toepassing van confidential clients en de client credentials grant relevant. Verder hebben we bij het opstellen van de Edukoppeling OAuth Best Practices gemerkt dat het zowel documentatie-technisch als beheermatig lastig is om onze eigen keuzes te vatten in een document dat is afgeleid van het NL GOV OAuth profiel. We zien eerder nadelen dan voordelen. Wel hebben we bij het opstellen van het Edukoppeling OAuth profiel onze eigen keuzes gemaakt, waarbij we in eerste plaats wel naar het NL GOV OAuth profiel hebben gekeken. We blijven de ontwikkelingen van NL GOV OAuth volgen.

In het Edukoppeling REST SaaS-profiel hebben we destijds het advies opgenomen om de API Design Rules[24] toe te passen. Later hebben we in de OAuth best practices de toepassing van de API Design Rules[25] verplicht gesteld. De werkgroep heeft voor dit document (Edukoppeling-architectuur versie 3.0) en de onderliggende profielen besloten om het ontwerp van API's buiten scope te plaatsen en hiermee ook de toepassing van de API Design Rules. We geven ketenpartners die in het kader van ketensamenwerkingen API's ontwikkelen wel het advies hun voordeel te doen met het bestaan van de API Design Rules.

- **Logius**
  De vorige versies van Edukoppeling waren gebaseerd op de overheidsstandaard Digikoppeling[26]. We hebben de afgelopen tijd een aantal keer ervaren dat de doorontwikkeling rond deze standaard achter liep op waar we binnen het onderwijs behoefte aan hebben. Dit was eigenlijk ook al de aanleiding in 2014 voor het creëren van de Edukoppeling-standaard. De behoefte voor een profiel voor RESTful API's heeft meer recent geleid tot aansluiting op het Kennisplatform API's[27]. De Edukoppeling OAuth Best Practices zijn zoals eerder aangegeven gebaseerd op het NL GOV OAuth 2.0 profiel[28] van het Kennisplatform API's. We merken dat als we voorlopen, Digikoppeling ons soms volgt, maar men maakt soms daar ook weer (andere) grote stappen waar we als onderwijs (nog) niet op willen of zelfs kunnen aansluiten. Dit is in de huidige situatie aan de orde doordat het Digikoppeling REST profiel[29] de toepassing van Federated Service Connectivity (FSC[30]) verplicht stelt. Dit past (nu) niet goed bij onze huidige afspraken in het onderwijs. Hoog over wordt met FSC invulling gegeven aan wat we binnen Edukoppeling al sinds 2014 toepassen

[21] Client credentials flow en Authorization code flow
22 Een confidential client is een client die draait in een omgeving waar vertrouwelijke gegevens (waaronder het client secret) veilig bewaard kunnen worden (bijvoorbeeld server-side webapplicaties).
[23] Zie editor's draft versie NL GOV Assurance profile for OAuth 2.0
[24] API Design Rules versie 17-01-2020
[25] NLGov REST API Design Rules 2.1.0
[26] Digikoppeling Overzicht Actuele Documentatie en Compliance 1.12.2
[27] Kennisplatform API's | developer.overheid.nl
[28] NL GOV Assurance profile for OAuth 2.0 v1.1.0
[29] Digikoppeling Koppelvlakstandaard REST-API 3.0.1
[30] Introductie | standaard.nl

met de inzet van mandaten en het Onderwijs serviceregister (OSR[31]). Destijds ging Digikoppeling (WUS/REST) uit van een point-to-point verbinding uit voor de uitwisseling van gesloten data tussen twee ketenpartners. De eerste versies van Edukoppeling voegde daar, afhankelijk van het scenario, één of twee ketenpartners aan toe in de rol van eindorganisatie. De eindorganisatie, over het algemeen de onderwijsorganisatie, geeft de ketenpartners toestemming om via de point-to-point verbinding de gesloten data uit te wisselen. Ook FSC ondersteunt verschillende scenario's en rollen[32] waarbij een bepaalde ketenpartner een andere toestemming geeft voor een bepaalde gegevensuitwisseling. Voordat ketenpartners met elkaar kunnen communiceren, leggen ze afspraken vast in een digitaal contract. Hierin staat precies welke API's door een organisatie gebruikt mogen worden. Deze contracten worden cryptografisch ondertekend, zodat ze niet kunnen worden aangepast zonder toestemming. Juist het gebruik van contracten die met (PKIoverheid) certificaten ondertekend moeten worden levert voor Edukoppeling een probleem op omdat onderwijsorganisaties dan verplicht gebruik moeten gaan maken van deze certificaten, iets wat in het verleden tot veel rompslomp leidde bij die organisaties. We zijn met Edukoppeling juist afgestapt van de verplichting dat een onderwijsorganisatie certificaten[33] moet gebruiken. Momenteel is men een HLD Signing service aan het ontwikkelen die een ketenpartner in staat stelt om het contract te ondertekenen zonder dat men over een certificaat hoeft te beschikken. De verwachting is dat op termijn ook in de FSC (core) standaard de toepassing van een ondertekenservice ondersteund gaat worden. We blijven de ontwikkelingen rond het Digikoppeling REST profiel en FSC daarom volgen.

Het Digikoppeling REST-profiel ondersteunt ondertekening en versleuteling van de payload[34]. Deze versie van de Edukoppeling-architectuur gaat nog uit van scenario's waar er point-to-point verbindingen zijn tussen ketenpartners. Wanneer er behoefte is aan het ondertekenen en/of versleutelen van de payload dan zullen we meer inhoudelijk naar de opties van het Digikoppeling REST-profiel[35] gaan kijken.

Digikoppeling bestaat uit verschillende documenten. Zo is er ook het document Digikoppeling Beveiligingstandaarden en voorschriften 2.0.1. Hierin staan onder andere voorschriften rond identificatie (OIN[36]) en authenticatie (PKIoverheid certificaat), maar ook voorschriften rond (m)TLS. Al sinds 2020 hebben we binnen Edustandaard de voorschriften rond TLS opgenomen in een aparte standaard, UBV TLS[37]. We maken hier sinds 2021 ook binnen Edukoppeling gebruik van. We zijn op dit punt dus al enige tijd ontkoppeld van Digikoppeling. Zowel Digikoppeling als de Edustandaard werkgroep IBP zien NSCS[38] als belangrijke bron voor de te maken keuzes rond TLS.

[31] Zie Welkom bij het Onderwijs serviceregister
[32] De rol van Delegator die aan de Delegatee toestemming geeft, zie FSC - Core 1.1.2
[33] Dat waren toentertijd DUO ODOC certificaten
[34] https://gitdocumentatie.logius.nl/publicatie/dk/restapi/3.0.1/#signing-encryptie-in-http-rest-context
[35] ADR Modules ADR-HTTP Message and payload signing with JAdES en ADR-HTTP Payload encryption
[36] Digikoppeling ligt de toepassing van OIN en PKIoverheid certificaten nader toe in een apart document Digikoppeling Identificatie en Authenticatie 1.5.0.
[37] https://www.edustandaard.nl/standaard_afspraken/uniforme-beveiligingsvoorschriften/
[38] NCSC - Home

Logius heeft sinds 2022 een NL GOV profile for CloudEvents[39] (NL GOV CloudEvents profiel) in beheer. Deze versie is gebaseerd op de CloudEvents specificatie die is ontwikkeld door de Serverless Working Group van de Cloud Native Computing Foundation. In 2024 is er een publieke consultatie geweest waarna versie 1.0 van het NL GOV CloudEvents profiel is gepubliceerd. Op 17 maart 2026 is versie 1.1[40] vastgesteld. Deze is gebaseerd op *CloudEvents - Version 1.0.1*. Het NL GOV CloudEvents profiel is een set Nederlandse afspraken over het gebruik van de internationale standaard CloudEvents. Het beschrijft hoe een plaatsgevonden gebeurtenis gerapporteerd en uitgewisseld kan worden, zoals een verhuizing of overlijden. Het doel is om gestandaardiseerd en effectief informatie uit te wisselen op een manier waarbij informatie centraal (bij de bron) geregistreerd blijft. Binnen het Groeifondsprogramma Npuls werken Studielink, SURF en MBO Digitaal samen aan het onderbrengen van het huidige Studielink / Cambo in één nieuwe centrale voorziening voor het Aanmelden, Inschrijven en Intekenen (ook wel project AII[41]). Het project heeft behoefte aan asynchrone communicatie via RESTful API's om een zogenaamde Event Driven Architecture (EDA) te kunnen realiseren. Hierbij wil men ook een asynchrone interactie kunnen toepassen. Om hierbij op een betrouwbare en robuuste manier gegevens uit te wisselen zijn er richtlijnen nodig zodat ook bij tijdelijke storingen het resultaat deterministisch is. Het nieuwe Edukoppeling Communicatie-profiel bevat deze richtlijnen. Deze richtlijnen staan in algemene zin al in het NL GOV CloudEvents profiel. Hierin missen we echter aansluiting op de Edukoppeling architectuur waarin we o.a. beveiliging regelen met ons eigen OAuth client credentials profiel en uitgaan van point-to-point communicatie tussen intermediairs die in een EDA context bepaalde functies voor hun rekening nemen.

- **PKIoverheid G4**
  PKIoverheid migreert naar een nieuwe generatie certificaten (G4[42]). Deze migratie is relevant voor Edukoppeling omdat PKIoverheid-certificaten kunnen worden toegepast als technisch vertrouwensanker. Ketenpartners moeten eigen certificaten gaan vervangen, maar moeten er ook voor zorgen dat systemen deze certificaten kunnen valideren. De nieuwe G4-(private)root certificaten worden niet automatisch door besturingssystemen en software vertrouwd. Afhankelijk van de toepassing moeten daarom de relevante G4-rootcertificaten en eventueel intermediate certificaten vooraf aan truststores worden toegevoegd. Ook moet worden gecontroleerd of de gebruikte software en infrastructuur de voor G4 relevante cryptografie, waaronder RSASSA-PSS[43], ondersteunt. Logius adviseert deze voorbereiding en het testen van G4 expliciet uit te voeren voordat productiecertificaten worden ingezet.
  De migratie kent een overgangsperiode waarin bestaande G3- en G1 Private-certificaten en de nieuwe G4-certificaten naast elkaar kunnen voorkomen. Bestaande certificaten hoeven niet direct te worden vervangen en blijven geldig tot hun eigen verloopdatum, maar uiterlijk tot november 2028. Ketenpartners moeten gedurende

[39] Logius | NL GOV profile for CloudEvents
[40] NL GOV profile for CloudEvents 1.1
[41] Aanmelden, Inschrijven, Intekenen | Npuls
[42] Zie Logius | Overstappen op de nieuwe generatie PKIoverheid-certificaten (G4) en Logius | Veelgestelde vragen G4-migratie
[43] G4 uses a modern cryptographic algorithm to sign all certificates: RSASSA-PKCS1-v1_5 has been replaced by RSASSA-PSS. Zie Logius | New generation of PKIoverheid Certificates

deze overgang rekening houden met het gelijktijdig kunnen valideren van de voor hun Edukoppeling-uitwisselingen relevante bestaande en nieuwe certificaatketens.

- **Federatief datastelsel (FDS)**[44]
  In de Meerjarenvisie Digitale Overheid 2025 – 2030[45] wordt aangegeven dat bestaande gegevensuitwisseling binnen de overheid nu vaak als niet transparant genoeg wordt ervaren. En er zijn voorbeelden van gebruik van foutieve gegevens in registraties, van onrechtmatig gegevensgebruik, of onveilige gegevensdeling. Er worden verschillende oorzaken genoemd en men ziet een organisatie-overstijgend gegevenslandschap als de oplossing. Deze lijn wordt ook in de Nederlandse Digitaliseringsstrategie neergezet die ook aansluit op ontwikkelingen binnen de Europese Unie[46]. Een verbeterde inrichting van het data-ecosysteem van de Nederlandse overheid wordt gezien in het federatief datastelsel. De Interbestuurlijke Data Strategie (IBDS) is een programma van het ministerie van BZK waar het FDS voor gegevensdeling en -integratie wordt ontwikkeld. Het FDS bestaat enerzijds uit een afsprakenstelsel voor de uitwisseling van gegevens en anderzijds uit de standaarden[47] en tools waarmee de uitwisseling ingericht kan worden. Hier wordt in een organisatie-overstijgend gegevenslandschap voorzien dat overheidsorganisaties én private partijen moet ontzorgen rond gegevensdeelvraagstukken. Wijzigingen in opgeslagen gegevens die voor andere overheidsorganisaties van belang zijn, worden via een abonnementsysteem aan de abonnees doorgegeven. Het lijkt er dus op dat FDS onder andere het Digikoppeling REST-profiel (met de FSC-standaard) en als onderdeel van de EDA-architectuur het gebruik van de NL GOV CloudEvents-standaard gaan voorschrijven. Voor de Edukoppeling-standaard is het van belang om de ontwikkelingen rond FDS en GDI[48] te volgen. We willen niet alleen, waar wenselijk, met Edukoppeling aansluiten op (onderdelen van) overheidsstandaarden, maar willen ook oog hebben voor de grenzen van de werkingsgebieden van relevante overheidsstandaarden en Edukoppeling.

[44] Wat is het Federatief Datastelsel? · Federatief Datastelsel · Realisatie IBDS
[45] Meerjarenvisie Digitale Overheid 2025 – 2030 | Versie 2025
[46] Data Spaces | Interoperable Europe Portal
[47] Standaarden voor het FDS · Federatief Datastelsel · Realisatie IBDS
[48] RE010 - Selectie FDS-standaarden – GDI-standaarden · Federatief Datastelsel · Realisatie IBDS

3. Ontwerpprincipes

1 Edukoppeling realiseert de technische logistiek voor vertrouwelijke gegevensuitwisseling[49]
- Rationale: De technische logistiek kan in uiteenlopende onderwijsketens worden toegepast.
- Implicaties
  - Edukoppeling heeft betrekking op de applicatielaag en de beveiliging van gesloten data[50] in transport en toegang tot API's
  - Edukoppeling gaat niet over het design van API's en schrijft geen bedrijfsprocessen of informatiemodellen voor.
  - Domeinspecifieke afspraken worden buiten Edukoppeling beheerd.

2 Iedere architectuurlaag[51] stelt kaders aan de onderliggende laag
- Rationale: Edukoppeling staat los van processen en semantiek, maar de keuze voor een API, interactiepatroon of beveiligingsmechanisme ontstaat niet vanuit technologie. De eisen die processen, informatie en samenwerking eraan stellen zijn bepalend. Hierdoor blijft techniek een middel om de samenwerking te ondersteunen en geen doel op zich.
- Implicaties
  - Organisatie- en procesafspraken stellen eisen aan de applicatielaag. De kenmerken van het proces bepalen welk interactiepatroon en welke architectuurstijl wordt toegepast
  - De informatielaag bepaalt de semantiek van gegevens en gebeurtenissen.
  - De applicatielaag realiseert deze afspraken met API's, events en beveiliging.
  - De technologische infrastructuur ondersteunt deze applicaties.

3 De betekenis van gegevens wordt vastgesteld in de informatielaag en technisch gerealiseerd in de applicatielaag.
- Rationale: Er is een scheiding tussen inhoud en logistiek. De informatielaag bepaalt de inhoud en betekenis van gegevens. Edukoppeling bepaalt de technische context waarbinnen deze gegevens veilig en interoperabel kunnen worden uitgewisseld. De API-architectuur richt zich op de envelop: interactie, beveiliging, autorisatiecontext en transport. Edukoppeling maakt wel gebruik van gegevens uit de informatielaag, zoals het organisatie identificatienummer (OIN[52]).
- Implicaties
  - Relevante identificerende kenmerken, onderdelen van API-contracten en eventcontracten sluiten aan op informatiemodellen.

