Telemetry API Implementatie-handleiding

Versie: 1.7 10-01-2024

fudura

fudura

Versie historie

Versie	Datum	Wijziging
1.0		Initiële versie
1.1	31-10-2022	Shared-eindpunten toegevoegd (voor gemachtigden)
1.2	26-04-2023	URL van developer-portal gewijzigd (https i.p.v. http)
1.3	28-04-2023	GetSharedCustomerIds-eindpunt toegevoegd (voor gemachtigden)
1.4	27-07-2023	Informatie over gebruikte tijdstippen toegevoegd
1.5	13-10-2023	CustomerId query parameter toegevoegd (voor gemachtigden)
1.6	30-11-2023	Styling
1.7	10-01-2024	GetDetailedMeteringPoints-eindpunt toegevoegd
1.8	08-01-2025	Fout verwijderd over afhandeling getchanges endpoint en algemene stijling en verbeteringen

fudura

Inhoudsopgave

Aan de slag 4~~5~~
Inleiding 4~~5~~
Ontwikkelaarsportaal 4~~5~~
Interactieve API-specificatie 4~~5~~
Tijdstippen 7~~8~~
API-sleutels 7~~8~~
Gebruik 7~~8~~
Telemetrieaanvragen 8~~9~~
Beoogd gebruik 8~~9~~
Implementatie 8~~9~~
1. Meetpunten en kanalen verkrijgen 8~~9~~
2. Metagegevens van het kanaal ophalen (optioneel) 9~~10~~
3. Telemetrie verkrijgen 9~~10~~
Aanvragen voor wijzigingsfeed 10~~11~~
Beoogd gebruik 10~~11~~
Implementatie 10~~11~~
1. Meetpunten en kanalen verkrijgen 10~~11~~
2. Metagegevens van het kanaal ophalen (eerste keer) 11~~12~~
3. Telemetrie verkrijgen (eerste keer) 11~~12~~
4. Wijzigingen ophalen 12~~13~~
5. Wijzigingen verwerken 13~~14~~
Data opvragen als tussenpersoon 14~~15~~
Introductie 14~~15~~
Klantnummers verkrijgen 14~~15~~
Gegevens opvragen namens een klant 14~~15~~
Deprecated eindpunten 14~~15~~

Aan de slag

Inleiding

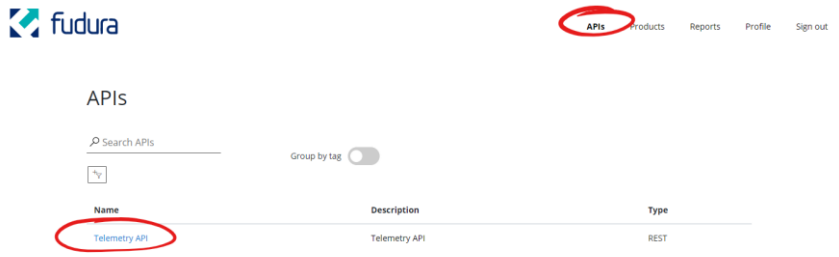
De Telemetrie-API van Fudura biedt toegang tot telemetrie en informatie over beschikbare telemetriestromen. Je kunt niet alleen actuele telemetrie opvragen, maar ook op de hoogte blijven van nieuwe en bijgewerkte telemetrie met een wijzigingsfeed. In dit document wordt uitgelegd hoe je de API kunt gebruiken.

