

RFC7807

Analyse en implementatie van gestandaardiseerde foutresponses

Ton Donker

ton.donker@enexis.nl

Versie: 20211103
3 november 2021



ENEXIS
GROEP



Change history

Version	Changes	Document
20211103	Eerste release	RFC7807_analyse_implementatie_20211103.pptx



Introductie

Zowel Geonovum¹ als DSO² verwijzen in hun richtlijnen naar RFC7807³ als standaard voor het gestructureerd representeren van het error response formaat behorende bij de HTTP error codes.

Dit document bevat een uitvoerige analyse van RFC7807 en de implementatiewijze ervan in de verschillende NL-overheids APIs.

Doelstelling van dit document is te komen tot concrete en gedetailleerde richtlijnen voor het ontwerpen van RFC7807-compliant APIs, gebaseerd op zowel de inzichten van de API werkgroepen van de NL-overheid als ervaringen in internationale werkgroepen.

Dit document is als volgt opgebouwd:

- allereerst worden de details van RFC7807 toegelicht;
- gevolgd door een analyse van de implementatiewijze van RFC7807 in de NL-overheids APIs;
- de onderdelen van het problems detail object van RFC 7807 worden verklaard;
- dit document wordt afgesloten met een concreet eindvoorstel voor implementatie van RFC7808 voor APIs in NL-energie sector;

1. <https://docs.geostandaarden.nl/api/API-Strategie-ext/#error-handling-0>
2. <https://iplo.nl/digitaal-stelsel/aansluiten/standaarden/api-en-uri-strategie/> (API B-48)
3. <https://datatracker.ietf.org/doc/html/rfc7807> - update van RFC7807 vinden we hier: <https://github.com/ietf-wg-httpapi/rfc7807bis/issues>

403 Access to Operation Denied



RFC7807

RFC7807 definieert de structuur van het problem details object en de JSON-representatie ervan bevat de volgende onderdelen¹ weergegeven in onderstaande tabel. Merk op dat de '400' status code een afzonderlijke representatie heeft. Deze wordt in de volgende slides getoond. De standaard attributen zijn:

Onderdeel ²	Type - format	Uitleg	Voorbeeld	Opmerkingen
type	string - URI	Verwijzing naar externe documentatie met human readable uitleg over veld status	https://datatracker.ietf.org/doc/html/rfc7231#section-6.5.4	Doelgroep: ontwikkelaar Doel: aanvullende documentatie
title	string	Korte, human readable beschrijving van de fout. Heeft een 1:1 relatie met veld 'status'	Resource could not be found	Doelgroep; consumer
status	int32	De HTTP-status code	404	Komt ook voor in header van response
detail	string	Gedetailleerde beschrijving van fout en mogelijke oplossing	The server has not found anything matching the Request-URI	Doelgroep: consumer Doelstelling: recovery van fout
instance	string - URI	URI van de aanroep die de fout heeft veroorzaakt	https://datapunt.voorbeeldgemeente.nl/api/v1/resource-naam?parameter=waarde	Doelgroep: beheerder Doelstelling: tracking en tracing in logfiles

1. Bronnen:

<https://datatracker.ietf.org/doc/html/rfc7807#section-3.1>

<https://opensource.zalando.com/restful-api-guidelines/models/problem-1.0.0.yaml>

<https://iplo.nl/digitaal-stelsel/aansluiten/standaarden/api-en-uri-strategie/> (2.5.17)

4 2. Cardinaliteit van de onderdelen is optioneel. Zie ook: <https://blog.axway.com/amplify-products/api-management/introduction-to-rfc-7807>



RFC7807 – voorbeeld 401

Voorbeelden:	Bron:
<pre>{ "type": "URI: {fout}", "code": "Systeemcode die het type fout aangeeft", "title": "Hier staat wat er is misgegaan", "status": 401 "detail": "Meer details over de fout staan hier", "instance": "urn:uuid:ebd2e7f0-1b27-11e8-accf-0ed5f89f718b" }</pre>	Geonovum documentatie: API Designrules Extensions (Nederlandse API Strategie IIb) (geostandaarden.nl)
<pre>{ "type": "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.2", "title": "Niet correct geauthenticeerd.", "status": 401, "detail": "The request requires user authentication. The response MUST include a WWW-Authenticate header field (section containing a challenge applicable to the requested resource).", "instance": "https://datapunt.voorbeeldgemeente.nl/api/v1/ressourcenam?parameter=waarde", "code": "authentication" }</pre>	Haal Centraal API: https://developer.overheid.nl/detail/vng-bevraging-ingeschreven-persoon/production/specification

Naast de standaard attributen zien we in deze voorbeelden dat attribuut 'code' is toegevoegd. RFC7807 staat uitbreiding van het problem details object toe met aanvullende attributen¹ om extra informatie mee te sturen.

