

Daarnaast moeten alle berichten gedefinieerd en geserialiseerd worden in het Protobuf 3 formaat. Hierdoor zijn berichten compact en op veel verschillende platformen toepasbaar door beschikbare standaard implementaties.

De rest van dit document behandelt de verschillende functionele flows in de koppelvlakken, de inhoud van de verschillende berichten die nodig zijn voor deze flows en tenslotte worden de MQTT details behandeld die nodig zijn voor het versturen van berichten.

Granulariteit

De berichten in de koppelvlakken zijn onafhankelijk van technische implementaties. Berichten worden verstuurd naar de functionele eenheid haltesysteem, waarbij een haltesysteem altijd slechts één set reisinformatie kan tonen. In praktische termen: De berichten zijn gedefinieerd voor een gewone enkelvoudige halte, waarbij één technische installatie één lijst met passages toont op één of meer schermen. Er is dus maar één schermbeeld per haltesysteem eventueel verdeeld over of getoond op meerdere schermen.

Een knooppunt waar meerdere displays staan die verschillende informatie geven, maar aangestuurd worden door een centrale (stations)processor is voor de koppelvlakken een verzameling losse haltesystemen, die ieder apart communiceren met de centrale en die ieder een eigen status en configuratie hebben. De koppelvlakken hebben daarmee geen mogelijkheid om over de status van de stationsprocessor zelf te rapporteren of om functionaliteit op stationsniveau te ondersteunen.