Ontwikkelaarsportaal

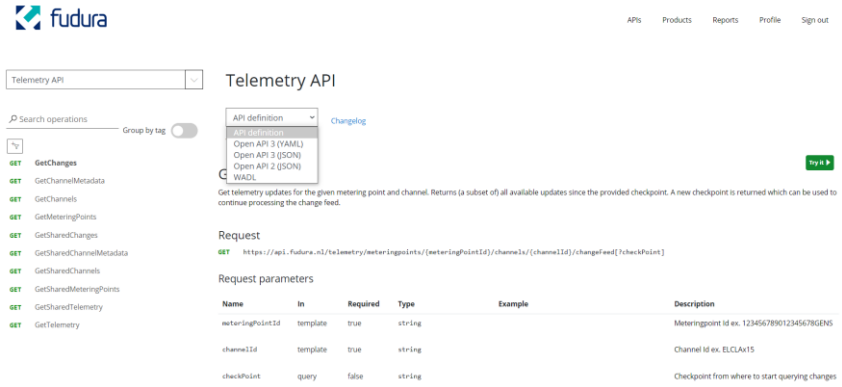
Om te beginnen heb je een account nodig voor Fudura's Ontwikkelaarsportaal (https://developers.fudura.nl). Dit account kun je verkrijgen via de afdeling Verkoop / Klantenservice. Je ontvangt dan op het opgegeven e-mailadres een uitnodiging om een account aan te maken.
Nadat je een wachtwoord hebt aangemaakt en bent ingelogd op jouw zojuist aangemaakte account, heb je toegang tot de specificatie voor de telemetrie-API. Je kunt dan de API-sleutels ophalen en een testaanvraag uitvoeren op de API.

Interactieve API-specificatie

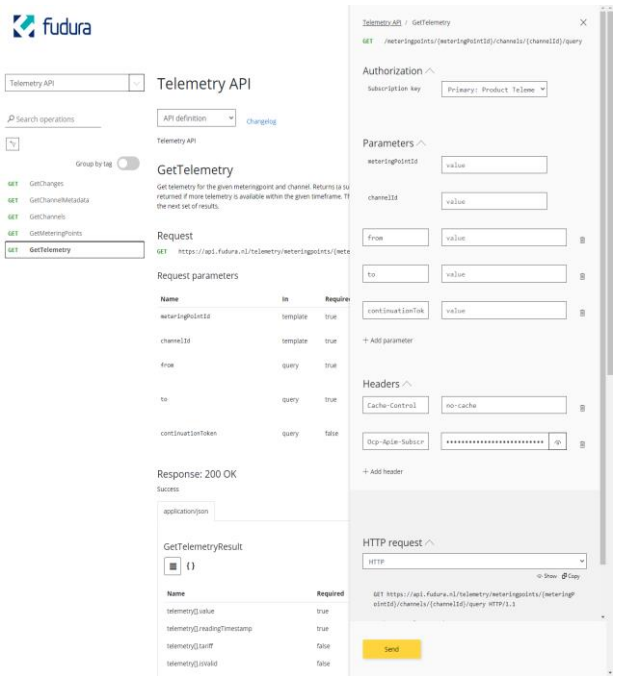
De specificatie voor de Telemetrie-API is te vinden onder “APIs” → “Telemetry API”.



De specificatie wordt zowel in een interactief, grafisch formaat als in een downloadbaar document aangeboden. Je kunt het gewenste formaat selecteren in de vervolgkeuzelijst:



Met de grafische interface kun je niet alleen door de eindpunten bladeren, maar ook rechtstreeks vanuit de browser aanvragen verzenden. De knop “try it” (aan de rechterkant) opent een nieuwe interface waarin je een aanvraag kunt uitproberen.



Nadat je de aanvraag hebt verzonden, kun je de geheimen onthullen ("Reveal secrets") of het volledige verzoek en de gebruikte headers met slechts één klik kopiëren ("Copy").



Wanneer je de vervolgkeuzelijst bij de HTTP-aanvraag opent, kun je voorbeelden krijgen van hoe je de aanvraag kunt implementeren in code voor de populairste programmeertalen.