1. <https://datatracker.ietf.org/doc/html/rfc7807#section-3.2>



RFC7807 – voorbeeld 400

Voorbeelden:	Bron:
<pre>{ "type": "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.1", "title": "Een of meerdere parameters zijn niet correct.", "status": 400, "detail": "De foutieve parameter(s) zijn: verblijfplaats__huisnummer.", "instance": "https://datapunt.voorbeeldgemeente.nl/api/v1/resource naam?verblijfplaats_huisnummer=ABC", "code": "paramsValidation", "invalidParams": [{ "type": "https://www.vng.nl/realisatie/api/validaties/integer", "name": "verblijfplaats__huisnummer", "code": "integer", "reason": "Waarde is geen geldige integer." }] }</pre>	<p>Haal Centraal API:</p> <p>https://developer.overheid.nl/detail/vng-bevraging-ingeschreven-persoon/production/specification</p>

In dit voorbeeld wordt een array *InvalidParams* meegegeven in de error response. Elke foutieve query parameter wordt apart vermeld in dit array. Het kan dus zijn dat het *detail* veld meerdere foutieve parameters omvat. Het voorbeeld dat RFC7807¹ geeft bevat alleen de velden *name* en *reason*. Array naam is in RFC7807 geformuleerd als *invalid-params* (met hyphen)².



RFC7807 implementatie in NL-overheid APIs

Er zijn wat nuance verschillen in de RFC7807 implementatie van NL-overheids APIs. Een aantal voorbeelden:

1. Hoort veld 'type' wel thuis binnen 'invalidParams' element¹? De APIs t.b.v. basisregistratie personen (BRP) passen dit veld wel toe², de APIs voor zaakgericht werken (ZGW) niet³. Beide repositories (BRP en ZGW) vallen onder de noemer van VNG Realisatie, als onderdeel van de Vereniging Nederlandse Gemeenten;
2. Cardinaliteit: de ZGW APIs maken alle velden (behalve het 'type') veld verplicht⁴, de Kadaster BAG APIs maken alle velden optioneel⁵, de Logius APIs hebben 'type' ook verplicht⁶;
3. Er wordt in de diverse APIs verschillend doorverwezen naar de waarde van veld 'type'. Zowel verwijzingen naar Microsoft (https://docs.microsoft.com/en-us/dotnet/api/system.net.httpstatuscode?#System_Net_HttpStatusCode_InternalServerError) als verwijzingen naar de IETF RFC's (<https://datatracker.ietf.org/doc/html/rfc7231#section-6.5.1>) komen voor⁷;
4. Extensie attribuut 'code' wordt niet in alle APIs gebruikt, de Logius APIs⁶ bijvoorbeeld gebruiken dit veld niet. DSO maakt in de richtlijnen geen melding van dit veld;

➤ **Conclusie: géén uniformiteit in implementatie van RFC7807 in de NL-overheid APIs**

1. <https://github.com/Geonovum/KP-APIs/issues/289>
2. <https://github.com/VNG-Realisatie/Haal-Centraal-BRP-bevragen>
3. <https://github.com/VNG-Realisatie/gemma-zaken>
4. <https://github.com/VNG-Realisatie/gemma-zaken>
5. <https://vng-realisatie.github.io/Haal-Centraal-BAG-bevragen/swagger-ui/>
6. <https://developer.overheid.nl/detail/logius-cor/production/specification>
7. <https://github.com/VNG-Realisatie/Haal-Centraal-BAG-bevragen/issues/437>