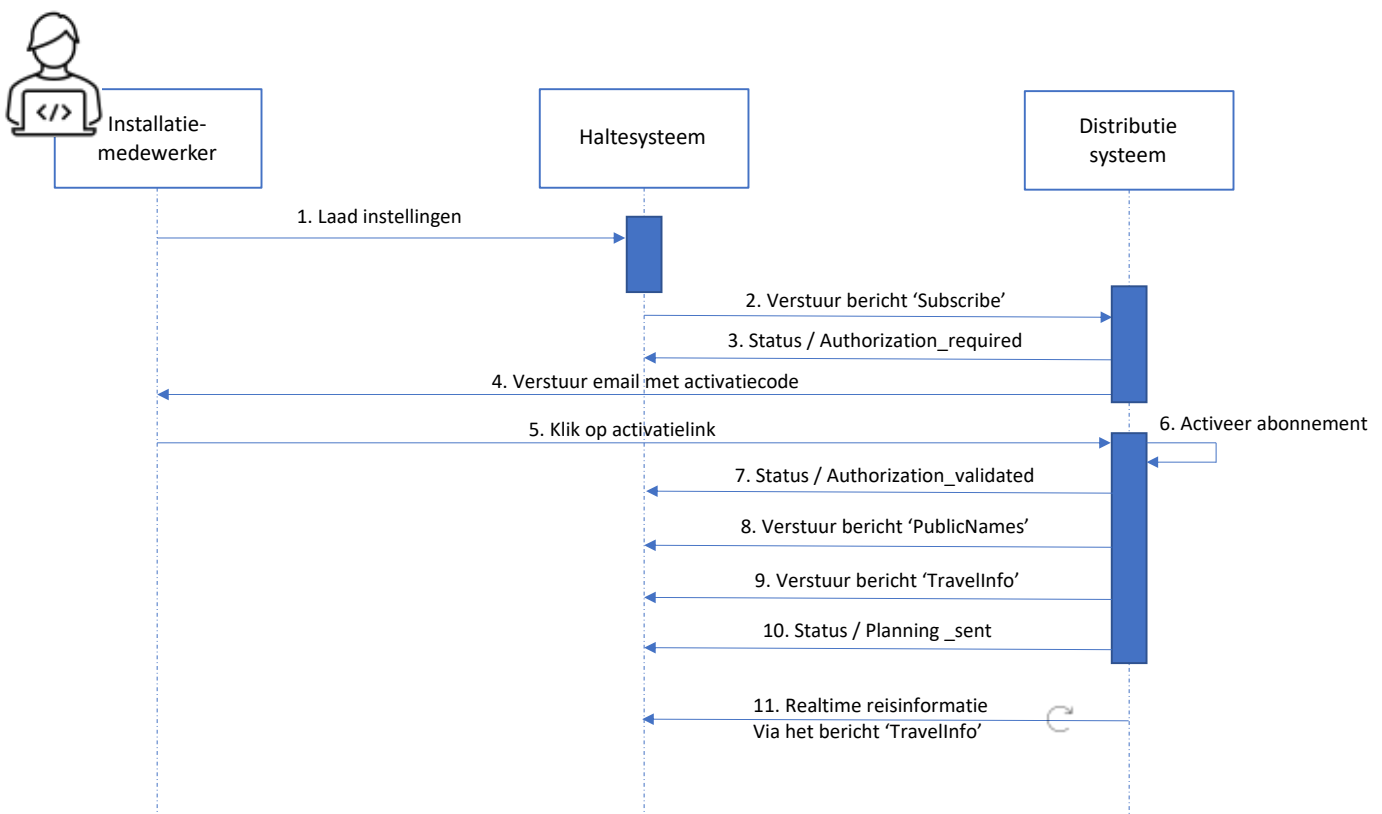
Flows

Aan- en afmelden

Het initiatief om een haltesysteem (weer) op te nemen in het systeem ligt bij het haltesysteem. Door het sturen van een aanmeldbericht abonneert het haltesysteem zich bij het distributiesysteem op de informatie van een of meer haltes, aangeduid met Quaycodes of Stopplacecodes zoals beschreven in het centraal halte bestand (CHB). Met hetzelfde aanmeldbericht maakt het haltesysteem zich ook bekend bij de dashboardsystemen. Die systemen gebruiken de Quay- of Stopplacecodes in het aanmeldbericht om praktische informatie zoals haltenaam en straatnaam op te halen uit het CHB.

Aanmelden gebeurt op basis van een client id (zie hoofdstuk MQTT voor details). Verder bevat het aanmeldbericht een aantal optionele velden, met deze velden wordt het distributiesysteem geïnformeerd over welke gegevens het haltesysteem wil ontvangen. Per veld kan gekozen worden voor: geen gegevens of alle wijzigingen.

Indien een haltesysteem zich voor de eerste keer aanmeldt, is het haltesysteem, naast het client id in de topicnaam, verplicht zich te identificeren met een emailadres van de beheer- of installatiepartij. Dit emailadres hoeft niet uniek te zijn. Ter autorisatie wordt door het distributiesysteem eenmalig per emailadres plus client id combinatie, een email gestuurd naar het emailadres van de leverancier van het haltesysteem ter verificatie. Dit emailbericht bevat een link waarmee het haltesysteem geautoriseerd wordt om data te ontvangen van het distributiesysteem. Na de verificatie wordt het haltesysteem direct actief. In het aanmeldbericht kan ook een omschrijving worden ingevuld zodat verschillende systemen uit elkaar kunnen worden gehouden.



Het aanmeldproces: eerste keer per haltesysteem

Op het aanmeldbericht wordt in het distributiesysteem een aantal controles uitgevoerd: alle verplichte velden moeten aanwezig zijn, Quay- of Stopplacecodes moeten bestaan in het CHB en het emailadres moet geldig zijn (per leverancier van Haltesystemen wordt een autorisatie-adres afgesproken en vastgelegd in het Distributiesysteem). Preconditie voor het aanmelden van een haltesysteem is dus dat de haltes waarover het haltesysteem informeert Quaycodes hebben in het CHB. Op alle aanmeldverzoeken wordt een antwoord teruggestuurd naar het haltesysteem met de status van de aanmelding (bericht 'SubscriptionResponse'). In geval van een foutief emailadres of ontbrekende informatie in het aanmeldbericht wordt de status 'REQUEST_INVALID' gecommuniceerd. Bij een stopcode die niet in CHB bekend is wordt de status 'STOP_INVALID' gecommuniceerd.

Na het doorlopen van de autorisatieprocedure stuurt het distributiesysteem de status 'AUTHORISATION_VALIDATED' om aan te geven dat de autorisatie gelukt is. Daarna wordt de meest actuele versie van de in het distributiesysteem bekende ritten opgehaald en eventueel geldige en toekomstige vrije teksten opgehaald en naar het haltesysteem verstuurd. Indien er ritten bekend zijn voor het betreffende haltesysteem, wordt na het versturen van de ritten de status 'PLANNING_SENT' gestuurd. Indien er geen geplande ritten zijn in de komende periode wordt de status 'NO_PLANNING' gecommuniceerd. In dit geval is het abonnement echter wel actief en kunnen er op elk moment, ritten dan wel vrije teksten naar het haltesysteem worden gestuurd.

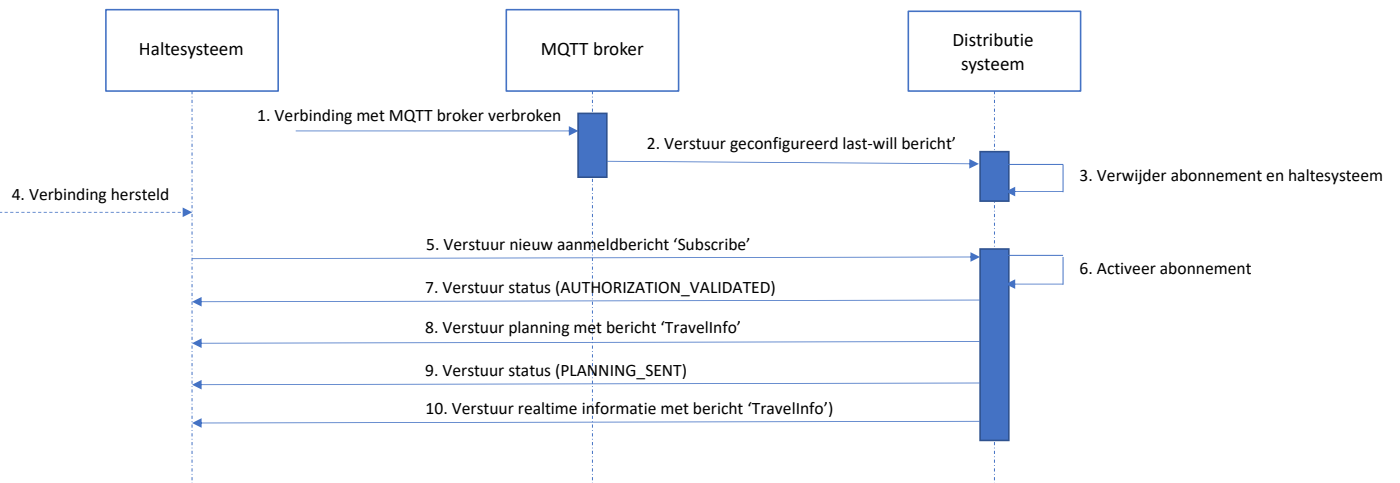
Dashboardsystemen kunnen zich ook abonneren op de aanmeldberichten en de antwoorden, zodat ze het aanmeldproces kunnen volgen en vastleggen.