[49] https://rosa.wikixl.nl/index.php/Id-fd10bf7a-e960-4484-8e19-bd8b86e7d911
[50] Het gaat om gegevens waarvoor je geautoriseerd moet worden voor toegang. De gegevens zijn niet voor publiek hergebruik beschikbaar. Dit kan om verschillende redenen zijn, zie https://data.overheid.nl/gesloten-datasets
[51] Zie Positionering van Edukoppeling in het Edustandaard vijflagen model
[52] OIN, zie Digikoppeling I&A https://gitdocumentatie.logius.nl/publicatie/dk/idauth/

4 Open standaarden zijn leidend: De architectuur baseert zich zoveel mogelijk op breed geaccepteerde open standaarden.
   - Rationale: Ketensamenwerkingen in het onderwijs bestaan uit een diversiteit aan ketenpartners (toezichthouders, onderwijsorganisaties, publieke en private dienstverleners). Toepassing van open standaarden voorkomt leveranciersafhankelijkheid, vergroot interoperabiliteit en maakt hergebruik van bestaande producten, libraries en kennis mogelijk. Door zoveel mogelijk aan te sluiten bij internationale standaarden blijft Edukoppeling toekomstvast en kunnen nieuwe ontwikkelingen eenvoudiger worden overgenomen.
   - Implicaties:
     - Edukoppeling specificeert profielen op basis van bestaande open standaarden, zoals OAuth 2.0 en ontwikkelt geen eigen protocollen.
     - Afwijkingen of sectorspecifieke aanvullingen zijn onwenselijk, maar blijven mogelijk, mits besproken in de Edukoppeling werkgroep zodat ook overwogen kan worden om de standaard aan te passen.
     - We voorkomen dat partijen onnodig maatwerk moeten ondersteunen.

5 Wanneer een organisatie namens een andere organisatie de gegevensuitwisseling uitvoert, moet dit onderdeel zijn van de autorisatiebeslissing.
   - Rationale: De authenticatie van een client zegt niet namens welke organisatie deze client de interactie uitvoert. In ketens waarin dienstverleners diensten uitvoeren voor een of meer onderwijsorganisaties moet de context van iedere gegevensuitwisseling expliciet kunnen worden beoordeeld.
   - Implicaties
     - Wanneer nodig moet er voor de autorisatiebeslissing een machtiging[53] geraadpleegd kunnen worden.

6 Edukoppeling sluit aan op de architectuurkaders van ROSA.
   - Rationale: ROSA vormt de sectorbrede referentiearchitectuur voor de informatievoorziening in het onderwijs en beschrijft de samenhang tussen organisaties, processen, informatievoorziening en technologie. Edukoppeling geeft invulling aan het onderdeel technische interoperabiliteit binnen de applicatielaag. Het realiseert de veilige, betrouwbare en gestandaardiseerde gegevensuitwisseling. Edukoppeling sluit aan bij de relevante ROSA-kaders, waaronder het IV-domein Gegevensuitwisseling, het ROSA-procesmodel Inrichten gegevensinteractie, de ROSA-architectuurprincipes en de bijbehorende ontwerpkaders.
   - Implicaties
     - Edukoppeling sluit voor begrippen, definities aan op het ROSA-begrippenkader[54]. Edukoppeling definieert uitsluitend aanvullende begrippen wanneer deze noodzakelijk zijn voor de (beschrijving van de) technische logistiek van gegevensuitwisseling[55].

[53] Edukoppeling onderkent het concept van een machtiging, maar definieert geen machtigingsregister. In vorige versies van Edukoppeling spraken we van een mandaat en het gebruik van een SaaS-dienst door een Eindorganisatie. In deze versie hebben we een meer algemene context en spreken we van een machtiging.
[54] Zie ROSA begrippenkader: https://rosa.wikixl.nl/index.php/Alfabetisch_overzicht_ROSA_Begrippenkader
[55] Zie Bijlage A: Begrippen

- De architectuurkeuzes binnen Edukoppeling zijn herleidbaar naar de relevante architectuurprincipes en ontwerpkaders van ROSA.
- Edukoppeling beschrijft uitsluitend de technische uitwerking van gegevensuitwisseling en neemt geen verantwoordelijkheid over voor proces-, informatie- of organisatiearchitectuur die binnen ROSA zijn belegd.
- Het ROSA IV-procesmodel ‘Inrichten gegevensinteractie’ vormt het uitgangspunt voor de inrichting en standaardisatie van gegevensuitwisseling tussen ketenpartners.

7 Transportbeveiliging is altijd verplicht: Alle gegevensuitwisseling binnen Edukoppeling vindt plaats via een beveiligd transportkanaal.

- Rationale: Gegevens die tussen organisaties worden uitgewisseld moeten tijdens transport worden beschermd tegen afluisteren, manipulatie en ongeautoriseerde toegang. Transportbeveiliging vormt een fundamentele voorwaarde voor vertrouwelijkheid, integriteit en authenticiteit van de gegevensuitwisseling. Deze bescherming staat los van de toepassing van identificatie, authenticatie en autorisatie en geldt voor iedere interactie, ongeacht het gekozen interactiepatroon of de aard van de uitgewisselde gegevens.
- Implicaties
  - De gegevensuitwisseling zelf en de beveiligingsgegevens worden uitsluitend over beveiligde verbindingen uitgewisseld.
  - Edukoppeling gaat uit van een point-to-point[56] verbinding tussen gegevensaanbieder en gegevensafnemer[57]

[56] Point-to-point betekent hier dat er een rechtstreekse verbinding wordt ingericht tussen de communicerende systemen.
[57] Tussenliggende intermediairs worden vooralsnog niet in beschouwing genomen. Ook worden ondertekening en versleuteling van de payload niet ondersteund. Binnen de EDA-context gaat Edukoppeling uit van een beveiligde point-to-pointverbinding tussen het publisher-intermediair en consumer-intermediair. End-to-end beveiliging van de payload over meerdere tussenliggende partijen valt buiten de scope van deze versie.

4. Bouwblokken

De architectuur gaat uit van een aantal bouwblokken[58] die gezamenlijk een veilige en beheersbare gegevensuitwisseling binnen een ketensamenwerking realiseren.

4.1. API

Een API vormt het gestandaardiseerde toegangspunt waarmee een ketenpartner gegevens of functionaliteit beschikbaar stelt aan andere ketenpartners. De API abstraheert de achterliggende implementatie en biedt een stabiel technisch contract waartegen andere applicaties kunnen integreren. Afhankelijk van het gekozen interactiepatroon realiseert een API een gegevensdienst of een eventvoorziening.

Gegevensdienst
Bij synchrone gegevensuitwisseling realiseert de API een gegevensdienst in de vorm van een RESTful API. De API stelt één of meer resources of functies beschikbaar via gestandaardiseerde HTTP-endpoints. De API verwerkt een verzoek direct of retourneert een technische respons wanneer de inhoudelijke verwerking op een later moment plaatsvindt.

Eventvoorziening
Een API wordt bij asynchrone gegevensuitwisseling gerealiseerd door een eventvoorziening. Deze heeft twee vormen, een publisher-intermediair aan de verzendende zijde en een consumer-intermediair aan de ontvangende zijde. De intermediairs vormen het gestandaardiseerde en beveiligde koppelvlak tussen de vertrouwensdomeinen van de ketenpartners. Zij bieden, afhankelijk van hun rol binnen een interactie, concrete API's aan voor eventlevering, subscriptionbeheer en eventuele operationele beheerfuncties.

4.2. Resource Server

Dit bouwblok is een OAuth-rol. De gegevensdienst of eventvoorziening vervult voor een concrete interactie de rol van Resource Server wanneer deze een beveiligde dienst of functionaliteit beschikbaar stelt waarvoor een geldig Access Token vereist is. De Resource Server valideert het Access Token en beoordeelt of de gevraagde interactie binnen de verleende autorisatie valt. Pas nadat deze controles succesvol zijn uitgevoerd, wordt toegang tot de beschermde gegevens of functionaliteit verleend. De Resource Server realiseert daarmee de technische handhaving van het autorisatiebeleid bij de feitelijke gegevensuitwisseling.

De uitgifte van een Access Token door de Authorization Server betekent niet dat iedere aanvraag automatisch moet worden toegestaan. De Resource Server neemt voor iedere interactie de uiteindelijke toegangsbeslissing. Daarbij kan hij, naast de informatie in het Access Token, aanvullende controles uitvoeren op de opgevraagde resource, de gevraagde handeling, de organisatorische context en eventuele actuele beleidsregels.
Functies:

1. beschikbaar stellen en beschermen van één of meer protected resources;
2. valideren van Access Tokens;

[58] Logische voorziening die functies (in een bepaalde rol) realiseert

3. afdwingen van scopes, acties en authorization_details;
4. toepassen van aanvullende autorisatie op basis van onder meer edu-from en edu-to;
5. routeren: selecteren van de juiste tenant of organisatorische context;
6. uitvoeren van de gevraagde API-operatie, accepteren van een event of verwerken van een subscriptionverzoek;
7. retourneren van een passende technische response of ontvangstbevestiging;

4.3. Protected resource

Een protected resource is functionaliteit of een gegevensbron waartoe alleen toegang kan worden verkregen met een geldig en passend Access Token. De Resource Server beoordeelt bij iedere interactie of de gevraagde toegang mag worden verleend. Een protected resource is daarmee de concrete functionaliteit van een API die door een Resource Server wordt beschermd. Een API kan één of meerdere protected resources bevatten.

Bij een gegevensdienst, een RESTful API, bestaat een protected resource uit één of meer API-endpoints of resources waarop een client gegevens kan raadplegen of wijzigen.

Bij een eventvoorziening worden de protected resources ontsloten via één of meer API's die door een publisher- of consumer-intermediair worden aangeboden. Afhankelijk van de functie van de intermediair kan dit onder meer een Publication API, Subscription API of Management API zijn. Iedere API kan één of meerdere protected resources beschikbaar stellen waarvoor een geldig Access Token vereist is. Voorbeelden van protected resources in de context van eventvoorzieningen zijn:

1. het publiceren van een gebeurtenis;
2. het ontvangen van een gebeurtenis;
3. het beheren van subscriptions;
4. het beheren van afleverpunten;
5. het raadplegen van contractinformatie;
6. het uitvoeren van beheer- of monitoringsfuncties.

4.4. Authorization Server

De Authorization Server realiseert de gelijknamige OAuth-rol. Edukoppeling schrijft het OAuth 2.0 Client Credentials profiel voor en hierbij authenticeert de Authorization Server client en verstrekt tokens waarmee toegang tot API’s kan worden verkregen. De Authorization Server vormt daarmee een belangrijk onderdeel van het technisch vertrouwensmodel binnen de ketensamenwerking. Tijdens onboarding worden clients, publieke sleutels, scopes, eventuele authorization_details en vertrouwensrelaties geregistreerd. Een implementatie kan het bestaan uit meerdere interne componenten, zoals een token endpoint, sleutelbeheer, policykoppelingen en een tokenservice.
Functies:

1 client-identificatie (zie clientidentiteit)
2 client-authenticatie (zie Authenticatiemiddel);
3 validatie van token request parameters;

4 indien van toepassing: verwerking van OAuth 2.0 Rich Authorization Requests[59] - authorization_details;
5 autorisatiebeslissing;
6 creatie token en ondertekening (zie Access Token);
7 intrekking, introspection of statusbeheer indien toegepast.

Clientidentiteit
De clientidentiteit is de technische identiteit waarmee een client bij de Authorization Server bekend is. Deze wordt meestal uitgedrukt met een client_id. De clientidentiteit is niet hetzelfde als de onderwijsorganisatie namens wie de client handelt.

Authenticatiemiddel
Het authenticatiemiddel is het bewijs waarmee de client aantoont de geregistreerde client te zijn. Mogelijke vormen zijn:
1 een gedeeld geheim bij client_secret_basic;
2 een private key bij private_key_jwt;
3 een certificaat en private key;
4 federatief ontdekte en gevalideerde sleutelmetadata.

De gekozen methode bepaalt hoe het bewijs wordt geleverd; de registratie en het beleid bepalen waarom het wordt vertrouwd.

Access Token
Het Access Token is een token voor toegang tot een protected resource. Het token heeft een specifiek doel en mag niet worden geïnterpreteerd als een algemeen bewijs dat een client binnen de gehele keten bevoegd is.

4.5. Clientapplicatie

De clientapplicatie is het bouwblok dat de rol van een OAuth-client realiseert. Het bouwblok beheert de technische clientidentiteit en het authenticatiemiddel, bouwt het token request op en gebruikt het verkregen Access Token voor toegang tot de protected resource.
Functies:
1 veilig beheren van secret of private key;
2 aanvragen en cachen van Access Tokens;
3 doorgeven van authorization_details;
4 behandelen van tokenfouten;
5 voorkomen van ongewenst hergebruik buiten de bedoelde resource.

4.6. Clientregistratie

De clientregistratie is een logisch bouwblok waarin de technische eigenschappen en toegestane configuratie van een client worden vastgelegd. Dit bouwblok maakt expliciet dat er bepaalde gegevens moeten worden vastgelegd. De gegevens worden o.a. gebruikt bij de OAuth-rol Authorization Server. Clientregistratie (onboarding) is altijd nodig, ook wanneer vertrouwen in sleutelmateriaal via PKI wordt ondersteund. PKI kan in andere gevallen de

[59] Zie RFC9396

registratie automatiseren, maar binnen Edukoppeling wordt uitgegaan van een ‘handmatige’ registratie met aanvullende afspraken die buiten de scope van Edukoppeling vallen maar wel onderdeel zijn van de ketensamenwerking.

Organisatie-identiteit
Het bouwblok Organisatie-identiteit ondersteunt de eenduidige identificatie van organisaties die deelnemen aan een ketensamenwerking. Het vormt de basis voor de organisatorische identiteit waarop clientregistratie, machtigingen en autorisatiecontext kunnen aansluiten.

Binnen Edukoppeling wordt aangesloten bij de OIN-nummersystematiek zoals beheerd door Logius. Deze nummersystematiek biedt een uniforme wijze om organisaties en, indien nodig, organisatieonderdelen duurzaam en eenduidig te identificeren. Een OIN identificeert de organisatie en moet worden onderscheiden van de technische identiteit van een OAuth-client (client_id) en van de authenticatiemiddelen waarmee een client zich bij een Authorization Server authenticeert.

De OIN-nummersystematiek kent een hiërarchische opbouw. Het hoofdnummer identificeert de juridische organisatie en vormt de basis voor de identificatie binnen de ketensamenwerking. Vanuit dit hoofdnummer kunnen één of meer OIN's voor organisatieonderdelen of organisatorische eenheden worden afgeleid. Hierdoor kan zowel op organisatieniveau als op het niveau van een organisatieonderdeel worden samengewerkt, terwijl de relatie met de overkoepelende organisatie behouden blijft.

Niet ieder OIN hoeft opgenomen te zijn in het landelijke OIN-register. Het OIN-register bevat uitsluitend de OIN's die door Logius zijn uitgegeven en geregistreerd. Binnen een ketensamenwerking kunnen aanvullende OIN's volgens dezelfde nummersystematiek worden toegepast. Het hoofdnummer vormt daarmee het stabiele anker voor de verificatie van de identiteit van een organisatie. Indien een specifiek OIN niet voorkomt in het OIN-register, kan de identiteit van de organisatie nog steeds worden vastgesteld aan de hand van het hoofdnummer, waarvoor authentieke registraties, zoals het Handelsregister of het Register Instellingen en Opleidingen (RIO), beschikbaar zijn. De ketensamenwerking bepaalt welke authentieke registratie hiervoor wordt gebruikt.
Functies:

1. eenduidig identificeren van ketenpartners met behulp van de OIN-nummersystematiek;
2. herleiden van een OIN tot het bijbehorende hoofdnummer;
3. ondersteunen van verificatie van de organisatorische identiteit (OIN of hoofdnummer) aan de hand van authentieke registraties;
4. leveren van een stabiele organisatie-identiteit voor clientregistratie, machtigingen en autorisatiecontext.