RFC7807 implementatie *Type*

- **Type:** bevat een URI referentie dat het problem type identificeert. RFC7807 adviseert het gebruik van HTML-pagina's met documentatie over het probleem type. Als er geen URI referentie kan worden opgegeven, bevat dit attribuut default waarde '*about:blank*'. Voorstel is om te verwijzen naar RFC7231¹ als standaard voor het HTTP protocol. Bijvoorbeeld, voor status code 404, zal attribuut *Type* waarde hebben:

<https://datatracker.ietf.org/doc/html/rfc7231#section-6.5.4>

Merk op dat niet alle HTTP status code in RFC7231 worden genoemd (bijv. code 422). Deze foutcode wordt gespecificeerd RFC4918:

<https://datatracker.ietf.org/doc/html/rfc4918#section-11.2>

Verder lijkt het zinvol dit veld verplicht te maken met '*about:blank*' als default waarde. Zie ook de RFC7807 beschrijving voor het toevoegen van nieuwe problem types²;

➤ **Conclusie: attribuut '*Type*' verplicht en verwijst naar RFC gestandaardiseerde documentatie**

1. Het HTTP protocol wordt beschreven in verschillende RFC's. Zie ook: <https://serverfault.com/questions/979639/is-rfc-7231-the-latest-version-about-the-http-1-1-protocol>
2. <https://datatracker.ietf.org/doc/html/rfc7807#section-4>



RFC7807 implementatie *Title*

- **Title:** een attribuut dat de fout kort omschrijft. De API provider heeft hier vrijheid in, mits omschrijving in het UK Engels is voorbeeld: *403: No permission to access requested resource*¹;

De korte omschrijving die in RFC7231 wordt gegeven lijkt ook een geschikte kandidaat², maar deze omschrijving is wel heel summier en kan in de OAS omschrijving worden opgenomen. Voorbeeld³: 'Forbidden' in description veld:

```
"403":
  description: Forbidden
  headers:
    api-version:
      $ref: '#/components/headers/api_version'
  content:
    application/problem+json:
      schema:
        $ref: '#/components/schemas/Foutbericht'
      example:
        type: https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.4
        title: U bent niet geautoriseerd voor deze operatie.
        status: 403
        detail: The server understood the request, but is refusing to fulfill it.
        instance: https://datapunt.voorbeeldgemeente.nl/api/v1/resourceenaam?parameter=waarde
        code: autorisation
```

403

Forbidden

Media type

application/problem+json

Example Value | Schema

```
{
  "type": "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.4",
  "title": "U bent niet geautoriseerd voor deze operatie.",
  "status": 403,
  "detail": "The server understood the request, but is refusing to fulfill it.",
  "instance": "https://datapunt.voorbeeldgemeente.nl/api/v1/resourceenaam?parameter=waarde",
  "code": "autorisation"
}
```

➤ **Conclusie:** attribuut 'Title' verplicht en vrij in te vullen cf. bovengenoemde conventie

1. Zie Appendix A voor een overzicht van mogelijke waarden voor *Title*
2. Zie kolom Description: <https://www.iana.org/assignments/http-status-codes/http-status-codes.xhtml>
3. <https://vng-realisatie.github.io/Haal-Centraal-BRP-bevragen/swagger-ui>