Indien een haltesysteem zich twee keer aanmeldt zonder zich tussendoor af te melden, wordt dit door het distributiesysteem afgehandeld als een afmeldactie, gevolgd door opnieuw aanmelden. Hierbij zullen opnieuw de geplande en actuele ritinformatie alsmede de actieve en geplande vrije teksten worden verstuurd.

Afmelden

Bij het opzetten van de verbinding met de broker configureert het haltesysteem het bericht Unsubscribe als een zogenaamd 'last-will' bericht. Indien de verbinding tussen haltesysteem en de MQTT broker wegvalt, publiceert de MQTT broker dit last-will bericht, dat door het distributiesysteem en de dashboardsystemen wordt ontvangen. Indien het distributiesysteem een last-will/afmeldbericht ontvangt zal het geen informatie meer sturen naar het haltesysteem tot het opnieuw is aangemeld. De dashboardsystemen zullen de status van het haltesysteem veranderen naar offline.

Voor het last-will bericht wordt het bericht Unsubscribe gevuld met 'is_permanent' = false en De timestamp wordt gevuld met het huidige tijdstip.



Het afmeldproces, gevolgd door het opnieuw aanmelden van het al eerder geautoriseerde haltesysteem

Om eventuele race-condities te voorkomen is het van belang dat het haltesysteem zich ook abonneert op zijn eigen unsubscribe berichten (bijvoorbeeld verstuurd door de broker als last will bericht), en unsubscribe berichten van het distributiesysteem. Na het ontvangen van een eigen unsubscribe bericht of een unsubscribe bericht van het distributiesysteem, moet het haltesysteem een nieuw aanmeldbericht sturen.

Het is daarnaast ook mogelijk om een haltesysteem permanent af te melden door middel van het Unsubscribe bericht, met waarde 'is_permanent' = true en het (huidige) tijdstip van afmelden. Dit is van belang indien een haltesysteem wordt verwijderd, gedeactiveerd of wordt vervangen. Dit voorkomt bijvoorbeeld extra logging van gegevens over een haltesysteem dat defect is en maakt (keten)beheer makkelijker. Indien een haltesysteem is afgemeld met 'is_permanent' = true en het haltesysteem meldt zich later weer aan (met dezelfde client id), moet het opnieuw de autorisatie via email doorlopen.

Reisinformatie

Op basis van het selectiefilter wordt er elke nacht een lijst van ritten met hun passeertijden in unix-timestamps verstuurd van distributie- naar haltesysteem. Hierdoor is de afnemer niet afhankelijk van operationele dagen, slechts het begrip een tijdstip in UTC om naar een tijdzone van het haltesysteem te converteren is van belang, in dit geval: Europe/Amsterdam. De passertijden worden geïdentificeerd door het veld 'pass_time_hash', dat constant blijft gedurende een rit. Voor de unix-timestamp wordt de variant gebruikt waarvan de waarde 0 gelijkstaat aan de datum 1-1-1970 00:00:00 GMT.

Als een haltesysteem zich aanmeldt bij het distributiesysteem, stuurt het distributiesysteem voor 62 uur (parameter) passeertijden naar het haltesysteem. Met die gegevens kan het haltesysteem reisinformatie tonen ook als de verbinding met het distributiesysteem is wegvallen. Nadat de set met passeertijden voor 62 uur is gestuurd, stuurt het distributiesysteem met hetzelfde bericht de actuele passeertijden. Elke nacht wordt, weer met hetzelfde bericht, een set van 24 uur nieuwe passeertijden verstuurd voor de periode 38 – 62 uur.

De geplande en de actuele passeertijden zijn voor het haltesysteem volkomen hetzelfde en worden gekenmerkt door een zelfde hashleutel (pass_time_hash).