4.7. Contractregister[60]

Contracten worden op één plek geregistreerd door de ketenpartner die de specificatie definieert; elke ketenpartner zal zo’n contract-register aanbieden. Deze plek, een contractregister, is de bron van waarheid voor de betreffende specificatie.

[60] Zie Edukoppeling communicatieprofiel link

Een contractregister maakt het mogelijk om afspraken over schemastructuren, eigenaarschap en kwaliteit te definiëren en centraal op te slaan en versioneren. Het zorgt ervoor dat systemen veilig met elkaar kunnen communiceren door afspraken over berichtenformaten, API-specificaties en events te beheren. Een contractregister maakt dit mogelijk voor synchrone en asynchrone APIs.

Technische contracten
Ketenpartner bepalen altijd gezamenlijk wie welke contracten opstelt. Wel gelden conform het Communicatie-profiel de volgende richtlijnen:

- De OpenAPI-specificatie (OAS) van de API wordt gedefinieerd door de ontvangende partij.
- De AsyncAPI-specificatie (AAS), i.e. de business data, wordt gedefinieerd door de producer, omdat deze eigenaar is van data en het gedrag. Hierbij stemt de producer de specificatie af op de eisen van de mogelijke consumers. Datacontracten hanteren idealiter waar mogelijk sectorstandaarden voor gegevensstructuren.

4.8. Consumer-intermediair

De consumer-intermediair is het externe, beveiligde ontvangstpunt van een ketenpartner voor asynchrone gegevensuitwisseling. De communicatie tussen consumer-intermediair en publisher-intermediair wordt gerealiseerd conform het Edukoppeling Communicatie-profiel en is beveiligd met het Edukoppeling OAuth-profiel. Een intermediair kan, afhankelijk van de interactie, zowel de rol van OAuth Client als die van Resource Server vervullen. Welke OAuth-rol een intermediair vervult, wordt bepaald door de richting van de betreffende interactie. Zo kan de consumer-intermediair een event ontvangen als Resource Server en de verdere routering naar het achterliggende consumerlandschap verzorgen. De inrichting van deze interne levering is implementatie-specifiek en valt buiten scope van Edukoppeling.
Functies:

1. Externe ontvangst
   a. ontvangen van Events;
   b. aanbieden van API conform Communicatie-profiel.
2. Beveiliging
   a. Resource Server;
   b. OAuth-validatie;
   c. tokenvalidatie;
   d. TLS.
3. Validatie
   a. Communicatie-profiel validatie;
   b. schema-validatie;
   c. verplichte metadata.
4. Acceptatie
   a. duurzame opslag;
   b. buffering;
   c. technische acceptatie.
5. Routering
   a. routeren naar interne consumers;

   b. tenantselectie;
   c. filtering.
6. Levering
   a. retry;
   b. replay;
   c. dead-letter;
   d. idempotentie.

4.9. Publisher-intermediair

De publisher-intermediair vormt het externe verzendpunt van een ketenpartner. Het ontvangt gebeurtenissen uit het achterliggende producerlandschap. Betreffende events worden vervolgens conform het Edukoppeling Communicatie-profiel geleverd op basis van geregistreerde subscriptions aan consumer-intermediair van ketenpartner(s). De communicatie tussen beide intermediairs wordt beveiligd met het Edukoppeling OAuth-profiel. De wijze waarop interne producers gebeurtenissen aan het publisher-intermediair aanbieden is implementatie-specifiek en valt buiten scope van Edukoppeling. Welke OAuth-rol een intermediair vervult, wordt bepaald door de richting van de betreffende interactie. Zo kan de publisher-intermediair een subscription aanvraag ontvangen als Resource Server.

Functies[61]:

1. Intern ontvangst
   a. controle van publicatierechten interne publishers.
2. Externe publicatie
   a. publiceren van events naar andere ketenpartners;
   b. realiseren van client voor consumer endpoint.
3. Externe ontvangst
   a. asynchrone response van consumer-intermediair kunnen ontvangen (en kunnen correleren aan extern gepubliceerd event).
4. Beveiliging
   a. OAuth Client richting consumer-intermediair (API);
   b. aanvragen en beheren van Access Tokens;
   c. TLS.
5. Contractvalidatie
   a. validaties conform Communicatie-profiel;
   b. schema-validatie;
   c. verplichte metadata controleren.
6. Routering
   a. bepalen welke consumer- intermediairs een event moeten ontvangen;
   b. toepassen van subscriptionregistraties.
7. Betrouwbare aflevering
   a. retry;
   b. buffering;
   c. foutafhandeling;
   d. correlatie.

[61] Zie Profiel voor communicatie via RESTful APIs

4.10. Machtiging

De API-architectuur onderkent het concept van een machtiging. De machtiging is een Edukoppeling concept en geen internationale standaard of onderdeel van OAuth 2.0 gerelateerde standaarden. Wel wordt Rich Authorization Requests (RFC9396) gebruikt voor transport van gegevens waarmee het bestaan van een machtiging kan worden geverifieerd. Met een machtiging kunnen leveranciers die namens een onderwijsorganisatie in de rol van client en API gegevens uitwisselen dit expliciet vastleggen. De machtiging moet te relateren zijn aan de onderwijsorganisatie-client-API (edu-to is edu-from), of onderwijsorganisatie-client (edu-from) en onderwijsorganisatie-API (edu-to) combinatie.

Een machtigingsregister[62] is een (centraal of decentraal) register waarin wordt vastgelegd welke systemen gegevens uitwisselen, namens welke organisatie en binnen welke context deze gegevens mogen worden uitgewisseld.

OAuth 2.0 Rich Authorization Requests - Authorization_details (RFC9396)

Het authorization_details object wordt binnen RFC9396 gedefinieerd en in de context van het Edukoppeling OAuth-profiel bevat het onder andere het type, edu-from en edu-to.

edu-from
edu-from is de organisatorische autorisatiecontext. De waarde identificeert de organisatie waarvoor of namens wie de client de interactie initieert, zoals vastgelegd in de relevante machtiging. De AS moet de waarde valideren.

edu-to
edu-to identificeert de organisatie aan de serverzijde die de client voor de betreffende interactie heeft gemachtigd. Het attribuut begrenst de organisatorische context waarvoor toegang wordt aangevraagd. De serverzijde kan de goedgekeurde waarde ook gebruiken om te routeren en de juiste datasetcontext te bepalen.

[62] Eerdere versies van de Edukoppeling-architectuur ging uit van een centraal Onderwijs Service Register voor het vastleggen van mandaten. We zien dat afsprakenstelsels nu dergelijke informatie ook decentraal beheren, zie Edu-V consentmanagement https://edu-v.atlassian.net/wiki/spaces/AFSPRAKENS/pages/75268097/Regie+op+gegevens

5. Ketensamenwerking

5.1. Inleiding

Binnen het onderwijs is er een grote diversiteit aan ketens en organisaties die daarbinnen ieder een eigen verantwoordelijkheid hebben. Onderwijsorganisaties zijn verantwoordelijk voor de uitvoering van het onderwijs en de registratie van onderwijsgegevens. DUO vervult wettelijke taken op het gebied van registratie, bekostiging en diplomering. Leveranciers ontwikkelen en beheren administratieve systemen, leeromgevingen en andere digitale voorzieningen. De wijze waarop de ketenpartners samenwerken is aan het veranderen. Voorheen werden informatiesystemen veelal voor een specifiek doel aan elkaar gekoppeld. Gegevensuitwisseling werd vaak beschreven vanuit een eenvoudige relatie tussen een bron en een afnemer. De bron beheert gegevens en stelt deze beschikbaar, terwijl de afnemer deze gebruikt voor de uitvoering van een eigen proces. Hoewel dit voor sommige situaties nog steeds passend is, sluit het minder goed aan bij moderne onderwijsketens. Mede doordat het aantal koppelingen toenam, ontstond een steeds complexer integratielandschap waarin wijzigingen moeilijk door te voeren waren en iedere nieuwe koppeling maatwerk vereiste. We zien nu dat meer en meer onderwijsprocessen uit een netwerk[63] van ketenpartners bestaan waarin gegevens gedurende een proces worden verrijkt, gecombineerd en opnieuw beschikbaar worden gesteld. Ketenpartners gebruiken gegevens van elkaar als onderdeel van hun eigen bedrijfsproces, voegen daar nieuwe informatie aan toe en stellen het resultaat vervolgens weer beschikbaar aan andere ketenpartners. Afhankelijk van de context hebben ketenpartners de rol van gegevensaanbieder[64], gegevensafnemer[65] of beide. Deze gedachte sluit ook aan bij moderne API- en eventgedreven architecturen. De opkomst van API's, cloudplatformen en eventgedreven architecturen heeft geleid tot een andere manier van koppelen. Applicaties worden steeds vaker ontwikkeld als zelfstandige diensten die gegevens beschikbaar stellen via gestandaardiseerde interfaces die breed inzetbaar zijn. Vaak in de vorm van API’s of publicatie van gebeurtenissen. Tegelijkertijd gebruiken zij de API's van andere ketenpartners of consumeren gebeurtenissen.

Deze ontwikkelingen raken verschillende architectuurlagen en een nieuwe Edukoppeling-architectuur moet hierop aansluiten. Er is behoefte aan een generiek afsprakenstelsel voor de technische logistiek, de wijze waarop gesloten data veilig, betrouwbaar en interoperabel tussen systemen van ketenpartners worden getransporteerd.

5.2. Grondslagenlaag

De grondslagenlaag beschrijft de juridische en beleidsmatige kaders waarbinnen organisaties gegevens mogen uitwisselen. Binnen het netwerk van samenwerkende ketenpartners vinden gegevensuitwisselingen plaats in uiteenlopende ketensamenwerkingen. Er gelden hierbij verschillende wettelijke grondslagen en sectorspecifieke afspraken. Bij uitwisseling van persoonsgegevens is de Algemene

[63] Zie ook de ideeën rond data spaces (Design Principles for Data Spaces | Position Paper)
[64] Zie: https://rosa.wikixl.nl/index.php/Begrip:92F9D167-CDE0-4516-8741-AD6A776F3419 ook wel gegevensverstrekker genoemd.
[65] Zie: https://rosa.wikixl.nl/index.php/Begrip:8817b7b1-80a0-4e7e-97c1-75cb2ec66b59. In de context van het OAuth-profiel is dit de API-afnemer.

Verordening Gegevensbescherming (AVG) een belangrijk wettelijk kader. De AVG stelt niet alleen eisen aan de wijze waarop persoonsgegevens worden verwerkt, maar ook aan de verantwoordelijkheden van de betrokken organisaties en de voorwaarden waaronder gegevens mogen worden uitgewisseld. Een belangrijk uitgangspunt van de AVG is dat persoonsgegevens uitsluitend mogen worden verwerkt wanneer daarvoor een geldige grondslag bestaat. Deze grondslag kan bijvoorbeeld voortvloeien uit een wettelijke verplichting of de uitvoering van een overeenkomst. Daarnaast schrijft de AVG voor dat persoonsgegevens alleen mogen worden gebruikt voor een duidelijk omschreven doel en niet voor andere doeleinden dan waarvoor zij zijn verzameld (doelbinding). Ook moeten uitsluitend die persoonsgegevens worden verwerkt die noodzakelijk zijn voor het doel (dataminimalisatie).

5.3. Organisatielaag

De grondslagenlaag stelt kaders voor de organisatielaag. Binnen de organisatielaag worden afspraken gemaakt over de samenwerking tussen organisaties en vormt de basis voor het noodzakelijke vertrouwen in elkaar. Deze afspraken beschrijven onder meer de rollen, verantwoordelijkheden en bevoegdheden van de deelnemende ketenpartners en de voorwaarden waaronder gegevens mogen worden uitgewisseld. Dit zorgt voor het benodigde vertrouwen om überhaupt gegevens met elkaar uit te wisselen. Bij de uitwisseling van persoonsgegevens moeten er tevens maatregelen getroffen worden om de privacy te borgen. Daarnaast worden de kenmerken van de bedrijfsprocessen vastgesteld. Deze proceskarakteristieken bepalen welke vormen van gegevensuitwisseling nodig zijn en vormen daarmee een belangrijk uitgangspunt voor de toe te passen interactiepatronen.

Organisatorisch vertrouwensmodel
Met het voortdurend uitwisselen van gegevens tussen ketenpartners ontstaat een sterke wederzijdse afhankelijkheid. Ketenpartners moeten erop kunnen vertrouwen dat de ontvangen gegevens afkomstig zijn van de juiste partij, dat deze gegevens integer zijn, dat zij onder de juiste voorwaarden worden gebruikt en dat uitsluitend bevoegde ketenpartners toegang krijgen. Dit vertrouwen beperkt zich niet tot de relatie tussen twee voorzieningen, maar strekt zich uit over de gehele keten. Een ketenpartner die gegevens ontvangt, moet niet alleen kunnen vaststellen welke ketenpartner met welke voorziening de gegevens heeft verzonden, maar eventueel ook namens welke onderwijsorganisatie de gegevens worden verwerkt. Even belangrijk is dat de verzendende ketenpartner erop moet kunnen vertrouwen dat de ontvangende ketenpartner uitsluitend toegang krijgt tot de gegevens waarvoor zij bevoegd is en deze alleen voor het afgesproken doel gebruikt. Vertrouwen ontstaat niet door de technische verbinding tussen twee voorzieningen, maar door de afspraken die ketenpartners maken. Deze afspraken worden vastgelegd als onderdeel van een zogenaamde ketensamenwerking. ROSA definieert een ketensamenwerking[66] als een concrete invulling van een scenario, waarbij er sprake is van een vorm van gegevensuitwisseling (interactie) tussen twee of meer ketenpartners ter ondersteuning van het afhandelen een of meerdere ketenprocess(tapp)en. Deze afspraken vormen gezamenlijk het organisatorisch vertrouwensmodel van de keten. Zonder deze organisatorische afspraken kan geen betrouwbare technische gegevensuitwisseling plaatsvinden. Om

[66] https://rosa.wikixl.nl/index.php/Begrip:27a6accf-472d-4415-bc5b-1e9de17bf288

gegevens veilig en gecontroleerd tussen ketenpartners uit te wisselen, moet het organisatorisch vertrouwensmodel worden vertaald naar een technisch vertrouwensmodel binnen de applicatielaag.

Privacy
Binnen iedere ketensamenwerking moet duidelijk zijn welke organisaties persoonsgegevens verwerken, welke rol zij daarbij vervullen en welke verantwoordelijkheden zij dragen. De AVG onderscheidt hierbij onder meer de verwerkingsverantwoordelijke en de verwerker. Wanneer een organisatie of dienstverlener namens een andere organisatie persoonsgegevens verwerkt of uitwisselt, moeten hierover expliciete organisatorische afspraken zijn gemaakt. Onderdeel van deze afspraken kan het gebruik van een machtiging[67] zijn. Hierin wordt vastgelegd namens welke (onderwijs)organisatie een ketenpartner binnen een ketensamenwerking persoonsgegevens uitwisselt met een bepaalde andere ketenpartner. Edukoppeling ondersteunt de technische toepassing van de organisatorische afspraken rond een machtiging. Tijdens de gegevensuitwisseling kan machtigingsinformatie uitgewisseld worden die kan worden gebruikt voor de autorisatiebeslissing.