RFC7807 implementatie *Status*

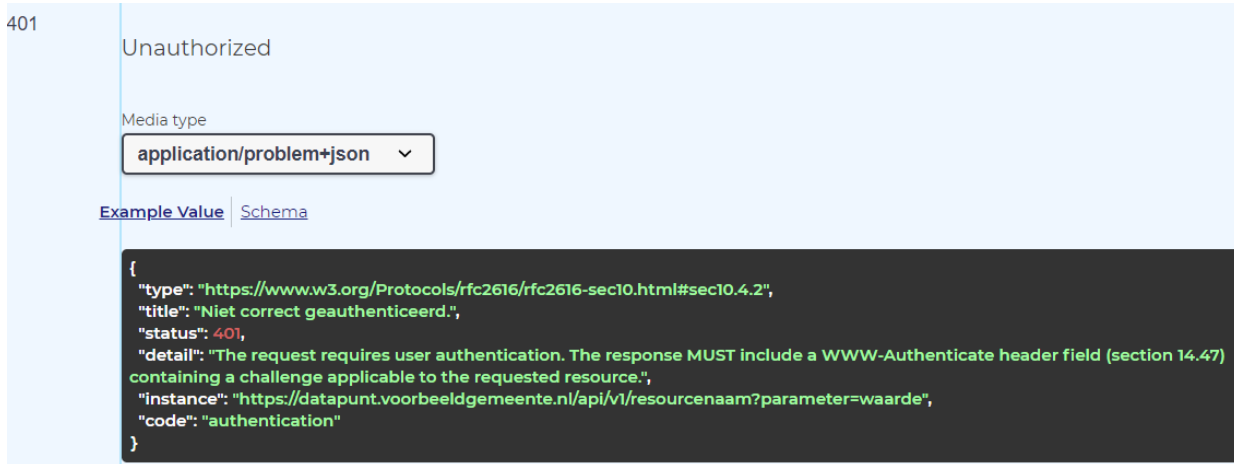
- **Status:** bevat de originele HTTP status code, gespecificeerd in RFC7231 en RFC4918 . Dit veld wordt verplicht want het is een vereist element in de error handling.
- **Conclusie:** attribuut '*Status*' verplicht en bevat de gestandaardiseerde HTTP status code



RFC7807 implementatie *Detail*

- **Status:** geeft gedetailleerde informatie zodat de consumer in staat wordt gesteld de error te corrigeren en tot een mogelijke oplossing te komen. Een optioneel, human readable veld dat vrij in te vullen is. Voertaal is UK Engels.

Voorbeeld¹:



401
Unauthorized

Media type
application/problem+json ▾

[Example Value](#) | [Schema](#)

```
{
  "type": "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.2",
  "title": "Niet correct geauthenticeerd.",
  "status": 401,
  "detail": "The request requires user authentication. The response MUST include a WWW-Authenticate header field (section 14.47) containing a challenge applicable to the requested resource.",
  "instance": "https://datapunt.voorbeeldgemeente.nl/api/v1/resource naam?parameter=waarde",
  "code": "authentication"
}
```

➤ **Conclusie:** attribuut 'Detail' optioneel en bevat gedetailleerd beschrijving van fout en mogelijke oplossing



RFC7807 implementatie *Instance*

- **Instance:** de URI die de URL bevat van de aanroep die de fout heeft veroorzaakt. Ook wel de fout-instantie. Dit is een optioneel veld.

Voorbeeld¹:



403

Forbidden

Media type

application/problem+json ▾

[Example Value](#) | [Schema](#)

```
{
  "type": "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.4",
  "title": "U bent niet geautoriseerd voor deze operatie.",
  "status": 403,
  "detail": "The server understood the request, but is refusing to fulfill it.",
  "instance": "https://datapunt.voorbeeldgemeente.nl/api/v1/resource naam?parameter=waarde",
  "code": "autorisation"
}
```

➤ **Conclusie:** attribuut 'Instance' optioneel en bevat de url van het request die tot de fout heeft geleid



Extensie veld Code

- **Code:** een aantal NL-overheids APIs bevat in de error response het veld *Code*. Geonovum beschrijft dit attribuut als 'een systeemcode die het type fout aangeeft'¹. Met *Code* kan naast de omschrijving in *Title*, een extra classificatie van de fout worden meegegeven. Een optioneel veld, dat indien toegepast gebruik dient te maken van de vastgestelde set systeemcodes².

Voorbeeld¹:

Foutsituatie	Status	Voorbeeld Title (NL)	Code
Geen parameter is meegegeven	400	Geef tenminste één parameter op	paramsRequired
Verplichte parameter(combinatie	400	Minimale combinatie van parameters moet worden opgegeven	paramsCombination
Niet toegestane parametercombinatie	400	De combinatie van opgegeven parameters is niet toegestaan	unsupportedCombi
Parametervalidatie	400	Een of meerdere parameters zijn niet correct	paramsValidation