Er zijn gevallen mogelijk waarbij er binnen de geldige geplande periode geen geplande ritten zullen plaatsvinden (bijvoorbeeld in het weekend, evenementenhaltes, etc.) en er ook geen geldige vrije teksten

actief zijn op dat moment. Hierbij kan een geval ontstaan dat er geen reisinformatie voor het haltesysteem beschikbaar is. In dit geval wordt een statusbericht verstuurd naar het haltesysteem met status 'NO_PLANNING'

Standaard berichtafhandeling

Ieder bericht wordt gecodeerd via protocol buffers. Het uitwisselingsformaat zorgt ervoor dat datatypes horizontaal worden *gepacked*. Er zijn daardoor minder bits nodig als het datatype niet volledig wordt gebruikt. Daarnaast kiezen we binnen het protocol voor het principe van uitwisselen in kolommen. De meeste databases maken gebruik van rijen, 1 rij komt vaak overeen met de attributen van 1 object. Door niet in rijen, maar in kolommen uit te wisselen krijgen we een tweede optimalisatie: informatie die zich herhaalt (denk aan: bestemmingen en publieke lijnummers) staan dicht bij elkaar en kunnen slimmer worden gecomprimeerd. In een kolom georiënteerde database hebben alle kolommen hetzelfde aantal elementen. Een rij is op basis van de index samen te voegen. Element 1 uit kolom 1, komt daarmee overeen met element 1 uit kolom 2. Het haltesysteem heeft bij het aanmelden geselecteerd welke kolommen het wil ontvangen, alleen deze kolommen worden door het distributiesysteem gesynchroniseerd.

Het synchronisatieproces gaat uit van van een vaste identificatie voor een rit, dit in de vorm van een hash sleutel ('pass_time_hash'). Op deze manier is iedere passeertijd uniek identificeerbaar en overschrijfbaar. Op basis van het selectiefilter worden de actuele passeertijden naar het haltesysteem verstuurd.

```

passing_times {
  pass_time_hash: 1487911693
  pass_time_hash: 3375514872
  target_departure_time: 1496090100
  target_departure_time: 1496086440
  expected_departure_time: 1496090100
  expected_departure_time: 1496086457
  trip_stop_status: PLANNED
  trip_stop_status: DRIVING
  destination {
    destination_name: "Marconiplein"
    destination_detail: ""
  }
  destination {
    destination_name: "Marconiplein"
    destination_detail: ""
  }
  line_public_number: "23"
  line_public_number: "23"
  generated_timestamp: 1496086082
  generated_timestamp: 1496086082
}
general_messages {}

```

Voorbeeld bericht 'TravellInfo' gevuld met actuele reisinformatie voor twee ritten

Vrije teksten

Naast passeertijden kunnen ook vrije teksten gecommuniceerd worden naar een haltesysteem. De berichten waarmee vrije teksten worden verstuurd hebben ook een hash en zijn uniek te identificeren met het veld 'message_hash'. De vrije tekst kan worden verwijderd met het 'GeneralMessageRemove' bericht. Het haltesysteem is verantwoordelijk voor het tonen van de vrije tekst tussen de message_start_time en de message_end_time, of tot de tekst wordt gewist. Elke vrije tekst wordt los verstuurd naar het haltesysteem, indien er meerdere vrije teksten geldig zijn, bepaalt het haltesysteem hoe dit getoond wordt.

Overzichtdisplay

Indien een haltesysteem een overzicht moet tonen van meerdere quays zijn hier twee mogelijkheden voor.

- a) Het haltesysteem meldt zich aan op meerdere quays: alle QuayCodes (NL:Q:xxxxxxxx) die onderdeel zijn van de StopPlace waarvoor het overzichtdisplay geldt, worden opgenomen in het aanmeldbericht. Reisinformatie en vrije teksten voor alle gevraagde quaycodes worden naar het haltesysteem verstuurd. Voordeel van deze aanpak is dat bijvoorbeeld een uitstaphalte die onderdeel is van een StopPlace kan worden uitgesloten.
- b) Het haltesysteem meldt zich aan op een StopPlace: één (of eventueel meerdere) StopPlace(s) worden opgenomen in het aanmeldbericht (NL:S:xxxxxxxx). Op basis van CHB wordt voor de geldige Quaycodes reisinformatie verstuurd. Dit heeft als voordeel dat indien de samenstelling van de StopPlace wijzigt en het Centrale Halte Bestand wordt aangepast, het distributiesysteem dit opnieuw inleest en het overzichtdisplay automatisch meegaat.

Client Id

Als bijlage van dit document is het bestand DrisKoppelVlak(versie).proto toegevoegd. Dit bestand is leidend voor het koppelvlak.

In dit hoofdstuk worden de verschillende 'messages' verder uitgewerkt.

ClientId:

	Veld	Aantal	Uitleg
ClientId	subscriber_owner_code	(1)	De SubscriberOwnerCode wordt gevuld met een waarde uit een overkoepelende lijst. In deze lijst worden alle partijen beschreven die de distributie-, dashboard- of haltesystemen leveren. Deze bijlage wordt door DOVA OV_Data beheerd.
	subscriber_type	(1)	Geeft het type van de subscriber aan 0=distributiesysteem, 1=dashboardsysteem en 2=haltesysteem.
	serial_number	(1)	Volgnummer dat de combinatie van de velden uniek maakt. Voor een haltesysteem wordt geadviseerd om het serienummer van de hardware te gebruiken.