Proceskarakteristieken
Binnen een ketensamenwerking werken twee of meer organisaties samen aan de uitvoering van één of meerdere ketenproces(stapp)en. Deze samenwerking krijgt vorm door gegevens uit te wisselen tussen de betrokken ketenpartners. Een ketensamenwerking bestaat daarbij doorgaans uit meerdere gegevensuitwisselingen die gezamenlijk bijdragen aan de afhandeling van een processtap. Niet iedere gegevensuitwisseling heeft echter dezelfde karakteristieken of stelt dezelfde eisen aan de onderliggende techniek.

De aard van het bedrijfsproces bepaalt in belangrijke mate hoe gegevens moeten worden uitgewisseld. Sommige processtappen vereisen dat een vervolgactie pas kan worden uitgevoerd nadat direct een antwoord is ontvangen. Deze processen zijn vaak tijdkritisch en vereisen een snelle en betrouwbare respons. Andere processen kunnen over een langere periode worden uitgevoerd, waarbij de verschillende processtappen onafhankelijk van elkaar plaatsvinden.

De variatie in proceskarakteristieken leidt tot verschillende interactiebehoeften. De Edukoppeling-architectuur ondersteunt deze variatie door niet een enkel interactiepatroon te verplichten, maar een kader te bieden waarin per use case een passend interactiepatroon[68] kan worden gekozen. Zo kan bij patronen waarbij directe terugkoppeling noodzakelijk is, een synchrone interactie met een gegevensdienst RESTful API ondersteund worden. Processen waarin gegevens onafhankelijk van elkaar kunnen worden verwerkt lenen zich juist beter voor communicatie middels asynchrone interacties.

[67] https://rosa.wikixl.nl/index.php/Begrip:C70d1782-8382-4f88-bf5f-b13c88fd291b, zie ook mandaat (https://rosa.wikixl.nl/index.php/Begrip:91361b16-35b0-4f8c-8bcb-34d5ae82db3f) en consent (https://edu-v.atlassian.net/wiki/spaces/AFSPRAKENS/pages/75268097/Regie+op+gegevens)
[68] Zie Transactiepatronen

5.4. Informatielaag

Om een zinvolle gegevensuitwisseling te realiseren moeten de betrokken ketenpartners informatie op dezelfde wijze interpreteren. Organisaties moeten daarom niet alleen overeenstemming hebben over de technische wijze waarop gegevens worden uitgewisseld, maar ook over de betekenis van de gegevens, de gebruikte identificerende kenmerken en de context waarin de gegevens worden gebruikt. Deze semantische interoperabiliteit wordt beschreven binnen de informatielaag van het Edustandaard-lagenmodel. Hierbinnen wordt de betekenis van gegevens, gebeurtenissen en bedrijfsobjecten vastgesteld.

Hoewel Edukoppeling geen uitspraken doet over de inhoud van gegevens, maakt de architectuur wel gebruik van gegevens uit de informatielaag om de gegevensuitwisseling correct uit te kunnen voeren. Het moet tenslotte bekend zijn welke ketenpartners betrokken zijn, namens welke organisatie wordt gehandeld en op welke organisatorische context (ketensamenwerking bijvoorbeeld) een uitwisseling betrekking heeft. Deze informatie is noodzakelijk om identificatie, authenticatie en autorisatie op een consistente wijze te kunnen uitvoeren. Door deze informatie expliciet ook in een logistieke context te plaatsen blijft Edukoppeling generiek toepasbaar voor uiteenlopende ketensamenwerkingen en kan de gegevensuitwisseling veilig, betrouwbaar en (semantisch) interoperabel worden uitgevoerd.

5.5. Applicatielaag

De applicatielaag vertaalt de kaders uit de andere architectuurlagen naar een technische inrichting voor gegevensuitwisseling. Zo wordt het organisatorische vertrouwensmodel vertaald naar een technisch vertrouwensmodel waarin wordt beschreven hoe systemen elkaar identificeren en authenticeren, hoe autorisatiebeslissingen worden genomen en hoe de organisatorische context (middels gebruik van een machtiging) van een gegevensuitwisseling wordt betrokken bij de toegang tot API’s. Edukoppeling schrijft het gebruik van aantal standaarden en profielen voor rond identificatie, authenticatie, autorisatie en beveiligde gegevensuitwisseling in communicatie met API’s (gegevensdienst of event-voorziening). Met Edukoppeling wordt binnen de applicatielaag een API-architectuur geboden waarin een resource-georiënteerde en een event-georiënteerde architectuur elkaar aanvullen. Beide architectuurstijlen vervullen een eigen rol binnen de gegevensuitwisseling en worden toegepast afhankelijk van de kenmerken van de ketensamenwerking en de onderliggende bedrijfsprocessen.
Een resource-georiënteerde architectuur vormt het uitgangspunt voor het beschikbaar stellen van gegevensdiensten. RESTful API's maken het mogelijk om gegevens op te vragen, opdrachten uit te voeren en direct terugkoppeling te ontvangen. Deze architectuurstijl sluit goed aan bij processen waarbij de voortgang afhankelijk is van een onmiddellijke reactie, zoals het opvragen van actuele gegevens.
Een event-georiënteerde architectuur (EDA) is met name geschikt voor processen waarbij organisaties onafhankelijk van elkaar moeten kunnen reageren op wijzigingen, waarbij meerdere afnemers dezelfde gebeurtenis kunnen verwerken of waarbij de verwerking langere tijd kan duren. In plaats van gegevens periodiek op te vragen bij gegevensdiensten, publiceren organisaties events via een eventvoorziening (publisher-intermediair). Andere ketenpartners kunnen zich abonneren[69] en deze events ontvangen en vervolgens op een

[69] De ketensamenwerking bepaalt gezamenlijk hoe het proces van abonneren plaatsvindt. Zie voor meer details het Communicatie-profiel

zelfgekozen moment verwerken via hun eventvoorziening (consumer-intermediair). Het biedt een meer losse koppeling tussen gegevensaanbieder en gegevensafnemer. Edukoppeling beschouwt deze architectuurstijlen nadrukkelijk niet als concurrerende alternatieven. Afhankelijk van de proceskarakteristieken kunnen synchrone API-interacties en asynchrone events daarom binnen één ketensamenwerking naast elkaar worden toegepast. Zo kan een gebeurtenis een wijziging signaleren, waarna één of meerdere ketenpartners via een API de actuele gegevens ophalen om hun eigen processen verder uit te voeren. We zien wel dat EDA in ketensamenwerkingen meer en meer de voorkeur heeft omdat deze beter aansluit bij de dynamiek, schaal en ontkoppelingsbehoefte van moderne gegevensuitwisseling. EDA vraagt wel om expliciete afspraken over semantiek, governance en beveiliging, omdat context en autorisatie niet impliciet via een directe interactie worden afgedwongen. EDA biedt samen met het Edukoppeling Profiel voor communicatie via API's[70] (hierna Communicatie-profiel) een robuust en toekomstbestendig fundament voor gegevensuitwisseling.

De volgende hoofdstukken werken deze applicatielaag verder uit. Allereerst worden de verschillende transactiepatronen beschreven waarmee de proceskarakteristieken van een ketensamenwerking worden vertaald naar generieke vormen van gegevensuitwisseling. Vervolgens wordt in de API-architectuur beschreven hoe deze transactiepatronen worden gerealiseerd met RESTful API's, eventgedreven communicatie en de daarbij behorende bouwblokken, beveiligingsmechanismen en standaarden.

[70] Link

6. Transactiepatronen

6.1. Inleiding

Een ketensamenwerking tussen ketenpartners bestaat uit één of meer technische interacties. Voor iedere interactie wordt onderscheid gemaakt tussen de partij die de technische interactie initieert en de partij die de interactie ontvangt. Op technisch niveau wordt hiervoor het client-servermodel gehanteerd. De Client initieert een request naar een Server die een daarvoor beschikbare interface aanbiedt. Deze technische rollen moeten niet worden verward met de functionele rollen[71]. Afhankelijk van de interactie kan zowel de gegevensaanbieder als de gegevensafnemer de rol van Client vervullen. Het client-servermodel beschrijft daarmee de technische communicatie, maar schrijft nog niet voor of de interactie bijvoorbeeld push of pull, synchroon of asynchroon is. Deze eigenschappen worden afzonderlijk bepaald aan de hand van de kenmerken van een interactie.

Figuur 4 - Client - server communicatie

De client – server communicatie is een simplificatie van de IT-landschappen in praktijk. End-to-end communicatie tussen client en server kan door verschillende componenten, infrastructureel en/of applicatief, heengaan. Een gegevensuitwisseling tussen ketenpartners vindt in de praktijk dan ook plaats binnen een breder applicatie- en proceslandschap. Een bedrijfsproces kan verschillende applicaties, integratievoorzieningen, intermediairs en technische interfaces omvatten. Een end-to-end uitwisseling raakt daarmee meerdere architectuurlagen[72]. Van de aanleiding en verantwoordelijkheden binnen de organisatie-

[71] Zie gegevensaanbieder en gegevensafnemer in hoofdstuk API-architectuur
[72] Zie de architectuurlagen in het hoofdstuk Bouwblokken
De architectuur gaat uit van een aantal bouwblokken die gezamenlijk een veilige en beheersbare gegevensuitwisseling binnen een ketensamenwerking realiseren.
API
Een API vormt het gestandaardiseerde toegangspunt waarmee een ketenpartner gegevens of functionaliteit beschikbaar stelt aan andere ketenpartners. De API abstraheert de achterliggende implementatie en biedt een stabiel technisch contract waartegen andere applicaties kunnen integreren. Afhankelijk van het gekozen interactiepatroon realiseert een API een gegevensdienst of een eventvoorziening.

Gegevensdienst
Bij synchrone gegevensuitwisseling realiseert de API een gegevensdienst in de vorm van een RESTful API. De API stelt één of meer resources of functies beschikbaar via gestandaardiseerde HTTP-endpoints. De API verwerkt een verzoek direct of retourneert een technische respons wanneer de inhoudelijke verwerking op een later moment plaatsvindt.

Eventvoorziening
Een API wordt bij asynchrone gegevensuitwisseling gerealiseerd door een eventvoorziening. Deze heeft twee vormen, een publisher-intermediair aan de

verzendende zijde en een consumer-intermediair aan de ontvangende zijde. De intermediairs vormen het gestandaardiseerde en beveiligde koppelvlak tussen de vertrouwensdomeinen van de ketenpartners. Zij bieden, afhankelijk van hun rol binnen een interactie, concrete API's aan voor eventlevering, subscriptionbeheer en eventuele operationele beheerfuncties.

Resource Server
Dit bouwblok is een OAuth-rol. De gegevensdienst of eventvoorziening vervult voor een concrete interactie de rol van Resource Server wanneer deze een beveiligde dienst of functionaliteit beschikbaar stelt waarvoor een geldig Access Token vereist is. De Resource Server valideert het Access Token en beoordeelt of de gevraagde interactie binnen de verleende autorisatie valt. Pas nadat deze controles succesvol zijn uitgevoerd, wordt toegang tot de beschermde gegevens of functionaliteit verleend. De Resource Server realiseert daarmee de technische handhaving van het autorisatiebeleid bij de feitelijke gegevensuitwisseling.

De uitgifte van een Access Token door de Authorization Server betekent niet dat iedere aanvraag automatisch moet worden toegestaan. De Resource Server neemt voor iedere interactie de uiteindelijke toegangsbeslissing. Daarbij kan hij, naast de informatie in het Access Token, aanvullende controles uitvoeren op de opgevraagde resource, de gevraagde handeling, de organisatorische context en eventuele actuele beleidsregels.
Functies:
beschikbaar stellen en beschermen van één of meer protected resources;
valideren van Access Tokens;
afdwingen van scopes, acties en authorization_details;
toepassen van aanvullende autorisatie op basis van onder meer edu-from en edu-to;
routeren: selecteren van de juiste tenant of organisatorische context;
uitvoeren van de gevraagde API-operatie, accepteren van een event of verwerken van een subscriptionverzoek;
retourneren van een passende technische response of ontvangstbevestiging;

Protected resource
Een protected resource is functionaliteit of een gegevensbron waartoe alleen toegang kan worden verkregen met een geldig en passend Access Token. De Resource Server beoordeelt bij iedere interactie of de gevraagde toegang mag worden verleend. Een protected resource is daarmee de concrete functionaliteit van een API die door een Resource Server wordt beschermd. Een API kan één of meerdere protected resources bevatten.

Bij een gegevensdienst, een RESTful API, bestaat een protected resource uit één of meer API-endpoints of resources waarop een client gegevens kan raadplegen of wijzigen.

Bij een eventvoorziening worden de protected resources ontsloten via één of meer API's die door een publisher- of consumer-intermediair worden aangeboden. Afhankelijk van de functie van de intermediair kan dit onder meer een Publication API, Subscription API of Management API zijn. Iedere API kan één of meerdere protected resources beschikbaar stellen waarvoor een geldig Access Token vereist is. Voorbeelden van protected resources in de context van eventvoorzieningen zijn:
het publiceren van een gebeurtenis;
het ontvangen van een gebeurtenis;
het beheren van subscriptions;
het beheren van afleverpunten;
het raadplegen van contractinformatie;

het uitvoeren van beheer- of monitoringsfuncties.

Authorization Server
De Authorization Server realiseert de gelijknamige OAuth-rol. Edukoppeling schrijft het OAuth 2.0 Client Credentials profiel voor en hierbij authenticeert de Authorization Server client en verstrekt tokens waarmee toegang tot API's kan worden verkregen. De Authorization Server vormt daarmee een belangrijk onderdeel van het technisch vertrouwensmodel binnen de ketensamenwerking. Tijdens onboarding worden clients, publieke sleutels, scopes, eventuele authorization_details en vertrouwensrelaties geregistreerd. Een implementatie kan het bestaan uit meerdere interne componenten, zoals een token endpoint, sleutelbeheer, policykoppelingen en een tokenservice.
Functies:
client-identificatie (zie clientidentiteit)
client-authenticatie (zie Authenticatiemiddel);
validatie van token request parameters;
indien van toepassing: verwerking van OAuth 2.0 Rich Authorization Requests - authorization_details;
autorisatiebeslissing;
creatie token en ondertekening (zie Access Token);
intrekking, introspection of statusbeheer indien toegepast.

Clientidentiteit
De clientidentiteit is de technische identiteit waarmee een client bij de Authorization Server bekend is. Deze wordt meestal uitgedrukt met een client_id. De clientidentiteit is niet hetzelfde als de onderwijsorganisatie namens wie de client handelt.

Authenticatiemiddel
Het authenticatiemiddel is het bewijs waarmee de client aantoont de geregistreerde client te zijn. Mogelijke vormen zijn:
een gedeeld geheim bij client_secret_basic;
een private key bij private_key_jwt;
een certificaat en private key;
federatief ontdekte en gevalideerde sleutelmetadata.

De gekozen methode bepaalt hoe het bewijs wordt geleverd; de registratie en het beleid bepalen waarom het wordt vertrouwd.

Access Token
Het Access Token is een token voor toegang tot een protected resource. Het token heeft een specifiek doel en mag niet worden geïnterpreteerd als een algemeen bewijs dat een client binnen de gehele keten bevoegd is.

Clientapplicatie
De clientapplicatie is het bouwblok dat de rol van een OAuth-client realiseert. Het bouwblok beheert de technische clientidentiteit en het authenticatiemiddel, bouwt het token request op en gebruikt het verkregen Access Token voor toegang tot de protected resource.
Functies:
veilig beheren van secret of private key;
aanvragen en cachen van Access Tokens;
doorgeven van authorization_details;
behandelen van tokenfouten;
voorkomen van ongewenst hergebruik buiten de bedoelde resource.