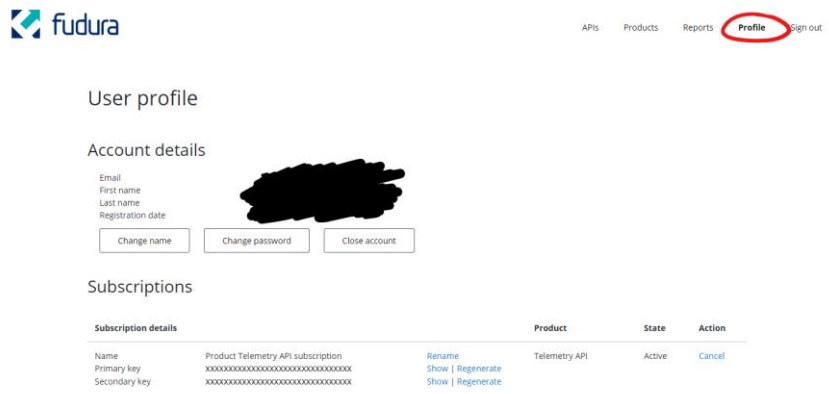


Tijdstippen

Alle tijdstippen die gebruikt worden in de Telemetry API zijn in UTC, in het "Zulu-time"-formaat (bijvoorbeeld: 2022-07-15T13:20:05Z). Voor elk verbruik is de ReadingTimestamp het tot-tijdstip. Bijvoorbeeld: voor het elektriciteitsverbruik van 23-07-25T15:45Z tot 2023-07-25T16:00Z zal de ReadingTimestamp 2023-07-25T16:00Z zijn.

API-sleutels

De API-sleutels zijn te vinden onder "Profile", onder "Subscriptions". Hier kun je de sleutels bekijken en opnieuw genereren**. Let op: Bewaar deze sleutels op een veilige plaats! Gebruik geen sleutel voor versiebeheer en deel deze niet met anderen. Vermoed je dat er een sleutel is uitgelekt? Genereer deze dan opnieuw en werk de sleutel in jouw toepassing bij.**



Gebruik

Er zijn meerdere manieren om de Telemetrie-API te gebruiken. Welke optie je moet kiezen, hangt af van uw gebruiksscenario:

- Wil je alleen toegang tot je telemetrie om deze te vergelijken of te weergeven in jouw eigen gebruikersportaal? Volg dan de instructies in hoofdstuk 2 'Telemetrieaanvragen'.

- Wil je jouw systemen up-to-date houden met bijgewerkte en nieuwe telemetrie? Gebruik dan de wijzigingsfeed zoals beschreven in hoofdstuk 3 'Aanvragen voor wijzigingsfeed'.

fudura

Telemetrieaanvragen

Beoogd gebruik

In dit hoofdstuk wordt uitgelegd hoe je een integratie met de Telemetrie-API kunt implementeren om telemetrie op te vragen. De hieronder beschreven methode is speciaal bedoeld voor query's op aanvraag, of voor gebruikssituaties waarin alleen de huidige momentopname van telemetrie nodig is. Moet het doelsysteem worden bijgewerkt met wijzigingen of nieuwe telemetrie? Maak dan gebruik van de wijzigingsfeed zoals beschreven in hoofdstuk 3 'Aanvragen voor wijzigingsfeed'.

Implementatie

Om te beginnen moet je enkele belangrijke concepten begrijpen:

- **EAN** Een 18-cijferig identificatienummer voor een (netwerk-)aansluiting.
- **MeteringPoint** Staat voor een gemeten aansluiting.
- **MeteringPointId** Een identificatienummer voor een meetpunt, bestaande uit een EAN, eventueel gevolgd door een toevoeging.
- **Channel** Vertegenwoordigt een kanaal met één enkel type meting.
- **ChannelId** Identificatienummer van een kanaal.

Het opvragen van telemetrie gebeurt via het eindpunt `/telemetry/meteringpoints/{meteringPointID}/channels/{channelId}/query`. Voordat je een verzoek kunt sturen naar dit eindpunt, moet er een meetpunt en kanaal gekozen worden waarvoor de meetdata wordt opgevraagd. Het proces om deze informatie te verkrijgen en uiteindelijk de telemetrie op te vragen, bestaat uit de volgende stappen:

1. Meetpunten en kanalen verkrijgen

De eerste stap is het opvragen van de beschikbare meetpunten, kanalen en de geautoriseerde periodes waarvoor meetdata kan worden uitgevraagd. Dit wordt gedaan met behulp van het eindpunt `/telemetry/detailed-meteringpoints`, dat in het ontwikkelaarsportaal wordt vermeld als de operatie **GetDetailedMeteringPoints**. Dit eindpunt geeft informatie over welke EAN's voor je beschikbaar zijn, welke meetpunten beschikbaar zijn onder elke EAN en welke tijdsperiode(n) je mag opvragen.
Als de lijst lang is, wordt deze gesplitst. Met de vervolgtoken (continuation token) die wordt teruggegeven kan het volgende resultaat worden opgevraagd.

fudura

2. Metagegevens van het kanaal ophalen (optioneel)

Met de resultaten uit stap 1 is er een lijst met kanalen waarvoor meetdata kan worden opgevraagd, maar nog niet veel context over de gegevens die een bepaald kanaal heeft. Het eindpunt `/telemetry/meteringpoints/{meteringPointId}/channels/{channelId}` (in het ontwikkelaarsportaal vermeld als de operatie **GetChannelMetaData**) kan worden gebruikt om metagegevens over het kanaal op te halen. Dit omvat o.a. het gegevenstype, het meetinterval, de beschrijving en de tijdstempel voor de eerste en laatste lezing.

3. Telemetrie verkrijgen

Als een meetpunt en een kanaal gekozen zijn, kan de telemetrie worden uitgevraagd met het eindpunt `/telemetry/meteringpoints/{meteringPointID}/channels/{channelID}/query` (in het ontwikkelaarsportaal vermeld als de operatie **GetTelemetry**). Bij het opvragen van grotere tijdsperiodes kan slechts een subset van de beschikbare telemetrie worden geretourneerd. In dat geval bevat het antwoord ook een vervolgtoken (continuation token). Dit token moet worden gebruikt om dezelfde query te herhalen, waarbij het token als queryparameter wordt toegevoegd. Hiermee wordt de query voortgezet en wordt de volgende set telemetrie geretourneerd. Zodra er geen vervolgtoken meer wordt geretourneerd, is de query voltooid.
De omvang van het antwoord is niet gegarandeerd, maar moet redelijk consistent zijn. Het aantal API-aanroepen dat nodig is om een query te voltooien zal dus samenhangen met de grootte van het opgevraagde tijdsbereik en het meetinterval van het opgevraagde kanaal.

Opmerkingen:

- Het eindpunt van de query vereist een bepaald datumbereik, dat binnen de autorisatieperiode voor het gegeven meetpunt valt. Anders levert het een *ongeautoriseerd* resultaat op.
- De parameters *van* en *tot* moeten volgens ISO 8601 als "Zulu-tijd" worden opgegeven (bijvoorbeeld: 2022-07-15T13:20:05Z).
- Zowel de queryparameter *from* als *to* is inclusief.

fudura

Aanvragen voor wijzigingsfeed

Beoogd gebruik

In dit hoofdstuk wordt uitgelegd hoe je een integratie met de Telemetrie-API implementeert, die wordt bijgewerkt met wijzigingen of nieuwe telemetrie. Raadpleeg voor query's op aanvraag hoofdstuk 2 'Telemetrieaanvragen'.

Implementatie

Om te beginnen moet je enkele belangrijke concepten begrijpen:

- **EAN** Een 18-cijferig identificatienummer voor een (netwerk-)aansluiting.
- **MeteringPoint** Staat voor een gemeten aansluiting.
- **MeteringPointId** Een identificatienummer voor een meetpunt, bestaande uit een EAN, eventueel gevolgd door een toevoeging.
- **Channel** Vertegenwoordigt een kanaal met één enkel type meting.
- **ChannelId** Identificatienummer van een kanaal.
- **Change feed checkpoint** Een token dat één enkel punt vertegenwoordigt in de reeks wijzigingen in de wijzigingsfeed.