➤ **Conclusie:** attribuut 'Code' optioneel en bevat vastgestelde systeemcodes



RFC7807 implementatie *invalidParams*

- **invalidParams:** Wanneer de fout is veroorzaakt door fouten in requestparameters (of request body), wordt *invalidParams* gevuld met details over elke foute parameter. In de RFC7807 standaard wordt hiervan een concreet voorbeeld gegeven¹, maar wordt dit niet nader gespecificeerd. Bij de NL-overheids API werkgroepen is op verschillende manier uitgewerkt². Gemeenschappelijk komt men tot het volgende inzicht, waarbij element *invalidParams* een array bevat dat uit de volgende onderdelen bestaat:

Veldnaam	type	cardinaliteit	omschrijving
name	string	verplicht	Naam van het veld met de ongeldige gegevens
code	string	optioneel	Systeemcode die het type fout aangeeft
reason	string	verplicht	Uitleg wat er precies fout is met de gegevens



RFC7807 implementatie *invalidParams*

Wanneer er fouten zitten op meerdere parameters, wordt er per validatiefout een *invalidParams* instantie opgenomen in het antwoord. Alle fouten worden dus teruggegeven. Voorbeeld¹:

```
GET {{baseUrl}}/ingeschrevenpersonen?verblijfplaats_huisnummer=ABC&burgerservicenummerzzz=999990482,999991905,999990421
```

```
{
  "invalidParams": [
    {
      "name": "verblijfplaats_huisnummer",
      "code": "integer",
      "reason": "Waarde is geen geldige integer."
    },
    {
      "name": "burgerservicenummerzzz",
      "code": "unknownParam",
      "reason": "Parameter is niet verwacht."
    }
  ],
  "type": "https://docs.microsoft.com/en-us/dotnet/api/system.net.httpstatuscode?#System_Net_HttpStatusCode_BadRequest",
  "title": "Een of meerdere parameters zijn niet correct.",
  "status": 400,
  "detail": "De foutieve parameter(s) zijn: verblijfplaats_huisnummer, burgerservicenummerzzz.",
  "instance": "https://www.haalcentraal.nl/haalcentraal/api/brp/ingeschrevenpersonen?verblijfplaats_huisnummer=ABC&burgerservicenummerzzz=999990482,999991905,999990421",
  "code": "paramsValidation"
}
```



RFC7807 implementatie *invalidParams*

- **invalidParams:** Merk op dat ook hier gebruik wordt gemaakt van optioneel attribuut *code* als systeem code om het type fout aan te geven. Dit is een optioneel veld dat gebruik maakt van een vaste set waarden¹. Indien dit veld wordt opgenomen in het *invalidParams* array, dan worden deze vaste waarden toegepast. De velden *name* en *reason* zijn verplicht waarbij *reason* vrij in te vullen is. Voertaal is UK Engels.
- **Conclusie:** attribuut '*invalidParams*' optioneel en indien toegepast bevat de verplichte array onderdelen '*name*' en '*reason*' met beschrijvende waarden



RFC7807 eindvoorstel

We komen tot het volgende eindvoorstel:

RESPONSE SCHEMA: `application/problem+json`

type required	string (Type) <code>non-empty</code> URI referentie naar het type fout, bedoeld voor developers
code	string (Code) Systeemcode die het type fout aangeeft
title required	string (Title) <code>non-empty</code> Generieke titel voor het type fout
status required	integer (Status) De HTTP status code
detail	string (Detail) Extra informatie bij de fout, indien beschikbaar
instance	string (Instance) URI met referentie naar dit specifiek voorkomen van de fout. Deze kan gebruikt worden in combinatie met server logs, bijvoorbeeld.

Waarbij *type*, *title* en *status* de verplichte velden zijn. Verder is er een samenhang tussen deze velden. Indien *type* de default waarde 'about:blank' bevat, dient *title* de overeenkomstige HTTP standaard omschrijving te bevatten¹. Voorbeeld:

```
{
  "type": "about:blank",
  "title": "Not found",
  "status": 404,
}
```

Als *type* wel is gedefinieerd, dient *title* juist **niet** de overeenkomstige HTTP standaard omschrijving te bevatten. Voorbeeld:

```
{
  "type": "https://example.com/probs/no-account-details",
  "title": "No account details found",
  "status": 404,
}
```

1. Bronnen:



RFC7807 eindvoorstel

We komen tot het volgende eindvoorstel:

invalidParams  Array of objects (FieldValidationError)

Array [
name required	string (Name) non-empty Naam van het veld met ongeldige gegevens
code	string (Code) Systeemcode die het type fout aangeeft
reason required	string (Reason) non-empty Uitleg wat er precies fout is met de gegevens

Indien gebruik wordt gemaakt van *invalidParams*, dienen de velden *name* en *reason* een waarde te bevatten. Veld *name* bevat de parameter naam waar de fout zit, veld *reason* de omschrijving van de fout.