Clientregistratie
De clientregistratie is een logisch bouwblok waarin de technische eigenschappen en toegestane configuratie van een client worden vastgelegd. Dit bouwblok maakt expliciet dat er bepaalde gegevens moeten worden vastgelegd. De gegevens worden o.a. gebruikt bij de OAuth-rol Authorization Server. Clientregistratie (onboarding) is altijd nodig, ook wanneer vertrouwen in sleutelmateriaal via PKI wordt ondersteund. PKI kan in andere gevallen de registratie automatiseren, maar binnen Edukoppeling wordt uitgegaan van een 'handmatige' registratie met aanvullende afspraken die buiten de scope van Edukoppeling vallen maar wel onderdeel zijn van de ketensamenwerking.

Organisatie-identiteit
Het bouwblok Organisatie-identiteit ondersteunt de eenduidige identificatie van organisaties die deelnemen aan een ketensamenwerking. Het vormt de basis voor de organisatorische identiteit waarop clientregistratie, machtigingen en autorisatiecontext kunnen aansluiten.

Binnen Edukoppeling wordt aangesloten bij de OIN-nummersystematiek zoals beheerd door Logius. Deze nummersystematiek biedt een uniforme wijze om organisaties en, indien nodig, organisatieonderdelen duurzaam en eenduidig te identificeren. Een OIN identificeert de organisatie en moet worden onderscheiden van de technische identiteit van een OAuth-client (client_id) en van de authenticatiemiddelen waarmee een client zich bij een Authorization Server authenticeert.

De OIN-nummersystematiek kent een hiërarchische opbouw. Het hoofdnummer identificeert de juridische organisatie en vormt de basis voor de identificatie binnen de ketensamenwerking. Vanuit dit hoofdnummer kunnen één of meer OIN's voor organisatieonderdelen of organisatorische eenheden worden afgeleid. Hierdoor kan zowel op organisatieniveau als op het niveau van een organisatieonderdeel worden samengewerkt, terwijl de relatie met de overkoepelende organisatie behouden blijft.

Niet ieder OIN hoeft opgenomen te zijn in het landelijke OIN-register. Het OIN-register bevat uitsluitend de OIN's die door Logius zijn uitgegeven en geregistreerd. Binnen een ketensamenwerking kunnen aanvullende OIN's volgens dezelfde nummersystematiek worden toegepast. Het hoofdnummer vormt daarmee het stabiele anker voor de verificatie van de identiteit van een organisatie. Indien een specifiek OIN niet voorkomt in het OIN-register, kan de identiteit van de organisatie nog steeds worden vastgesteld aan de hand van het hoofdnummer, waarvoor authentieke registraties, zoals het Handelsregister of het Register Instellingen en Opleidingen (RIO), beschikbaar zijn. De ketensamenwerking bepaalt welke authentieke registratie hiervoor wordt gebruikt.
Functies:
eenduidig identificeren van ketenpartners met behulp van de OIN-nummersystematiek;
herleiden van een OIN tot het bijbehorende hoofdnummer;
ondersteunen van verificatie van de organisatorische identiteit (OIN of hoofdnummer) aan de hand van authentieke registraties;
leveren van een stabiele organisatie-identiteit voor clientregistratie, machtigingen en autorisatiecontext.

Contracten worden op één plek geregistreerd door de ketenpartner die de specificatie definieert; elke ketenpartner zal zo'n contract-register aanbieden. Deze plek, een contractregister, is de bron van waarheid voor de betreffende specificatie.

Een contractregister maakt het mogelijk om afspraken over schemastructuren, eigenaarschap en kwaliteit te definiëren en centraal op te slaan en versioneren. Het zorgt ervoor dat systemen veilig met elkaar kunnen communiceren door afspraken over berichtenformaten, API-specificaties en events te beheren. Een contractregister maakt dit mogelijk voor synchrone en asynchrone APIs.

Technische contracten
Ketenpartner bepalen altijd gezamenlijk wie welke contracten opstelt. Wel gelden conform het Communicatie-profiel de volgende richtlijnen:
De OpenAPI-specificatie (OAS) van de API wordt gedefinieerd door de ontvangende partij.
De AsyncAPI-specificatie (AAS), i.e. de business data, wordt gedefinieerd door de producer, omdat deze eigenaar is van data en het gedrag. Hierbij stemt de producer de specificatie af op de eisen van de mogelijke consumers. Datacontracten hanteren idealiter waar mogelijk sectorstandaarden voor gegevensstructuren.

Consumer-intermediair
De consumer-intermediair is het externe, beveiligde ontvangstpunt van een ketenpartner voor asynchrone gegevensuitwisseling. De communicatie tussen consumer-intermediair en publisher-intermediair wordt gerealiseerd conform het Edukoppeling Communicatie-profiel en is beveiligd met het Edukoppeling OAuth-profiel. Een intermediair kan, afhankelijk van de interactie, zowel de rol van OAuth Client als die van Resource Server vervullen. Welke OAuth-rol een intermediair vervult, wordt bepaald door de richting van de betreffende interactie. Zo kan de consumer-intermediair een event ontvangen als Resource Server en de verdere routering naar het achterliggende consumerlandschap verzorgen. De inrichting van deze interne levering is implementatie-specifiek en valt buiten scope van Edukoppeling.
Functies:
Externe ontvangst
ontvangen van Events;
aanbieden van API conform Communicatie-profiel.
Beveiliging
Resource Server;
OAuth-validatie;
tokenvalidatie;
TLS.
Validatie
Communicatie-profiel validatie;
schema-validatie;
verplichte metadata.
Acceptatie
duurzame opslag;
buffering;
technische acceptatie.
Routering
routeren naar interne consumers;
tenantselectie;
filtering.
Levering
retry;

replay;
dead-letter;
idempotentie.

Publisher-intermediair
De publisher-intermediair vormt het externe verzendpunt van een ketenpartner. Het ontvangt gebeurtenissen uit het achterliggende producerlandschap. Betreffende events worden vervolgens conform het Edukoppeling Communicatie-profiel geleverd op basis van geregistreerde subscriptions aan consumer-intermediair van ketenpartner(s). De communicatie tussen beide intermediairs wordt beveiligd met het Edukoppeling OAuth-profiel. De wijze waarop interne producers gebeurtenissen aan het publisher-intermediair aanbieden is implementatie-specifiek en valt buiten scope van Edukoppeling. Welke OAuth-rol een intermediair vervult, wordt bepaald door de richting van de betreffende interactie. Zo kan de publisher-intermediair een subscription aanvraag ontvangen als Resource Server.

Functies:
Intern ontvangst
controle van publicatierechten interne publishers.
Externe publicatie
publiceren van events naar andere ketenpartners;
realiseren van client voor consumer endpoint.
Externe ontvangst
asynchrone response van consumer-intermediair kunnen ontvangen (en kunnen correleren aan extern gepubliceerd event).
Beveiliging
OAuth Client richting consumer-intermediair (API);
aanvragen en beheren van Access Tokens;
TLS.
Contractvalidatie
validaties conform Communicatie-profiel;
schema-validatie;
verplichte metadata controleren.
Routering
bepalen welke consumer- intermediairs een event moeten ontvangen;
toepassen van subscriptionregistraties.
Betrouwbare aflevering
retry;
buffering;
foutafhandeling;
correlatie.

Machtiging
De API-architectuur onderkent het concept van een machtiging. De machtiging is een Edukoppeling concept en geen internationale standaard of onderdeel van OAuth 2.0 gerelateerde standaarden. Wel wordt Rich Authorization Requests (RFC9396) gebruikt voor transport van gegevens waarmee het bestaan van een machtiging kan worden geverifieerd. Met een machtiging kunnen leveranciers die namens een onderwijsorganisatie in de rol van client en API gegevens uitwisselen dit expliciet vastleggen. De machtiging moet te relateren zijn aan de onderwijsorganisatie-client-API (edu-to is edu-from), of onderwijsorganisatie-client (edu-from) en onderwijsorganisatie-API (edu-to) combinatie.

Een machtigingsregister is een (centraal of decentraal) register waarin wordt vastgelegd welke systemen gegevens uitwisselen, namens welke organisatie en binnen welke context deze gegevens mogen worden uitgewisseld.

context en de betekenis en structuur van de uitgewisselde gegevens in de informatielaag, naar de applicaties en technische voorzieningen waarmee de gegevens daadwerkelijk worden uitgewisseld. De Edukoppeling architectuur beschrijft niet dit volledige end-to-end landschap. Binnen Edukoppeling richten we ons op de gestandaardiseerde technische interactie tussen voorzieningen van ketenpartners. Om deze interactie onafhankelijk van de inrichting van de achterliggende applicatielandschappen te kunnen beschrijven, wordt als technisch basismodel uitgegaan van client-servercommunicatie. Een client initieert daarbij een technische interactie met een server die een interface aanbiedt.

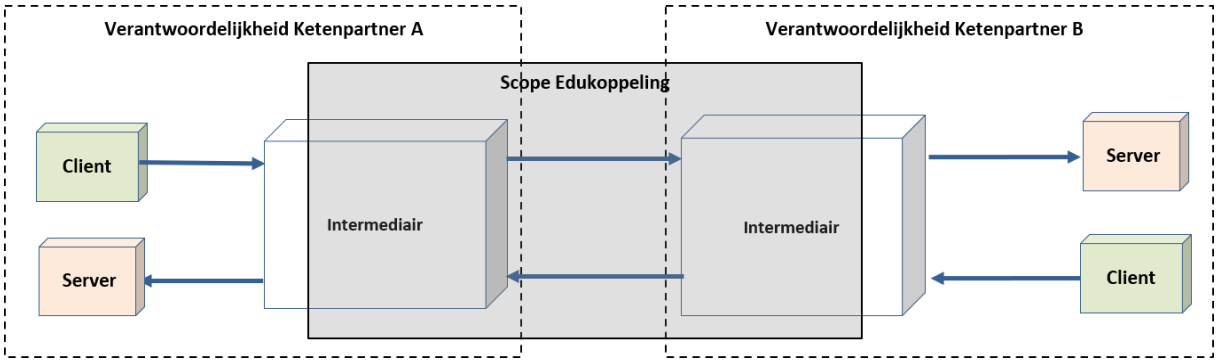
Toepassing binnen Edukoppeling
De Edukoppeling architectuur ondersteunt zowel een RESTful als een EDA architectuurstijl. We gaan voor beide uit van een technische overdracht van gegevens tussen ketenpartners via het client – server basismodel. Bij RESTful onderkennen we een enkelvoudige client en een gateway als intermediair met meerdere servers in het achterlandschap. Bij EDA onderkennen we een achterliggende producer en consumer. Die communiceren daarbij niet rechtstreeks met elkaar. De ketenpartners gebruiken een publisher-intermediair en consumer-intermediair voor de gestandaardiseerde uitwisseling van gebeurtenissen. Bij het afleveren van een gebeurtenis initieert het publisher-intermediair de technische interactie en vervult het daarmee de rol van Client. Het consumer-intermediair biedt de ontvangende API en vervult voor deze interactie de rol van Server. Kortom, het gestandaardiseerde perspectief van een Edukoppeling-interactie vindt plaats tussen beide intermediairs. Dit zorgt ervoor dat elke ketenpartner deze intermediar naar eigen inzicht kan inrichten en beheren, zonder dat het achterliggende IT-landschap van een ketenpartner wordt beïnvloed. Een ketenpartner bepaalt dus zelf hoe de interactie van eigen intermediay naar Client/Server plaatsvindt en hoe de scheiding van rollen daarin plaatsvindt. E.e.a. is schematisch weergegeven in Figuur 5.

OAuth 2.0 Rich Authorization Requests - Authorization_details (RFC9396)

Het authorization_details object wordt binnen RFC9396 gedefinieerd en in de context van het Edukoppeling OAuth-profiel bevat het onder andere het type, edu-from en edu-to.

<u>edu-from</u>
edu-from is de organisatorische autorisatiecontext. De waarde identificeert de organisatie waarvoor of namens wie de client de interactie initieert, zoals vastgelegd in de relevante machtiging. De AS moet de waarde valideren.

<u>edu-to</u>
edu-to identificeert de organisatie aan de serverzijde die de client voor de betreffende interactie heeft gemachtigd. Het attribuut begrenst de organisatorische context waarvoor toegang wordt aangevraagd. De serverzijde kan de goedgekeurde waarde ook gebruiken om te routeren en de juiste datasetcontext te bepalen.



Figuur 5 -Intermediar per ketenpartner

Architectuurstijl en de technische interactie zijn afzonderlijke ontwerpkeuzes. Een ontvangende partij definieert een API-endpoint waar berichten naar verstuurd kunnen worden. Het uitgangspunt voor dit API-endpoint is technologie agnostisch; losstaand van ketenpartner specifieke technologie keuzes voor implementatie van een client/ server. Wel is binnen Edukoppeling losse koppeling een belangrijk uitgangspunt. Dit betekent niet dat alle interacties volledig ontkoppeld moeten zijn. Er kunnen situaties zijn waar synchrone uitwisseling wenselijk of zelfs noodzakelijk is. Het is context- en procesafhankelijk welke vorm van uitwisseling nodig is. Edukoppeling stelt dat de keuze in eerste instantie context- en procesafhankelijk gemaakt dient te worden, maar stelt ook dat indien bij een context/proces óók asynchrone uitwisseling mogelijk is dat bij voorkeur een asynchrone uitwisseling wordt gerealiseerd. We sturen daarmee wel op asynchrone uitwisseling (binnen karakteristieken van context en proces), maar sluit het gebruik van synchrone uitwisseling nadrukkelijk niet uit. Ketenpartners kunnen bewust kiezen om een synchrone/ hechte koppeling te accepteren.

6.2. Kenmerken van een interactie

6.2.1. Richting (eenzijdig / tweezijdig)

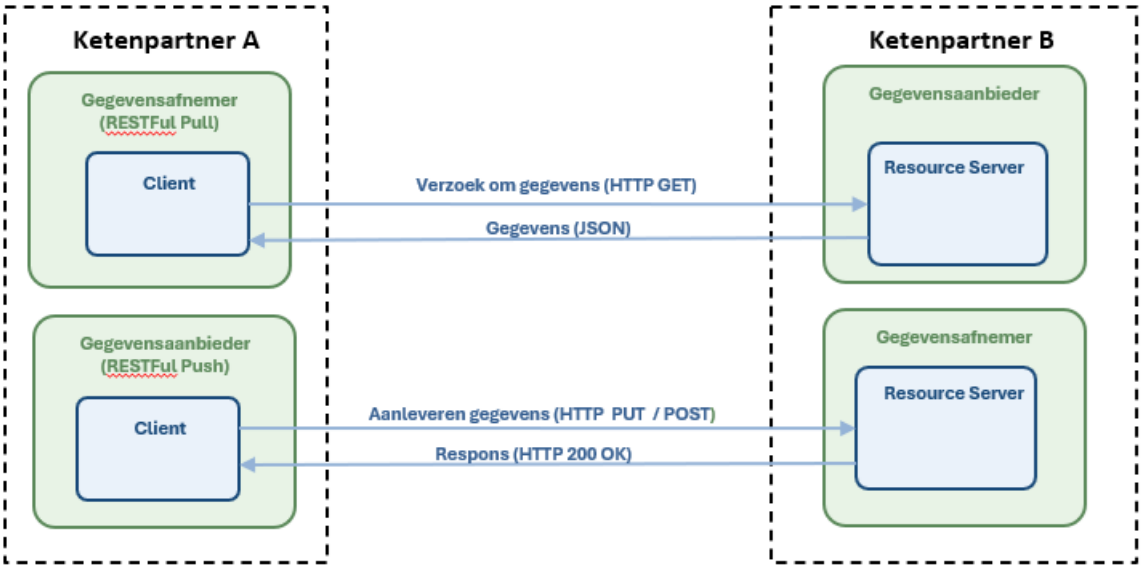
Het kenmerk richting maakt onderscheid tussen eenzijdige of tweezijdige communicatie. Met het perspectief richting geven we (op functioneel niveau) aan of een transactie eenzijdig of tweezijdig is. Bij een tweezijdige interactie heeft een inhoudelijke vraag bijvoorbeeld een inhoudelijk antwoord nodig voor het vervolg van de processtap[73]. De interactie heeft daarmee een vraag-en-antwoordkarakter.