Berichten Koppelvlak Distributiesysteem - Haltesysteem

Subscribe

	Veld	Aantal	Uitleg	
Subscribe	client_id	(1)	Id waarmee client zich aanmeld.	
	stop_code	(1..n)	Een of meer quaycodes of één stopplacecode uit het Centrale Halte Bestand (CHB) waarop het haltesysteem zich wil abonneren en reisinformatie en vrije teksten van wil ontvangen. Beginnend met "NL:Q" of "NL:S". (verplicht). Er kan slechts geabonneerd worden op 1 stopplace (NL:S) en als meerdere quaycodes worden gestuurd moeten deze behoren tot dezelfde stopplace.	
	display_properties	(0..1)	Geeft de kenmerken van het ontvangende haltesysteem aan	
	DisplayProperties	text_characters	(0..1)	Het aantal tekens dat het display van het haltesysteem beschikbaar heeft voor de bestemmingsnaam. Het distributiesysteem kan daarmee bepalen welke bestemmingstekst (lengte) en detailbestemmingstekst gestuurd moeten worden. Indien deze waarde niet wordt gespecificeerd, wordt de langste waarde meegestuurd. Dit is van toepassing op haltesystemen die geen (praktische) lengtebeperking hebben.
	destination_determination	(1)	Geeft aan welke bestemmingsteksten naar het haltesysteem moeten worden verstuurd: <ul style="list-style-type: none"> - MAX_CHARACTERS het haltesysteem wil een bestemmingstekst en detail bestemmingstekst ontvangen met maximaal de lengte als gegeven in tekst_characters. Het distributiesysteem stuurt de eerstpassende bestemmingstekst en detailbestemmingstekst. (als tekst_characters = 18 wordt een bestemmingstekst 16 gestuurd.) - SELF_DETERMING het haltesysteem wil alle bestemmingsteksten en detailbestemmingsteksten ontvangen die het distributiesysteem heeft en selecteert zelf welke tekst wordt getoond, rekening houdend met de mogelijkheden van de schermen. 	
	filter_parameters	(0..1)	Geeft de parameters (T1, T2, P1, P2) waarmee	

			de KV8 reisinformatie gefilterd moet worden door het distributiesysteem.
FilterParameters	filter_on	(0..1)	Geeft aan of er wel of niet gefilterd moet worden.
	waitingtime_low	(0..1)	De ondergrens voor de wachttijd (T1).
	waitingtime_high	(0..1)	De bovengrens voor de wachttijd (T1).
	percentage_low	(0..1)	Het percentage (P1) waarmee de wachttijd af moet wijken onder de ondergrens van de wachttijd (T1).
	percentage_high	(0..1)	Het percentage (P2) waarmee de wachttijd af moet wijken boven de ondergrens (T1).
	filter_filter	(0..1)	Geeft aan welke gegevens van een rit gestuurd moeten worden naar het haltesysteem. Elk veld van dit object correspondeert met een gelijknamige kolom in het bericht 'PassingTimes' waarvan wordt aangegeven hoe dit moet worden verstuurd. Geldige waarden zijn opgenomen in de enumeratie 'Delivery'. Mogelijk zijn: <ul style="list-style-type: none"> - NEVER: standaardwaarde. Stuur dit veld nooit. (Protobuff 3 kent geen optionele velden. Als een zender (repeated) velden leeg laat, wordt er geen data voor deze velden in het bericht geplaatst. Echter bij het deserialiseren aan de ontvangende kant worden de velden gevuld met een defaultwaarde, zoals gespecificeerd in de protobuff documentatie.) - ALWAYS: in elk bericht wordt de meest recente waarde gestuurd. Elke onderstaand veld spreekt voor zich: de waarde van het veld is in het PassingTimes bericht gedocumenteerd.
FieldFilter	target_arrival_time	(0..1)	De aankomsttijd volgens dienstregeling. Deze kan meegestuurd worden, bijvoorbeeld om vertraging relatief aan die tijd te tonen
	target_departure_time	(0..1)	De vertrektijd volgens dienstregeling. Deze kan meegestuurd worden, bijvoorbeeld om vertraging relatief aan die tijd te tonen
	expected_arrival_time	(0..1)	De verwachte aankomsttijd. Kan gebruikt worden voor een haltesysteem dat aankomsttijden toont.
	expected_departure_time	(0..1)	De verwachte vertrektijd. Kan gebruikt worden voor een haltesysteem dat vertrektijden toont.