Het verplicht stellen van *Type*, *Title* en *Status* en vinden we ook terug bij het definiëren van nieuwe problem types¹ en bij Zalando².

1. <https://datatracker.ietf.org/doc/html/rfc7807#section-4>
2. <https://github.com/zalando/restful-api-guidelines/pull/3>





RFC7807 internationaal

- Zalando ondersteunt RFC7807. Clients dienen in de Accept header aan te geven dat response codering volgens RFC7807 moet plaatsvinden: <https://opensource.zalando.com/restful-api-guidelines/#176>
- RFC7807 compliant publieke internationale APIs zijn te vinden via: <https://github.com/APIs-guru/openapi-directory/search?q=application%2Fproblem%2Bjson>
- Swedbank Pay: <https://developer.swedbankpay.com/checkout/v2/features/technical-reference/problems>

- :



Handige links

- Voorbeeld foutresponse met verwijzing naar RFC7231 voor attribuut Type: <https://github.com/Geonovum/KP-APIs/issues/292>
- Voorbeeld foutresponse met verwijzing naar Microsoft uitleg voor attribuut Type: <https://github.com/VNG-Realisatie/Haal-Centraal-BRP-bevragen/issues/804>
- Dimitri van Hees (Kadaster) over RFC7807: <https://geoforum.nl/t/formaat-foutafhandeling-input-validatie-api-50/1848/2>
- Kadaster APIs met RFC7807 toegepast: <https://developer.overheid.nl/apis?organisatie=00000001802327497000>
- Bevragen Ingeschreven Personen (optionele attributen problem type): <https://vng-realisatie.github.io/Haal-Centraal-BRP-bevragen/swagger-ui>
- Github Geonovum issue met ZGW APIs en verplichte attributen: <https://github.com/Geonovum/KP-APIs/issues/289>
- REST API Error Handling - Problem Details Response: <https://blog.restcase.com/rest-api-error-handling-problem-details-response/>
- Overzicht foutcodes en RFC's: <https://www.iana.org/assignments/http-status-codes/http-status-codes.xhtml>
- Asbjørn Ulsberg on RFC7807 What's Your Problem?: <https://www.youtube.com/watch?v=ZWlUn5BHDBc>
- Mooi overzicht HTTP status codes en uitleg: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>
- Foutafhandeling feature file van VNG: <https://github.com/VNG-Realisatie/Haal-Centraal-common/blob/master/features/foutafhandeling.feature>
- Concrete response codes behorend bij feature file: <https://github.com/VNG-Realisatie/Haal-Centraal-BAG-bevragen/pull/501>
- RFC7807 dependencies: <https://greenbytes.de/tech/webdav/rfc7807.references.svg>



Appendix A: waarden voor *Title*

Status code	Omschrijving	Title
200	OK	The request succeeded
201	Created	The request successfully created a resource
204	No Content	The server successfully fulfilled the request but returns no response body
400	Bad Request	The request cannot be fulfilled due to bad syntax
401	Unauthorized	User failed to authenticate properly
403	Forbidden	Authorization failed due to insufficient permissions
404	Not found	The specified resource does not exist
405	Method Not Allowed	The server does not support the requested HTTP method
406	Not Acceptable	The server does not support the client-request media type to return the response payload
409	Conflict	The request conflicts with current state of the target resource
415	Unsupported Media Type	The server does not support the request payload's media type
429	Too many requests	Too many requests. Blocked due to rate limiting
500	Internal Server Error	An internal server error has occurred
503	Service Unavailable	Service unavailable



**SAMEN WERKEN WE AAN EEN
BETROUWBARE EN DUURZAME
ENERGIEVOORZIENING
VOOR VANDAAG EN VOOR DE
TOEKOMST.**

**For more information, please contact:
CST-Integratie@enexis.nl**



ENEXIS
GROEP