- Bij een eenzijdige transactie communiceert een ketenpartner een gebeurtenis, melding of wijziging zonder dat een inhoudelijke reactie nodig is om het proces voort te zetten. De verzender is functioneel niet afhankelijk van de verwerking door de ontvanger.
- Bij een tweezijdige transactie verwacht de initiërende ketenpartner wel een inhoudelijk antwoord of resultaat dat noodzakelijk is voor de voortgang van het proces.

[73] Binnen Digikoppeling wordt in sommige situaties de aanwezigheid van een technisch responsebericht impliciet geïnterpreteerd als tweezijdige interactie. Edu-V kijkt nadrukkelijker naar het functionele karakter van de transactie en beschouwt meldingen daarom als eenzijdig, ook wanneer technisch een request-response mechanisme wordt gebruikt. Binnen Edukoppeling maken we daarom een duidelijk onderscheid tussen de technische interactie en het functionele procesmatige perspectief.

6.2.2. Overdrachtswijze (push / pull)

Het kenmerk overdrachtswijze beschrijft vanuit welke rol een specifieke gegevensinteractie wordt geïnitieerd. Dit is onafhankelijk van de technische vorm[74] waarin deze wordt gerealiseerd. Hierbij worden de rollen gegevensaanbieder[75] en gegevensafnemer[76] onderscheiden. De gegevensaanbieder stelt de gegevens of gebeurtenissen beschikbaar; de gegevensafnemer gebruikt deze binnen de eigen verwerking.
Deze rollen gelden voor de betreffende gegevensinteractie en staan los van de vraag welke ketenpartner binnen de ketensamenwerking bronhouder of beheerder van de gegevens is. Een ketenpartner kan daardoor in verschillende gegevensinteracties zowel de rol van gegevensaanbieder als die van gegevensafnemer vervullen. Wanneer de gegevensaanbieder de overdracht naar de gegevensafnemer initieert, spreken we van push. Wanneer de gegevensafnemer de interactie initieert om gegevens van de gegevensaanbieder te verkrijgen, spreken we van pull. Push en pull beschrijven daarmee de wijze waarop een afzonderlijke gegevensinteractie wordt geïnitieerd. Zij zeggen op zichzelf niets over de gekozen architectuurstijl, zoals RESTful of eventgedreven communicatie, en moeten worden onderscheiden van het initiatief voor een bovenliggende bedrijfs- of ketentransactie. Een dergelijke transactie kan uit meerdere gegevensinteracties bestaan en daarbij zowel push- als pull-interacties bevatten.



Figuur 6 - OAuth en gegevensafnemer en -aanbieder rollen bij RESTful pull en push uitwisseling

6.2.3. Interactie (timing)

Met het kenmerk interactie belichten we de technische wijze van communicatie (communicatieprotocol) tussen computersystemen en in het bijzonder de afhankelijkheid in tijd tussen verzoek en reactie. We maken hierbij onderscheid in een synchrone en asynchrone interactie. Technisch gezien is het transport afgehandeld wanneer het

[74] HTPP request en response (RFC 9110: HTTP Semantics | RFC Editor)
[75] Zie: https://rosa.wikixl.nl/index.php/Begrip:92F9D167-CDE0-4516-8741-AD6A776F3419 ook wel gegevensverstrekker genoemd.
[76] Zie: https://rosa.wikixl.nl/index.php/Begrip:8817b7b1-80a0-4e7e-97c1-75cb2ec66b59. In de context van het OAuth-profiel is dit de API-afnemer.

communicatieprotocol een ontvangstbevestiging heeft verstuurd. We bedoelen hierbij dus niet de verwerking van gegevens. Een melding kan bijvoorbeeld zowel synchroon als asynchroon worden verzonden. De keuze wordt bepaald door de aard van het proces. De keuze voor synchrone of asynchrone interactie staat in principe los van de gekozen architectuurstijl. Zowel binnen een REST-georiënteerde als een Event Driven Architectuur kunnen synchrone en asynchrone interacties voorkomen. De kenmerken kunnen daarmee onafhankelijk worden gecombineerd. Sommige combinaties liggen vanuit de eigenschappen van een architectuurstijl echter meer voor de hand dan andere.

- Synchroon: Bij een synchrone interactie wacht de initiërende ketenpartner binnen dezelfde interactie op een response van de ontvangende ketenpartner. De response wordt direct via dezelfde verbinding en binnen hetzelfde communicatieprotocol teruggegeven. Dit wordt ook wel een request-response-interactie genoemd. De beschikbaarheid en responstijd van de ontvangende voorziening zijn direct van invloed op het verloop van de interactie. Technisch is aan het wachten op een response een time-out verbonden. Het initiërende systeem moet kunnen omgaan met het uitblijven van een response en bepalen hoe de interactie wordt vervolgd, bijvoorbeeld door het request opnieuw aan te bieden of de fout aan het bovenliggende proces terug te melden.

Timing	• Het patroon is synchroon in tijd.
Voordelen	• Eenvoud en voorspelbaarheid van de communicatie. • De autorisatie kan direct worden toegepast op het moment van het request
Nadelen	• De client moet wachten op een antwoord • Over het algemeen een hechte koppeling tussen gegevensafnemer en gegevensaanbieder.
Wanneer toepassen	Er kunnen zowel gegevens vanuit het request geleverd worden (push-melding) als vanuit de response (pull-bevraging).

- Asynchroon: Bij een asynchrone interactie hoeft de initiërende ketenpartner niet binnen dezelfde interactie te wachten op voltooiing van de verwerking door de ontvangende ketenpartner. De ontvangst van een bericht en de inhoudelijke verwerking daarvan zijn van elkaar gescheiden. Een eventuele inhoudelijke response of bevestiging vindt plaats in een afzonderlijke interactie. Dit vermindert de afhankelijkheid in tijd en ondersteunt lossere koppeling en schaalbaarheid. Het initiërende systeem moet rekening houden met het ontbreken van een directe inhoudelijke response. Daarmee wordt het belangrijk om hierbij ook afspraken te maken over bijvoorbeeld aflevering, bevestiging, retries, duplicaten en foutafhandeling..

Timing	• Het patroon is asynchroon in tijd.
Voordelen	• De mate van koppeling is minder hecht dan bij een synchrone interactie. • Draagt bij aan schaalbaarheid en robuustheid van infrastructuur • Verbeterde interoperabiliteit - lossere koppeling tussen systemen
Nadelen	• Introduceert extra complexiteit. Zo zijn aanvullende maatregelen nodig om autorisatie en authenticatie over meerdere stappen te waarborgen. Naast een client en server is er ook sprake van 1 of 2 intermediairs. Deze complexiteit moet beheersbaar worden gehouden door duidelijke architectuurafspraken en standaardisatie (rationale CloudEvents profiel) • Directe validatie in de ontvanger is vaak een ingewikkelder proces

Wanneer toepassen	Het wordt toegepast in situaties waarin computersystemen onafhankelijk moeten kunnen functioneren. Het is een schaalbare oplossing die goed piekbelastingen kan opvangen. Asynchrone interactie kan op verschillende manieren worden gerealiseerd, bijvoorbeeld via[77]: • callbacks of webhooks; • messaging-systeem. Kaders voor asynchrone interacties worden binnen de Edukoppeling-architectuur beschreven in het Edukoppeling Communicatie-profiel[78].

6.2.4. Koppeling (hecht / los)

De mate waarin computersystemen van elkaar afhankelijk zijn wordt aangeduid als de mate van koppeling. Dit is dus breder dan het kenmerk 'interactie' wat met name de mate van afhankelijkheid in de context van tijd typeert. De mate van koppeling is breder (architectuur) en heeft betrekking op de mate van kennis van elkaars systemen men moet hebben, vaak in de vorm van contracten[79]. Ook de mate van beschikbaarheid en openheid en onafhankelijke implementeerbaarheid spelen hierbij een rol. Zonder al te diep in deze zaken te treden kunnen wel het volgende stellen:

- Bij hechte koppeling zijn computersystemen sterk afhankelijk van elkaar, zowel in termen van beschikbaarheid als in termen van kennis van elkaars interfaces en gedrag. Hechte koppeling kan leiden tot problemen bij hoge belasting of wanneer de server tijdelijk niet beschikbaar is.

- Losse koppeling betekent dat computersystemen zo min mogelijk van elkaar afhankelijk zijn. Zij communiceren via gestandaardiseerde interfaces of via intermediairs, zonder dat zij directe kennis hebben van elkaars implementatie. Een belangrijk aspect van losse koppeling is dat veranderingen in één computersysteem minder impact hebben op andere computersystemen. Dit bevordert flexibiliteit, schaalbaarheid en onderhoudbaarheid. Tegelijkertijd vraagt losse koppeling om duidelijke afspraken over gegevensstructuren, semantiek en governance, omdat impliciete afhankelijkheden worden vervangen door expliciete contracten.

Binnen Edukoppeling is losse koppeling een belangrijk uitgangspunt, maar dit betekent niet dat alle interacties volledig ontkoppeld moeten zijn. Er kunnen situaties zijn waar synchrone uitwisseling wenselijk of zelfs noodzakelijk is. Ketenpartners kunnen bewust en onderbouwd kiezen om een synchrone / hechte koppeling te accepteren.

6.2.5. Verwerking (synchroon / asynchroon)

Het kenmerk verwerking heeft betrekking op de functionele afhandeling van de ontvangen gegevens en niet op de wijze waarop computersystemen technisch met elkaar communiceren (interactie). Het beschrijft de wijze waarop de inhoudelijke afhandeling van een transactie plaatsvindt: direct binnen dezelfde processtap (synchroon) of losgekoppeld in

[77] Polling is niet opgenomen, we zien dit als een antipatroon.
[78] Link naar profiel XXXXXX
[79] Zoals OAS en AAS.

tijd (asynchroon). Een interactie kan synchroon verlopen, terwijl de verwerking asynchroon plaatsvindt.

Bij synchrone verwerking wordt de inhoudelijke verwerking uitgevoerd binnen dezelfde logische processtap als waarin de interactie plaatsvindt. De initiërende ketenpartner wacht op de uitkomst van de verwerking voordat het proces verder kan gaan. Bij asynchrone verwerking wordt de verwerking losgekoppeld van de initiële interactie. De ontvangende ketenpartner accepteert een verzoek of melding, maar verwerkt deze op een later moment.

De keuze voor synchrone of asynchrone verwerking moet expliciet worden afgewogen op basis van de proceskarakteristieken en staat in principe los van de te kiezen interactievorm.

6.3. Transactiepatronen

De keuze voor een transactiepatroon wordt bepaald door een samenhang van de hiervoor genoemde kenmerken. De kenmerken moeten in samenhang worden beschouwd om te komen tot een bepaald transactiepatroon. Binnen Edukoppeling onderkennen we een aantal gestandaardiseerde transactiepatronen. Dit zijn:

- bevraging;
;
- melding;
- melding + bevestiging;
- notificatie + ophalen;
- bulk synchronisatie (dataset-overdracht).

Transactiepatroon	Richting	Overdrachtswijze	Interactie	Koppeling	Verwerking
Bevraging	Tweezijdig	Pull	Sync	Gemiddeld / Hecht	Sync
Melding	Eenzijdig	Push	Async of Sync	Los	Async
Melding + bevestiging	Eenzijdig + aparte vervolginteractie	Push (+ optioneel vervolg push/pull)	Async en/of Sync (mixed)	Gemiddeld	Async
Notificatie + ophalen	Eenzijdig en Tweezijdig (hybride)	Push + Pull	Async en Sync (mixed)	Gemiddeld	Async
Bulk synchronisatie	Eenzijdig of Tweezijdig	Push of Pull	Sync of Async	Gemiddeld	Async

6.3.1. Bevraging

Doel: opvragen van gegevens of status

Bij het transactiepatroon bevraging bevraagt de gegevensafnemer de gegevensaanbieder gericht om gegevens of statusinformatie. Daarbij is het inhoudelijke antwoord noodzakelijk voor de voortgang van het proces bij de gegevensafnemer. Het patroon wordt toegepast wanneer een processtap afhankelijk is van actuele of gevalideerde informatie, bijvoorbeeld bij het raadplegen van registraties, het controleren van rechten of het uitvoeren van validaties.

- De transactie is functioneel tweezijdig van aard: de gegevensafnemer initieert de interactie en verwacht een inhoudelijke respons van de gegevensaanbieder. Het initiatief ligt daarmee bij de gegevensafnemer (overdrachtswijze pull). Deze kenmerken zijn inherent aan het functionele karakter van een bevraging en vormen geen afzonderlijke ontwerpkeuze.

- De gegevensafnemer vraagt over het algemeen een volledig en consistent beeld op van de gevraagde gegevens of status op een bepaald moment in de tijd. Om deze reden wordt een bevraging meestal gerealiseerd via een synchrone interactie, waarbij de gegevensafnemer tijdens dezelfde interactie wacht op een antwoord van de gegevensaanbieder.

- Door de afhankelijkheid in tijd en beschikbaarheid tussen gegevensafnemer en gegevensaanbieder kent dit patroon doorgaans een relatief hechte koppeling. Deze koppeling kan echter worden beperkt door gebruik te maken van gestandaardiseerde interfaces, stabiele contracten en generieke API-principes. Hierdoor kunnen systemen onafhankelijker evolueren terwijl de functionele interactie behouden blijft.

- De gegevensaanbieder verwerkt het verzoek direct en retourneert het resultaat binnen dezelfde interactie (synchrone verwerking). Interne onderdelen van de verwerking kunnen eventueel asynchroon worden ingericht, zolang dit transparant blijft voor de gegevensafnemer en geen invloed heeft op het overeengekomen gedrag van de interactie.

6.3.2. Melding

Doel: communiceren van een gebeurtenis of wijziging

Bij het transactiepatroon melding communiceert een gegevensaanbieder een gebeurtenis of een gegevenswijziging. De gegevensaanbieder informeert gegevensafnemer(s) zonder afhankelijk te zijn van een inhoudelijke reactie of directe verwerking.

Eventgedreven interactie is met name geschikt voor ketens waarin meerdere partijen afhankelijk zijn van wijzigingen in gegevens, zoals mutaties in registraties of statusupdates. Door de ontkoppeling kunnen nieuwe partijen eenvoudig aansluiten zonder impact op bestaande computersystemen. Tegelijkertijd stelt dit patroon hogere eisen aan governance, semantiek en beveiliging, omdat de context van events expliciet moet worden vastgelegd en behouden.

- De transactie is functioneel eenzijdig van aard. Het initiatief ligt bij de gegevensaanbieder (overdrachtswijze push), omdat deze bepaalt wanneer een gebeurtenis relevant genoeg is om te publiceren. De gegevensafnemer hoeft geen inhoudelijk antwoord terug te geven om de transactie functioneel compleet te maken.

- Een melding kan zowel synchroon als asynchroon worden verstuurd. Bij synchrone interactie blijft de response beperkt tot een technische ontvangstbevestiging.

- De mate van koppeling is doorgaans relatief los, vooral wanneer meldingen via een intermediair of event broker worden gepubliceerd. Hierdoor hoeven gegevensaanbieder geen kennis te hebben van individuele gegevensafnemers en kunnen meerdere gegevensafnemers onafhankelijk van elkaar gebeurtenissen verwerken.

- De functionele verwerking van de melding vindt altijd asynchroon plaats, omdat de gegevensaanbieder niet afhankelijk is van het resultaat van de verwerking bij de gegevensafnemer(s).