number_of_coaches	(0..1)	Dit geeft aan hoeveel (gekoppelde) rytuigen/voertuigen op een rit worden ingezet. Dit kan getoond bij metro/tram haltes, maar kan ook bij andere modaliteiten.
trip_stop_status	(0..1)	Dit veld geeft aan wat de status van de rit is. Displays die alleen vertrektijden volgens dienstregeling weergeven, zouden ervoor kunnen kiezen dit veld niet te ontvangen.
transport_type	(0..1)	Onder andere nodig om het juiste icoon te tonen als het voertuig op de halte aankomt of de wachttijd 0 minuten is. E.e.a. conform de Mijksenaar richtlijn.
wheelchair_accessible	(0..1)	Hiermee kan op een display getoond worden of het voertuig van een rit toegankelijk is voor rolstoelen.
is_timingstop	(0..1)	Hiermee bepaalt het haltesysteem welke informatie getoond wordt na aankomst van het voertuig op de halte.
stop_code	(0..1)	Relevant voor haltesystemen geabonneerd op meerdere quays, waarbij het van belang is te weten op welke quay een vertrek betrekking heeft.
destinations	(0..1)	
show_cancelled_trip	(0..1)	
block_code	(0..1)	
occupancy	(0..1)	
line_public_number	(0..1)	
side_code	(0..1)	
line_direction	(0..1)	Relevant om te kunnen filteren op het aantal te tonen ritten per lijn en richting.
line_color	(0..1)	Relevant voor haltesystemen die een lijnachtergrondkleur kunnen tonen
line_text_color	(0..1)	Relevant voor haltesystemen die een lijnvoorggrondkleur kunnen tonen
line_icon	(0..1)	Relevant voor haltesystemen die een icoon of afbeelding kunnen tonen
destination_color	(0..1)	Relevant voor haltesystemen die een bestemmingsachtergrondkleur kunnen tonen
destination_text_color	(0..1)	Relevant voor haltesystemen die een bestemmingsvoorggrondkleur kunnen tonen

	destination_icon	(0..1)	Relevant voor haltesystemen die een bestemmingsicoon of afbeelding kunnen tonen
	generated_timestamp	(0..1)	Relevant om eventueel "out of order" berichten te kunnen detecteren en te kunnen verifiëren dat altijd de laatste informatie wordt gebruikt
	journey_number	(0..1)	Ritnummer voor eventueel traceren van een rit op verschillende haltes
	description	(0..1)	Een omschrijving van de locatie van het haltesysteem of een begeleidende tekst. Wordt door het distributiesysteem meegestuurd in het autorisatieproces en kan door het dashboard worden getoond

SubscriptionResponse

	Veld	Aantal	Uitleg
SubscriptionResponse	success	(1)	Boolean-waarde die aangeeft of het haltesysteem succesvol aangemeld is bij het distributiesysteem (samenvatting van onderstaande status)
	status	(1)	Een van de volgende statuses ¹ (zie ook hoofdstuk 'Flows'): <i>REQUEST_INVALID</i> - Het aanmeldbericht voldoet niet aan de specificatie: een of meerdere velden is ongeldig. <i>STOP_INVALID</i> - Het aanmeldbericht heeft de juiste opbouw, maar het haltesysteem meldt zich aan op een bij het distributiesysteem onbekende stopplace- of quaycode (op basis van bron CHB). <i>AUTHORISATION_REQUIRED</i> - Het aanmeldbericht is juist, maar de betreffende combinatie van e-mailadres en systeemidentificatie zijn nog niet geautoriseerd. <i>PLANNING_SENT</i> - Indien het haltesysteem succesvol is aangemeld en er een planning is, wordt deze status verstuurd na het succesvol versturen van de planning. <i>NO_PLANNING</i> - Indien de aanmelding succesvol is, maar er geen planning is voor de betreffende quay of stopplacecode wordt deze status verstuurd ² . <i>AUTHORISATION_VALIDATED</i> - Het systeem is door een gebruiker zojuist geautoriseerd en de aanmelding is hiermee voldaan.
	timestamp	(1)	Tijdstip waarop het antwoord op het aanmeldbericht is samengesteld.

¹ Er is in de enumeratie 'Status' ruimte gelaten voor eventuele toekomstige uitbreidingen. De responses zijn gegroepeerd op categorie.

² Dit betekent letterlijk dat er geen passeertijden zijn voor de aangemelde halte(s). Er kunnen wel vrijeteksten zijn.

PublicName

Na de autorisatie via email, haalt het distributiesysteem de haltegegevens uit het CHB. Indien alle quaycodes of de stopplacecode die het haltesysteem heeft gestuurd bekend zijn in het CHB, wordt het bericht PublicName gestuurd met de gegevens van de quays plus de bovenliggende stopplace en place, indien gedefinieerd.

	Veld	Aantal	Uitleg
PublicName	public_name_place	(0..1)	Publieke naam van de place waar de quaycodes toe behoren.
	public_name_stop_place	(0..1)	Publieke naam van de stopplace waar de quaycodes toe behoren.
	stop_place_code	(0..1)	De code van de stopplace (NL:S) waar de quaycodes toe behoren.
	quay_names		Een lijst met de publieke naam van de haltes waarover het haltesysteem informeert.
	QuayName	quay_code	(1..n)
public_name_quay		(1..n)	Publieke naam van de Quay zoals bekend bij het CHB.