Het opvragen van wijzigingen gebeurt via het eindpunt `/telemetry/meteringpoints/{meteringPointId}/channels/{channelId}/changes`. Voordat je een verzoek kunt sturen naar dit eindpunt, moet er een meetpunt en kanaal gekozen worden waarvoor de meetdata wordt opgevraagd. Daarnaast ~~....en de initiële telemetrie lade~~ moet je de telemetrie initieel laden~~n.~~ Het proces om deze informatie te verkrijgen ~~en uiteindelijk te wijzigingen aan te brengen,~~ bestaat uit de volgende stappen:

Met opmerkingen [Hv1]: Wat had deze zin moeten zijn?

Met opmerkingen [Av2R1]: Volgens mij hoeft deze zin er niet te staan. Het is logisch om een initiele import te doen van data, maar volgens mij als je de getchanges opvraagt zonder token dan krijg je alles wat er is, heb het nog even bij Dajo gevraagd. Vooralsnog die zin weggehaald

1. Meetpunten en kanalen verkrijgen

De eerste stap is het opvragen van de beschikbare meetpunten, kanalen en de geautoriseerde periodes waarvoor meetdata kan worden uitgevraagd. Dit wordt gedaan met behulp van het eindpunt `/telemetry/detailed-meteringpoints`, dat in het ontwikkelaarsportaal wordt vermeld als de operatie **GetDetailedMeteringPoints**. Dit eindpunt geeft informatie over welke EAN's voor je beschikbaar zijn, welke meetpunten beschikbaar zijn onder elke EAN en welke tijdsperiode(n) je mag opvragen.
Als de lijst lang is, wordt deze gesplitst. Met de vervolgtoken (continuation token) die wordt teruggegeven kan het volgende resultaat worden opgevraagd.

2. Metagegevens van het kanaal ophalen (eerste keer)

Met de resultaten uit stap 1 is er een lijst met kanalen waarvoor meetdata kan worden opgevraagd, maar nog niet veel context over de gegevens die een bepaald kanaal heeft. Het eindpunt `/telemetry/meteringpoints/{meteringPointId}/channels/{channelId}` (in het ontwikkelaarsportaal vermeld als de operatie **GetChannelMetaData**) kan worden gebruikt om metagegevens over het kanaal op te halen. Dit omvat o.a. het gegevenstype, het meetinterval, de beschrijving en de tijdstempel voor de eerste en laatste lezing. Het resultaat bevat ook een eigenschap **LastChangeFeedCheckPoint**. De doeltoepassing moet deze waarde onthouden en deze doorgeven bij de eerste aanroep van het **GetChanges-eindpunt**.

3. Telemetrie verkrijgen (eerste keer)

Als een meetpunt en een kanaal gekozen zijn, kan de telemetrie worden uitgevraagd met het eindpunt `/telemetry/meteringpoints/{meteringPointID}/channels/{channelID}/query` (in het ontwikkelaarsportaal vermeld als de operatie **GetTelemetry**). Bij het opvragen van grotere tijdsperiodes kan slechts een subset van de beschikbare telemetrie worden geretourneerd. In dat geval bevat het antwoord ook een vervolgtoken (continuation token). Dit token moet worden gebruikt om dezelfde query te herhalen, waarbij het token als queryparameter wordt toegevoegd. Hiermee wordt de query voortgezet en wordt de volgende set telemetrie geretourneerd. Zodra er geen vervolgtoken meer wordt geretourneerd, is de query voltooid.
De omvang van het antwoord is niet gegarandeerd, maar moet redelijk consistent zijn. Het aantal API-aanroepen dat nodig is om een query te voltooien zal dus samenhangen met de grootte van het opgevraagde tijdsbereik en het meetinterval van het opgevraagde kanaal.

Opmerkingen:

- Het eindpunt van de query vereist een bepaald datumbereik, dat binnen de autorisatieperiode voor het gegeven meetpunt valt. Anders levert het een *ongeautoriseerd* resultaat op.
- De parameters ~~*van*~~ *from* en ~~*tot*~~ *to* moeten volgens ISO 8601 als "Zulu-tijd" worden opgegeven (bijvoorbeeld: 2022-07-15T13:20:05Z).
- De queryparameter *from* is exclusief.
- De queryparameter *to* is inclusief.