Binnen Edukoppeling is het van belang dat asynchrone patronen consistent worden toegepast en dat duidelijke afspraken worden gemaakt over statusmodellen, foutafhandeling en herlevering. Daarom is er ook een Communicatie-profiel[80] opgesteld dat bijvoorbeeld meer concreet invulling geeft op Asynchrone publish-subscribe interactie.

Publish-subscribe:
De client is een applicatie in de rol van 'producer' die 'events' publiceert: dataregistraties die een gebeurtenis en de bijbehorende context vastleggen. De server is een applicatie in de rol van 'consumer' die gepubliceerde events consumeren. Consumers abonneren zich op bepaalde type events. Er kunnen één of meerdere applicaties in de rol van 'intermediary' zijn die zorgdragen voor het routeren van events naar consumers op basis van contextuele informatie. Dit is vergelijkbaar met het publish-subscribe-patroon.

Deze componenten hebben de volgende interacties:

- Berichtenverkeer partij A naar partij B:
  - De producer van partij A produceert events (via zijn eigen Intermediary van partij A) naar de Intermediary van partij B
  - De Intermediary B persisteert het event zodanig dat consumer partij B deze kan consumeren óf stuurt het event direct door naar consumer partij B.
  - Consumer partij B verkrijgt het event van Intermediary B gerouteerd/gepushed óf consumeert de events van Intermediary B.
- Berichtenverkeer partij B naar partij A:
  - De producer van partij B produceert events (via zijn eigen Intermediary van partij B) naar de Intermediary van partij A.
  - De Intermediary A persisteert het event zodanig dat consumer partij A deze kan consumeren óf stuurt het event direct door naar consumer partij A.
  - Consumer partij A verkrijgt het event van Intermediary A gerouteerd/gepushed óf consumeert de events van Intermediary A.

6.3.3. Melding + bevestiging

Doel: communiceren van een gebeurtenis met latere inhoudelijke bevestiging
Bij dit patroon meldt een ketenpartner een gebeurtenis of gegevenswijziging, waarna op een later moment een inhoudelijke bevestiging van verwerking wordt vereist. Het patroon combineert daarmee de eigenschappen van een melding met de behoefte aan formele terugkoppeling over de verwerking.

[80] Link xxxxx

- De initiële melding (overdrachtswijze push) is functioneel eenzijdig. Het functionele initiatief ligt bij de gegevensaanbieder. De gegevensaanbieder bepaalt wanneer de melding wordt verstuurd. Met de melding wordt een wijziging, gebeurtenis of nieuwe toestand gecommuniceerd die door de gegevensafnemer verder verwerkt moet worden.

- De interactie kan zowel synchroon als asynchroon worden verstuurd. Bij synchrone interactie ontvangt de gegevensaanbieder direct een technische ontvangstbevestiging via hetzelfde communicatiekanaal. De inhoudelijke verwerking vindt daarna los van deze interactie plaats. Bij asynchrone interactie wordt de melding via messaging of eventing aangeboden zonder directe technische response. Ontvangst en verwerking worden volledig ontkoppeld in de tijd.

- De koppeling tussen partijen is sterker dan bij een eenvoudige melding, omdat een functionele relatie bestaat tussen de initiële melding en de latere bevestiging. Toch blijft de tijdsafhankelijkheid beperkt doordat verwerking ontkoppeld plaatsvindt.

- De verwerking van de melding is asynchroon. De gegevensafnemer verwerkt de gegevens op een later moment

Notificatie gevolgd door ophalen
Doel: signaleren dat gegevens beschikbaar zijn waarna afnemer deze zelf ophaalt
Dit patroon combineert een melding met een daaropvolgende bevraging. Een ketenpartner publiceert een notificatie dat gegevens beschikbaar zijn of gewijzigd zijn, waarna gegevensafnemers zelfstandig besluiten om de betreffende gegevens op te halen.

- De notificatie is functioneel eenzijdig. De gegevensaanbieder informeert gegevensafnemers (push) dat relevante informatie beschikbaar is gekomen. Het ophalen van de gegevens is daarentegen een tweezijdige pull-gebaseerde interactie, waarbij de gegevensafnemer zelf bepaalt wanneer en welke gegevens worden opgevraagd (pull).

- De koppeling tussen systemen is gemiddeld. De gegevensaanbieder hoeft niet te weten wanneer of door welke gegevensafnemers gegevens worden opgehaald, terwijl gegevensafnemers onafhankelijk kunnen bepalen hoe zij notificaties verwerken.

- Dit patroon combineert de voordelen van eventgedreven signalering met gecontroleerde gegevensverstrekking. Het voorkomt dat grote of privacygevoelige datasets direct via notificaties worden verspreid en geeft gegevensafnemer controle over het moment van ophalen (asynchrone verwerking).

6.3.4. Bulk synchronisatie (dataset-overdracht)

Doel: uitwisselen of synchroniseren van grotere datasets

Bulk synchronisatie betreft een transactiepatroon waarbij grotere hoeveelheden gegevens of volledige datasets tussen partijen worden uitgewisseld. Het patroon wordt toegepast wanneer het niet efficiënt of wenselijk is om wijzigingen afzonderlijk via individuele transacties of events te communiceren.

- De richting van de transactie kan variëren. In sommige situaties initieert de gegevensafnemer de synchronisatie door gegevens op te vragen (pull), terwijl in andere situaties de gegevensaanbieder periodiek datasets distribueert (push). Afhankelijk van de inrichting kan de interactie daarmee eenzijdig of tweezijdig zijn.

- Het doel is doorgaans het verkrijgen of herstellen van een volledige en consistente dataset, bijvoorbeeld bij initiële systeemvulling of periodieke synchronisatie.

- De interactie kan synchroon of asynchroon plaatsvinden

- De mate van koppeling is doorgaans middelmatig. Hoewel computersystemen minder afhankelijk zijn van directe interactie, bestaan vaak nog expliciete afspraken over formaat, frequentie en volledigheid van datasets. Bulk-verwerking vraagt bovendien aanvullende aandacht voor beveiliging, autorisatie en privacy, omdat vaak grote hoeveelheden gegevens tegelijk worden verwerkt.

- De verwerking vindt in de praktijk vaak asynchroon plaats vanwege de omvang van de datasets en de duur van de verwerking. Hiervoor worden vaak achtergrondprocessen, batchverwerking of tijdelijke opslagmechanismen toegepast.

6.4. Ontwerpkaders

1 Het gekozen transactiepatroon sluit aan op de functionele eisen en proceskarakteristieken van de ketensamenwerking. De keuze voor een interactie- of transactiepatroon volgt uit de kenmerken van de ketenprocesstap. Niet de beschikbare techniek, maar de behoefte aan directe respons, verwerkingstijd, betrouwbaarheid, schaalbaarheid en autonomie van ketenpartners is leidend.

2 Richting, overdrachtswijze, interactie en verwerking beschrijven verschillende aspecten van een transactiepatroon.
   a. Richting beschrijft of functioneel een inhoudelijke reactie nodig is (eenzijdig of tweezijdig).
   b. Overdrachtswijze beschrijft welke ketenpartner de gegevensoverdracht of actie start.
   c. Interactie beschrijft de technische afhankelijkheid in tijd.
   d. Verwerking beschrijft wanneer de inhoudelijke afhandeling plaatsvindt.

3 Interactie en verwerking worden afzonderlijk ontworpen: Synchrone interactie betekent niet automatisch synchrone verwerking. Een bericht kan synchroon worden aangeleverd en technisch worden bevestigd, terwijl de inhoudelijke verwerking later plaatsvindt. Voor iedere transactie worden interactietiming en verwerkingstiming afzonderlijk vastgesteld en gedocumenteerd.

4 Synchroon wordt bij voorkeur toegepast bij functionele tijdsafhankelijkheid: Synchrone interactie introduceert directe afhankelijkheid van beschikbaarheid en responstijd. Het past wanneer het proces niet verder kan zonder een onmiddellijk resultaat, bijvoorbeeld bij een bevraging, validatie of eenvoudige opdracht.

5 Asynchroon wordt toegepast als ontkoppeling of langdurige verwerking belangrijk zijn: Asynchrone interactie is passend wanneer verwerking niet direct hoeft plaats te vinden, tijdelijke onbeschikbaarheid moet kunnen worden opgevangen, meerdere afnemers betrokken zijn of de verwerking langere tijd duurt. Langdurige, schaalgevoelige of zelfstandig uitvoerbare verwerking wordt bij voorkeur asynchroon ingericht.

6 De mate van koppeling is een zelfstandige ontwerpkeuze: Synchrone interactie leidt vaak tot sterkere koppeling, maar hoeft niet per definitie zo te zijn. Zo kan een point-to-point messaging koppeling sterk gekoppeld zijn, terwijl een gestandaardiseerde REST-API relatief los gekoppeld kan worden ingericht.

7 De overdrachtswijze wordt bepaald door de functionele rol die de overdracht initieert: bij push initieert de gegevensaanbieder de overdracht naar de gegevensafnemer; bij pull initieert de gegevensafnemer de interactie om gegevens bij de gegevensaanbieder te verkrijgen.

8 Technische ontvangst en inhoudelijke bevestiging worden gescheiden: Een technische ontvangstbevestiging toont alleen aan dat een bericht technisch is ontvangen en/of geaccepteerd. Zij zegt niets over de inhoudelijke verwerking.

Wanneer een inhoudelijke verwerkingsuitkomst nodig is, wordt deze als afzonderlijke interactie gemodelleerd en gecorreleerd aan de oorspronkelijke melding of opdracht.

9 Samengestelde patronen worden expliciet gemodelleerd: De gegevensuitwisseling binnen een ketenprocesstap kan meerdere patronen combineren. Zo bestaat notificatie gevolgd door ophalen uit een push-gebaseerde melding en een pull-gebaseerde bevraging. Melding plus bevestiging bestaat uit een initiële melding en een afzonderlijke terugkoppeling.

10 Beveiliging en privacy gelden voor iedere interactie: Ook binnen een samengesteld patroon moet iedere afzonderlijke interactie voldoen aan eisen voor, identificatie, authenticatie en transportbeveiliging.

7. API-architectuur

7.1. Ontwerpkaders

1 API (contract) first[81]: Gegevensuitwisseling tussen applicaties wordt gerealiseerd via gestandaardiseerde API's. Dit betekent dat de uitwisseling tussen ketenpartners binnen een ketensamenwerking vooraf expliciet wordt beschreven in één of meer technische contracten, voordat de implementatie wordt gerealiseerd. Deze contracten vormen de afspraak tussen gegevensaanbieder en gegevensafnemer en beschrijven onder meer de interface, de berichtenstructuren, de beveiliging, de wijze van identificatie en authenticatie en de ondersteunde interacties.

2 Technisch vertrouwen is een afgeleide van organisatorisch vertrouwen. Identificatie, authenticatie en autorisatie kunnen uitsluitend betekenis krijgen wanneer vooraf is vastgesteld welke organisaties samenwerken, welke verantwoordelijkheden zij hebben en onder welke voorwaarden zij gegevens mogen verwerken. Onderdeel hiervan zijn identificatie en authenticatie

3 Een ketensamenwerking moet unieke en betrouwbare identificatie van organisaties[82] toepassen: We gebruiken hiervoor bij voorkeur het OIN[83] omdat dit een bestaand OIN-stelsel is dat wordt toegepast binnen Digikoppeling voor identificatie van een rechtspersoon of organisatieonderdeel. De clientregistratie moet de relatie vastleggen tussen de technische client (client_id) en de organisatie (OIN) waarvoor deze client opereert. Indien een OIN niet in het landelijke OIN-register voorkomt, wordt de identiteit van de organisatie vastgesteld via het hoofdnummer en een daarvoor aangewezen authentieke registratie.

4 Clientauthenticatie is verplicht en beperkt tot gestandaardiseerde methoden. Iedere Client moet zich bij de Authorization Server authenticeren voordat een Access Token wordt uitgegeven. De toegestane clientauthenticatiemethoden is beperkt tot een beheersbare set, zodat ketenpartners deze interoperabel en veilig kunnen ondersteunen. De client wordt geïdentificeerd met een door de AS toegewezen client_id. De client_id is herleidbaar naar de identiteit van de organisatie die verantwoordelijk is voor de client.

5 Verificatie van machtiging verplicht waar nodig: Daar waar dit voor een bepaalde interactie van een client is vastgelegd moeten authorization_details met edu-from en edu-to worden opgenomen in het request naar het token endpoint van de Authorization Server. Dit is het geval als er gebruikt wordt gemaakt van een API namens een onderwijsorganisatie. De Authorization Server neemt na verificatie van de machtiging edu-from en edu-to op in het Access Token. De API-architectuur onderkent hiermee het concept van een machtiging, maar definieert niet het machtigingenbeheer.

[81] https://docs.geostandaarden.nl/api/API-Strategie/#api-first-strategie
[82] https://rosa.wikixl.nl/index.php/Id-d0d50170e49d45529c4a0c37d9dab4cc
[83] Zie https://gitdocumentatie.logius.nl/publicatie/dk/idauth/1.5.0/#bijlage-1-nummersystematiek-oin-en-hrn

6 Resource- en event-georiënteerd naast elkaar: Actuele toestand wordt geraadpleegd bij de bron; gebeurtenissen informeren over relevante wijzigingen. Gegevensdiensten (RESTful API's) en eventvoorzieningen zijn complementair. Gegevensdiensten worden toegepast voor het opvragen van actuele gegevens, directe validatie en opdrachten. Eventvoorzieningen worden toegepast voor het signaleren van wijzigingen, losgekoppelde samenwerking en langlopende verwerking. De keuze volgt uit de proceskarakteristieken van de ketensamenwerking.

7 Voor ieder koppelvlak bestaat een machineleesbaar, versioneerbaar technisch API-contract. Het Communicatie-profiel positioneert het contractregister[84] als bron van waarheid voor specificaties en benoemt dat contracten afspraken vastleggen over schemastructuren, eigenaarschap, kwaliteit, berichtformaten. De voorschriften rond API-contracten zijn in het Communicatie-profiel beschreven.

8 Er is geen directe verbinding tussen publishers en consumers; Bij asynchrone interactie wordt uitgegaan van intermediairs aan zowel publisher- als consumerzijde die met elkaar communiceren. Deze intermediairs zorgen voor robuuste uitwisseling van gegevens door maatregelen zoals buffering, routering, monitoring

9 Lifecyclebeheer is onderdeel van de API-architectuur: API's, tokens, sleutels, certificaten en contracten hebben een lifecycle. Wijzigingen moeten beheerst plaatsvinden met aandacht voor versiebeheer, backward compatibility, deprecatie, sleutelrotatie en certificaatvervanging.

10 De API-architectuur gebruikt breed toegepaste open internationale industrie- en Edustandaard standaarden. Voor RESTful API's worden HTTP, JSON en OpenAPI toegepast. Voor eventvoorzieningen wordt AsyncAPI toegepast. Voor beveiliging worden OAuth 2.0, JWT, Edustandaard UBV TLS en waar relevant RFC 9396 gebruikt.

11 Bij ketentesten tussen acceptatieomgevingen moet voor authenticatie en transportbeveiliging gebruik worden gemaakt van certificaten en sleutelmateriaal die uitsluitend voor de acceptatieomgeving zijn uitgegeven. Een productie-identifier moet worden gebruikt.

7.2. Toelichting API-architectuur

Binnen de Edukoppeling-architectuur staat het gebruik van API's centraal. Een API vormt het gestandaardiseerde toegangspunt waarmee een ketenpartner gegevens of functionaliteit beschikbaar stelt aan andere ketenpartners. De API abstraheert de achterliggende implementatie en biedt een stabiel technisch contract waartegen gegevensafnemers kunnen integreren. Toegang tot de met Edukoppeling beveiligde API's vindt plaats op basis van de OAuth 2.0 Client Credentials Grant.