Een haltesysteem kan zich aanmelden met één of meer quaycodes of met een stopplacecode.

Als een haltesysteem zich bijvoorbeeld aanmeldt voor data van stopplace 'NL:S:50000101', dan wordt de volgende data verstuurd in public_names.

```
public_name_place = [Utrecht centraal];
public_name_stop_place = [CS Jaarbeurszijde];
stop_place_code = [NL:S:50000101];
stop_code=[NL:Q:50000120,NL:Q:50000121,NL:Q:50000122,NL:Q:50000123,NL:Q:50000124,NL:Q:50000125,NL:Q:50000126,NL:Q:50000127,NL:Q:50000128,NL:Q:50000129];
public_name_quay = [Halte C1, Halte C2, Halte C3, Halte C4, Halte C5, Halte C6, Halte C7, Halte C8, Halte C9, Halte C10];
```

Als een haltesysteem zich aanmeldt voor data van quay 'NL:Q:50000120' dan wordt de volgende data verstuurd in public_names.

```
public_name_place = [Utrecht centraal];
public_name_stop_place = [CS Jaarbeurszijde];
stop_place_code = [NL:S:50000101];
public_name_quay = [Halte C1];
stop_code=[NL:Q:50000120];
```

Het distributiesysteem controleert regelmatig het CHB op wijzigingen. Als de gegevens quays of de stopplace wijzigen, stuurt het distributiesysteem een Unsubscribe, zodat het haltesysteem zich opnieuw aanmeldt en de nieuwe gegevens ontvangt.

Unsubscribe

	Veld	Aantal	Uitleg
Unsubscribe	client_id	(1)	Client Id waarmee systeem is aangemeld
	is_permanent	(1)	<i>Indien nee:</i> dit betreft meestal een last-will bericht als het haltesysteem tijdelijk zijn verbinding met de MQTT broker is verloren. Maar het kan ook verstuurd worden door het distributiesysteem als het gepland uit de lucht gaat. Het wordt ook verstuurd door het distributiesysteem als er voor de halte wijzigingen in het CHB worden geconstateerd. <i>Indien ja:</i> het systeem wordt buiten dienst genomen of wordt niet meer op de huidige manier ingezet. Autorisatiegegevens worden door het distributiesysteem gewist.
	timestamp	(0..1)	Tijdstip waarop het bericht is samengesteld.

Na de ontvangst van een Unsubscribe van een haltesysteem (of van zichzelf) wacht het distributiesysteem op het haltesysteem om zich opnieuw aan te melden met een Subscribe. In het distributiesysteem wordt het haltesysteem inactive (=offline) tot het zich opnieuw aanmeldt¹.

Als het haltesysteem zich afmeldt met de permanent vlag geset, worden alle gegevens van het haltesysteem uit het distributiesysteem verwijderd en bestaat het voor het distributiesysteem niet meer.

¹ Let op dat het distributiesysteem bestaat uit twee clusters. Een haltesysteem kan de verbinding met cluster A verliezen en een nieuwe verbinding opzetten met cluster B en zich daar aanmelden. Voor cluster A is het haltesysteem dan inactive en voor cluster B active.

TravellInfo

	Veld	Aantal	Uitleg
TravellInfo	passing_times	(0..1)	Bevat een lijst met passeertijden van vertrekkende ritten (PassingTimes)
	pass_time_hash	(1..n)	(Zie ook "Standaard berichtafhandeling") Voor elke vertrekkende rit wordt een unieke hash berekend dat het haltesysteem kan gebruiken om de ritpassage te identificeren. Dit veld wordt altijd gevuld.
	target_arrival_time	(0..n)	Aankomsttijd volgens dienstregeling van de rit op de halte. Dit kan gebruikt worden om de vertraging ten opzichte van de dienstregeling te berekenen en tonen (bijvoorbeeld "11:10 +3"). De waarde betreft een unix timestamp, oftewel het aantal seconden sinds 1 januari 1970, in UTC. Afnemend haltesysteem is verantwoordelijk om deze informatie te vertalen naar de huidige tijd in tijdzone "Europe/Amsterdam" (tenzij anders gespecificeerd). Dit veld wordt niet verstuurd voor een beginhalte.
	target_departure_time	(0..n)	Vertrektijd volgens dienstregeling van de rit op de halte. Dit betreft een timestamp (zie hierboven). Dit veld wordt niet verstuurd voor een eindhalte.
	expected_arrival_time	(1..n)	De actuele prognose van de aankomsttijd op de betreffende halte van deze rit. Dit betreft een timestamp (zie hierboven), dit veld is altijd leeg voor beginhaltes. Het veld is altijd gevuld met de best bekende aankomsttijd.
	expected_departure_time	(1..n)	De actuele prognose van de vertrektijd op de betreffende halte van deze rit. Dit betreft een timestamp (zie hierboven), dit veld kan niet worden gefilterd, en is altijd leeg voor eindhaltes. Het veld is altijd gevuld met de best bekende vertrektijd.
	number_of_coaches	(1..n)	Het aantal rijtuigen of gekoppelde voertuigen dat deze rit uitvoert.
	trip_stop_status	(1..n)	Ritstatus voor het vertrek op de betreffende halte van de rit. Waarde is op basis van de enumeratie TripStopStatus ¹ , met als mogelijke waarden: <ul style="list-style-type: none"> - PLANNED - de rit is gepland - CANCELLED - de rit is opgeheven door een probleem van tijdelijke aard en wordt niet uitgevoerd. - DRIVING - de rit is begonnen en er is een actuele status bekend, maar de betreffende halte is nog niet bereikt - ARRIVED - de rit is aangekomen op de halte². - PASSED - de rit is de halte gepasseerd