Met opmerkingen [Av3]: De parameters vertalen lijkt me niet zo nuttig

fudura

4. Wijzigingen ophalen

Nu de historische gegevens zijn opgehaald, is het tijd om het **GetChanges**-eindpunt te gebruiken om op de hoogte te blijven van nieuwe telemetrie. Het **GetChanges**-eindpunt kan worden aangeroepen met de parameters **meteringPointId** en **channelId**. Gebruik voor de **eerste** aanroep van dit eindpunt het controlepunt dat eerder via het **GetChannelMetaData**-eindpunt is opgehaald.

Het antwoord zal alle wijzigingen bevatten die sinds het controlepunt in de telemetrie zijn aangebracht voor deze **meteringPointId** en **channelId**. Dit omvat zowel nieuwe als bijgewerkte meetwaarden. Het antwoord zal ook een eigenschap **checkPoint** bevatten als er meer gegevens beschikbaar zijn. Er kunnen nog meer aanvragen voor dit eindpunt worden gedaan, totdat de eigenschap **checkPoint** niet meer in het antwoord voorkomt. Dat betekent dat je weer bij bent met de meest recente telemetrie.

Onder wijzigingen worden alle aanpassingen aan waarden verstaan, dit zijn:

- Nieuwe meetwaardes
- Correctie van een meetwaarde
- Aangepast validatieresultaat

Zorg ervoor dat de doeltoepassing de laatste checkpoint-waarde onthoudt die in de aanvraag was gebruikt, om op een later tijdstip vanaf dit controlepunt verder te gaan.

Het **GetChanges**-eindpunt retourneert alleen recente wijzigingen in de telemetrie. Dat houdt het volgende in:

- Wijzigingen welke langer dan 10 dagen geleden zijn aangebracht worden niet geretourneerd.

Wijzigingen welke korter dan 10 dagen geleden zijn aangebracht worden wel geretourneerd. Ook wanneer deze wijzigingen meetmomenten verder in het verleden betreffen. ~~Wijzigingen die in het verre verleden zijn aangebracht worden *niet* geretourneerd.~~

-
- ~~Recente wijzigingen in de telemetrie op datums in het verleden worden *wel* geretourneerd.~~

Een controlepunt is **slechts 10 dagen geldig**. Na deze periode worden controlepunten door het systeem verwijderd. Er wordt **geen waarschuwing** gegeven wanneer een ongeldig controlepunt wordt gebruikt. In plaats daarvan bevat het antwoord dan gegevens voor de controlepunten die nog steeds bestaan voor uw telemetriegegevens. Daarom raden we aan om het **GetChanges**-eindpunt minimaal dagelijks op te vragen, of vaker indien nodig.

heeft opmaak toegepast: Lettertype: 11 pt

heeft opmaak toegepast: Lettertype: 11 pt

Met opmaak: Lijstalinea, Met opsommingstekens + Niveau: 1 + Uitgelijnd op: 0,63 cm + Inspringen op: 1,27 cm

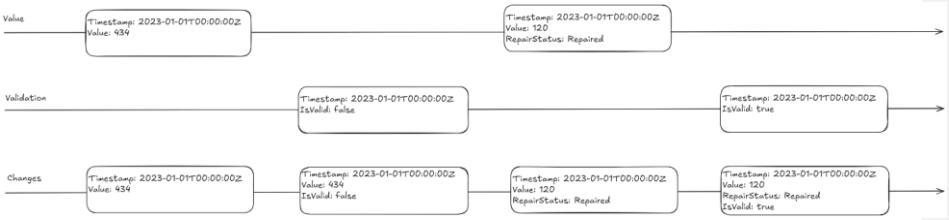
Met opmerkingen [Hv4]: Moet hier nog een indicatie bij?