[84] Zie **Fout! Verwijzingsbron niet gevonden.**

7.2.1. Technisch vertrouwensmodel

Op basis van het organisatorische vertrouwensmodel wordt binnen een ketensamenwerking het technische vertrouwen ingericht. Daarbij worden twee samenhangende, keuzes gemaakt. De eerste keuze betreft hoe wordt vastgesteld welke systemen mogen worden vertrouwd (identificatie). Dit kan bijvoorbeeld doordat een client vooraf wordt geregistreerd, of doordat een gezaghebbend register wordt geraadpleegd, of doordat vertrouwen via een federatieve trust chain[85] kan worden afgeleid. De tweede keuze betreft de manier waarop de verificatie van die identiteit wordt gebaseerd (authenticatie). Hiervoor wordt een technisch vertrouwensanker gebruikt, bijvoorbeeld een vooraf geregistreerd gedeeld geheim, een vooraf geregistreerde publieke sleutel of een vertrouwde certificaatautoriteit (CA) binnen een Public Key Infrastructure (PKI). Het technisch vertrouwensmodel beschrijft dus hoe deelnemende partijen en hun systemen elkaar identificeren en authenticeren. Iedere organisatie en hun systeem moet eenduidig identificeerbaar zijn, zodat kan worden vastgesteld welke ketenpartner deelneemt aan de gegevensuitwisseling. Vervolgens wordt met authenticatie de identiteit van het systeem vastgesteld. De keuzes die Edukoppeling heeft gemaakt zijn dat een client vooraf wordt geregistreerd en dat clientauthenticatie kan worden uitgevoerd op een manier die passend is voor het risicoprofiel van de uitwisseling. Clientauthenticatie kan ingericht worden voor een laag en verhoogd risicoprofiel.

Clientauthenticatie bij een laag risicoprofiel

De gesloten data bevatten geen of slechts beperkt gevoelige informatie en hebben een beperkte impact bij misbruik, er is een laag risico. Client-authenticatie bij het OAuth token-endpoint gebeurt in dit geval op basis van een gedeeld geheim (client-id en wachtwoord) via HTTP Basic Authentication[86].

Clientauthenticatie bij een verhoogd risicoprofiel

Er geldt een verhoogd risicoprofiel als een API gesloten data ontsluit en de impact bij misbruik hoog zal zijn. Er worden hogere eisen gesteld aan clientauthenticatie. Client-authenticatie wordt in dit geval uitgevoerd op basis van een vertrouwde certificaatautoriteit (CA) binnen een Public Key Infrastructure (PKI). Organisaties beschikken over digitale certificaten die zijn uitgegeven door een vertrouwde Certificate Authority (CA). Door middel van deze certificaten kunnen partijen elkaar cryptografisch identificeren en authenticeren. Het voordeel van dit model is de sterke zekerheid over identiteit en de duidelijke governance via certificaatuitgifte. Nadeel is dat het minder flexibel is: onboarding kan tijd kosten en internationale of kleinere partijen hebben niet altijd toegang tot dezelfde PKI-infrastructuur. Het digitale certificaat wordt gebruikt voor creatie van een ondertekende JWT[87]. Deze dient als bewijs dat de client over de private sleutel beschikt. Deze beveiligingsmaatregelen passen bij een verhoogd risicoprofiel, een ketensamenwerking waarin integriteit, traceerbaarheid en schade bij misbruik aanzienlijk zijn.

Clientauthenticatie bij een verhoogd risico met toepassing van overheidsstandaarden

In de basis worden de beveiligingsmaatregelen van het verhoogd risicoprofiel toegepast, maar daarnaast worden ook overheidsstandaarden en voorzieningen toegepast, zoals

[85] OpenID Federation 1.0:
[86] Zie Edukoppeling OAuh 2.0 client credantials profiel en de basic authentication optie
[87] Zie Edukoppeling OAuh 2.0 client credantials profiel en de OIDC private_key_jwt methode

PKIoverheid[88] certificaten. Omdat dit profiel ook al de toepassing van het OIN voorschrijft krijgt men met PKIoverheid ook een sterke identificatie van de ketenpartner omdat de uitgifte van het certificaat gebonden is aan strikte controles door erkende aanbieders (TSP's) onder toezicht van Logius[89]. Het biedt hiermee hogere mate van betrouwbaarheid en sluit aan op afspraken binnen het overheidsdomein.

Edukoppeling stelt dat in ieder geval in communicatie met de overheid (DUO) wordt gekozen voor clientauthenticatie op basis van 'Verhoogd risico met toepassing van overheidsstandaarden'. Zie voor meer details het Edukoppeling OAuth 2.0 Client Credentials profiel[90].

Machtiging
Een organisatie kan met haar systemen namens verschillende onderwijsorganisaties gegevens uitwisselen met verschillende ketenpartners. Wanneer een organisatie gegevens uitwisselt met een andere ketenpartner dan moet kunnen worden vastgesteld namens welke organisatie wordt gehandeld en toegang wordt gevraagd. Binnen Edukoppeling wordt dit scenario ondersteund met een machtiging[91]. Een machtiging is een bevoegdheid die een entiteit verleent aan een andere entiteit om in naam van eerstgenoemde bepaalde handelingen te verrichten. De autorisatiebeslissing wordt in dit geval niet uitsluitend gebaseerd op de authenticatie van de client, maar ook op de organisatorische context waarin deze handelt. De client kan bij de tokenaanvraag deze context meegeven (zie Edukoppeling OAuth 2.0 client credentials profiel). Met de gegevens die de client meelevert kan worden afgeleid of de client bevoegd is om namens een bepaalde onderwijsorganisatie gegevens uit te wisselen met een bepaalde derde partij.

Transportbeveiliging
TLS zorgt ervoor dat gegevens onderweg niet onderschept of gewijzigd kunnen worden. Het zorgt voor vertrouwelijkheid en integriteit tijdens het transport van de gegevens. Voor transportbeveiliging moet TLS worden toegepast conform de Edustandaard UBV TLS[92]. Het gebruik van een PKIoverheid-certificaat is daarbij niet verplicht, tenzij dit vanuit de betreffende ketensamenwerking als eis wordt gesteld. Een ketensamenwerking mag voor de communicatie tussen de client en Resource Server mTLS toepassen. Ook hiervoor bepaalt de ketensamenwerking welke certificaten (CA) moeten worden toegepast

7.2.2. Standaarden

De Edukoppeling API-Architectuur sluit aan op breed toegepaste internationale open standaarden. Het gebruik van open standaarden voorkomt leveranciersafhankelijkheid, ondersteunt interoperabiliteit tussen ketenpartners en maakt aansluiting mogelijk op bestaande producten, cloudplatformen en publieke voorzieningen.

API standaarden

[88] https://www.pkioverheid.nl/
[89] PKIoverheid-certificaten bevatten het Organisatie-identificatienummer (OIN). De CA's die een PKIoverheid certificaat uitgeven verifiëren de identiteit van de organisatie in het handelsregister (Logius | PKIoverheid)
[90] Link OAuth profiel
[91] https://rosa.wikixl.nl/index.php/Begrip:C70d1782-8382-4f88-bf5f-b13c88fd291b
[92] Zie Edukoppeling OAuth 2.0 client credentials profiel en UBV TLS: https://www.edustandaard.nl/standaard_afspraken/uniforme-beveiligingsvoorschriften/

De relevante standaarden voor synchrone en asynchrone interacties worden in detail toegelicht in het Edukoppeling Communicatie-profiel[93].

Voor synchrone interacties wordt uitgegaan van RESTful API's die gegevens en functionaliteit beschikbaar stellen via HTTP-gebaseerde interfaces. Daarbij wordt gebruikgemaakt van JSON als primair formaat voor gegevensuitwisseling. Voor de formele beschrijving van RESTful API's wordt aangesloten op de OpenAPI Specification (OAS[94]). Hiermee kunnen API-contracten op een gestandaardiseerde wijze worden beschreven, gepubliceerd en gevalideerd. OAS ondersteunt daarnaast automatische documentatie, validatie en codegeneratie, waardoor interoperabiliteit tussen aanbieders en afnemers wordt vergroot.

Voor EDA interacties wordt gebruikgemaakt van CloudEvents als gestandaardiseerd eventmodel. CloudEvents definieert uniforme metadata voor gebeurtenissen, zoals identificatie, bron, tijdstip en type event. Hierdoor kunnen publishers, brokers en subscribers interoperabel samenwerken, onafhankelijk van de gebruikte messaging-technologie. Voor de beschrijving van asynchrone API's en eventstromen wordt aangesloten op de AsyncAPI Specification (AAS[95]). AsyncAPI vervult hierbij een vergelijkbare rol als OAS voor RESTful API's en ondersteunt het formeel beschrijven van eventkanalen, berichtenstructuren en bindings naar messaging-platformen.

Beveiligingsstandaarden
Voor beveiliging wordt binnen Edukoppeling gebruikgemaakt van de OAuth 2.0 Client Credentials grant. De relevante standaarden voor beveiliging worden in detail toegelicht in het Edukoppeling OAuth client credentials profiel voor RESTful API's[96]. OAuth biedt een gestandaardiseerd autorisatiemodel waarbij clients gecontroleerd toegang verkrijgen tot API's via Access Tokens. De toepassing van OAuth 2.0 zorgt voor flexibiliteit en een duidelijke scheiding in rollen zodat rechten flexibel kunnen worden toegekend en beheerd. Het gebruik van tokens maakt het mogelijk om autorisatie los te koppelen van authenticatie. Een belangrijk kenmerk van moderne ketens is dat zij steeds vaker een open karakter krijgen. Dit betekent dat nieuwe partijen relatief eenvoudig moeten kunnen aansluiten en dat gegevensuitwisseling niet beperkt is tot een vaste groep deelnemers. Verder onderkennen we dat niet elke ketensamenwerking hetzelfde beveiligingsniveau vereist. Daarom ondersteunt de architectuur meerdere risicoprofielen. Dit maakt het mogelijk om beveiliging proportioneel toe te passen, afhankelijk van de gevoeligheid van de gegevens en de risico's binnen de keten.

7.2.3. Gebruik van certificaten en identifiers in een ketentest

Edukoppeling gaat in principe om voorschriften voor productieomgevingen. We merken echter dat bij ketentesten (acceptatieomgevingen) verschillende ketens verschillende keuzes maken rond identiteiten en te gebruiken certificaten. Dit is inefficiënt en introduceert onnodige risico's. Een goed ingericht testbeleid draagt bij aan de veiligheid van Edukoppeling-implementaties. Met de onderstaande afspraken creëren we meer uniformiteit

[93] Link comm profiel
[94] https://swagger.io/specification/
[95] https://www.asyncapi.com/docs/reference/specification/v3.1.0
[96] Link Oauth profiel

hoe de met Edukoppeling beveiligde API's in een acceptatieomgeving worden geconfigureerd. Bij een ketentest gelden de volgende afspraken:

1. De certificaten hebben een ander vertrouwensanker[97] dan in productie.
    a. Bij toepassing van PKIoverheid: PKIoverheid vereist de toepassing van een andere root-CA voor testcertificaten[98]. Voor testomgevingen worden TRIAL-certificaten[99] met een PKIoverheid TRIAL root certificaat geboden. Hierbij kan een ketensamenwerking (eventueel in overleg met Logius) een eigen "fauxTSP" CA-certificaat[100] genereren.
    b. Het testen van sleutelmanagement is onderdeel van het testprogramma.
2. Keuzes rond gegevens vallen in principe buiten de scope van Edukoppeling, maar met betrekking tot identificerende kenmerken van organisaties (in het PKIoverheid certificaat) stellen we dat deze binnen een acceptatieomgeving overeen moeten komen met die van productie[101]. Het gebruik van een productie-identifier (OIN) mag echter niet automatisch leiden tot productierechten en toegang tot productiedata.

[97] Een testcertificaat heeft een andere root-CA dan in productie. Testcertificaten worden zo niet impliciet vertrouwd in de productieomgeving (zie ook Digikoppeling beveiligingsvoorschriften PKIoverheid programma van eisen https://gitdocumentatie.logius.nl/publicatie/dk/beveilig/2.0.1/#pkioverheid-programma-van-eisen)
[98] Zie #3 https://gitdocumentatie.logius.nl/publicatie/dk/beveilig/2.0.1/#pkioverheid-programma-van-eisen
[99] GitHub - pkioverheid/g4-trial: Generate TRIAL certificates for the PKIoverheid G4 hierarchies · GitHub
[100] https://github.com/pkioverheid/g4-trial?tab=readme-ov-file#overview
[101] Logius heeft eerder in communicatie met DUO het volgende gesteld*: De concensus tijdens de sessie is dat het gebruik van een productie-OIN niet noodzakelijk slecht is in testsituaties omdat dit zoveel mogelijk de productiesituatie benaderd, mits er gebruik gemaakt wordt van niet-productiecertificaten*. (zie 2026-02-11-Verslag-Edustandaard-Werkgroep-Edukoppeling.pdf)

8. Bijlage A: Begrippen

Begrip	Definitie	Bron
API	Een API is een gestandaardiseerd technisch contract waarmee een applicatie gegevens of functionaliteit beschikbaar stelt aan andere applicaties. Binnen Edukoppeling omvat het begrip API zowel synchrone RESTful API's als asynchrone API's voor eventpublicatie, eventontvangst en subscriptionbeheer.	Edukoppeling
Open en gesloten data	Open Data zijn gegevens die in een open formaat door iedereen voor alle doeleinden vrij gebruikt, hergebruikt en gedeeld kunnen worden. De nadruk voor Open Data ligt met name bij de gegevens van de overheid. Gegevens die om reden van privacy, veiligheid, wettelijke verplichtingen en dergelijk niet onder de definitie vallen noemen we in dit document Gesloten Data.	Digikoppeling Architectuur 2.1.0
Confidential client[102]	Een confidential client is een client die draait in een omgeving waar vertrouwelijke gegevens (waaronder het client secret) veilig bewaard kunnen worden (bijvoorbeeld server-side webapplicaties).	RFC6749
Bulk-verwerking	Ook wel batchverwerking, betreft het uitwisselen of verwerken van grotere hoeveelheden gegevens in één samenhangende operatie, in plaats van individuele transacties per bericht of API-aanroep.	Edukoppeling
REST	Representational State Transfer	Fielding Dissertation
RESTful API	Een RESTful API is een application programmable interface (API) die HTTP-methoden, zoals GET, POST, PUT, PATCH en DELETE, gebruikt om resources te beheren. Resources worden geïdentificeerd via URIs (Uniform Resource Identifiers) en worden doorgaans geretourneerd in JSON- of XML-formaat. We noemen ze RESTful omdat ze niet aan alle	Wikipedia

[102] https://datatracker.ietf.org/doc/html/rfc6749#section-2.1

	REST[103] principes[104] hoeven te voldoen.	
Vertrouwensmodel (trustmodel)	Het samenhangende geheel van organisatorische, juridische, semantische en technische afspraken waarmee ketenpartners elkaars identiteit, bevoegdheden en digitale verklaringen kunnen vertrouwen.	Edukoppeling
Vertrouwensanker (trustanchor)	Een vooraf vertrouwde bron waarmee de geldigheid van digitale identiteiten, certificaten, sleutels of metadata kan worden vastgesteld en waarop het vertrouwensmodel uiteindelijk is gebaseerd.	Edukoppeling

[103] REST - Wikipedia
[104] Dit Edukoppeling-profiel gaat uit van gesloten data waarbij ketenpartners weten wat ze van elke vragen binnen een bepaalde ketensamenwerking. Ondersteuning van HATEOAS lijkt niet noodzakelijk.