fudura

5. Wijzigingen verwerken

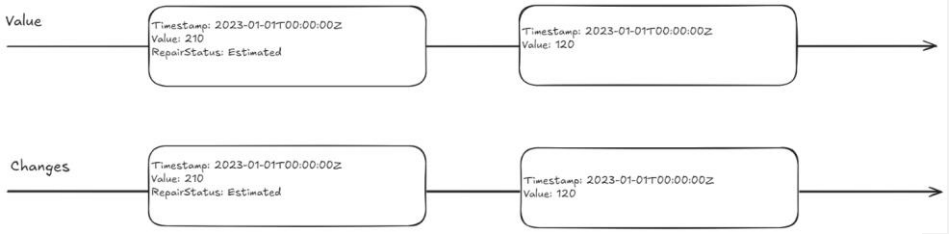
Het **GetChanges**-eindpunt krijgt de data vanuit een wijzigingsfeed waarin alle updates over telemetrie voor het opgegeven kanaal worden vastgelegd. Dit betekent dat de resultaten meerdere updates van dezelfde meting (tijdstempel) kunnen bevatten. Alle metingen of updates in het resultaat zijn chronologisch geordend.

Er zijn meerdere redenen waarom er meerdere updates kunnen zijn op eenzelfde tijdstip, zoals correcties en validatie. Aangezien het validatieproces asynchroon is, zal het validatieresultaat altijd een latere update zijn van de meting. Het volgende diagram toont een scenario waarin een enkele meting wordt gecollecteerd en gevalideerd, waarna een correctie wordt aangebracht op die meting en ook wordt gevalideerd.



Een ander voorbeeld zou een schatting kunnen zijn, waarbij de daadwerkelijke meting later wordt gecollecteerd bij de meter.

Met opmerkingen [Hv5]: Verwarrende zin. Wat wordt hiermee bedoeld?



Om de metingen te verwerken, moet elke volgende meting voor een gegeven tijdstip worden beschouwd als een **volledige update** die alle beschikbare eigenschappen voor die meting bevat.

Met opmerkingen [Hv6]: Ik mis hier een beetje wat de klant er concreet mee moet. Dit lijkt meer context dan daadwerkelijke instructies

Met opmerkingen [Av7R6]: Ik snap je, het gaat erover dat statussen en dergelijke ook geupdate kunnen zijn. Ik stel echter voor om dit voor nu zo te laten.

Data opvragen als tussenpersoon

Introductie

Een Fudura-klant kan een tussenpersoon machtigen om namens de klant de gegevens op te vragen. Zodra deze goedkeuring is verwerkt door de klantenservice van Fudura, kan de gemachtigde tussenpersoon de gegevens van de klant opvragen met het API-account van de tussenpersoon. Er is geen noodzaak om API-inloggegevens of autorisatiesleutels te delen. Het enige wat de tussenpersoon hoeft te weten, is het klantnummer.

Klantnummers verkrijgen

Een tussenpersoon kan het **GetSharedCustomerIds**-eindpunt gebruiken om een lijst met klantnummers van de klanten, voor wie de tussenpersoon gemachtigd is, op te halen.

Gegevens opvragen namens een klant

Alle eindpunten die in de vorige hoofdstukken zijn beschreven, hebben een optionele customerid-queryparameter. Deze parameter kan als volgt worden gebruikt:

- Laat leeg om uw eigen gegevens op te halen.
- Gebruik het klantnummer van de klant die jou heeft gemachtigd om namens hen gegevens op te halen.
- Gebruik een ***** om alle gegevens op te halen waar je toegang toe hebt, ongeacht de klant.

‘Deprecated’ eindpunten

Een aantal eindpunten zijn vervallen (‘deprecated’), maar zijn wel nog beschikbaar om ‘backwards compatible’ te zijn. D.w.z. dat ze compatibel zijn met oudere protocollen. Maak gebruik van de voorgestelde alternatieven om zeker te zijn dat jouw applicatie ook nog functioneert in de volgende versies.

Deprecated eindpunt	Alternatief
GetSharedChanges	GetChanges + parameter customerId
GetSharedChannelMetadata	GetChannelMetadata + parameter customerId
GetSharedChannels	GetChannels + parameter customerId
GetSharedMeteringPoints	GetMeteringPoints + parameter customerId
GetSharedTelemetry	GetTelemetry + parameter customerId