¹ Deze enumeratie lijkt op de TripStopStatus enumeratie van Koppelvlak 7/8, maar met 'CANCELLED' ipv CANCEL.

² Deze status wordt herhaald gestuurd zolang het voertuig op de halte staat.

			- UNKNOWN - van het voertuig op de rit is geen actuele status bekend
	transport_type	(1..n)	Modaliteit voor betreffende vertrek, bijvoorbeeld om een symbool te tonen. Waarde is op basis van de BISON enumeratie TransportType, met als mogelijke waarden: <ul style="list-style-type: none"> - BUS - TRAM - METRO - TRAIN - BOAT
	wheelchair_accessible	(1..n)	Geeft aan of het voertuig op de rit rolstoeltoegankelijk is (bijvoorbeeld om een symbool te tonen)
	is_timingstop	(1..n)	Geeft aan of het betreffende vertrek een tijdhaltte voor de betreffende rit is.
	stop_code	(1..n)	Quay van het betreffende vertrek. Van belang voor haltesystemen die geabonneerd zijn op meerdere quays en onderscheid willen maken.
	destinations	(1..n)	Lijst met bestemming- en detailbestemminggegevens van de rit.
Destination	destination_name	(1..n)	Bestemming van de rit. Dit wordt door het distributiesysteem gevuld met de meest toepasselijke bestemming (qua lengte) voor het haltesysteem aan de hand van de aangeleverde maximum lengte in het Subscribe bericht. Of het wordt gevuld met een lijst met alle lengtes van de bestemming die bekend zijn in het distributiesysteem indien het haltesysteem heeft aangegeven zelf de best passende bestemming te kiezen. (SELF_DETERMINING).
	destination_detail	(0..n)	Idem voor de detail bestemmingen.
	show_cancelled_trip	(0..1)*	Geeft aan of een vervallen rit wel of niet getoond moet worden. Mogelijke waarden: <ul style="list-style-type: none"> - TRUE: de vervallen rit wordt wel getoond. - FALSE: de vervallen rit wordt niet getoond. In beide gevallen kan een verklarende vrije tekst getoond worden.
	block_code	(0..1)*	Geeft de omloop waarbinnen de rit wordt uitgevoerd, zoals opgenomen in de planning van de betreffende rit. Wordt o.a. gebruikt door een haltesysteem met Vecom detectie.
	occupancy	(0..1)*	Actuele bezettingsgraad van het voertuig (0 – 100%).
	line_public_number	(1..n)	Lijnnummer of andere identificatie van de lijn zoals bij de klant bekend (bijvoorbeeld 9K, 251, 12 of N1)
	side_code	(1..n)	Perronaanduiding (bijvoorbeeld A, G, etc)

	line_direction	(1..n)	Numerieke waarde die de richting van de betreffende rit aangeeft. Dit kan door het haltesysteem worden gebruikt om een bepaalde lijn/richting combinatie maar een beperkt aantal keer te tonen.
	line_color	(1..n)	Lijnkleur indien bekend (als achtergrond indien gecombineerd met tekst)
	line_text_color	(1..n)	Lijnkleur indien bekend (als voorgrond indien gecombineerd met tekst)
	line_icon	(1..n)	Lijnicoon om te tonen indien bekend. Bevat een referentie naar een icoon dat aanwezig is in het haltesysteem.
	destination_color	(1..n)	Bestemmingskleur indien bekend (als achtergrond indien gecombineerd met tekst)
	destination_text_color	(1..n)	Bestemmingskleur indien bekend (als voorgrond indien gecombineerd met tekst)
	destination_icon	(1..n)	Bestemmingsicoon om te tonen indien bekend. Bevat, een referentie naar een icoon dat aanwezig is in het haltesysteem.
	generated_timestamp	(1..n)	Het tijdstip waarop de reisinformatie over de betreffende rit gegenereerd is. Kan gebruikt worden om te verifiëren dat opeenvolgende berichten in de juiste volgorde worden verwerkt.
	journey_number	(1..n)	Ritnummer zoals bekend bij de vervoerder.
	general_messages		Bevat een lijst met vrije teksten (GeneralMessage)
GeneralMessage	message_hash	(1..n)	(Zie ook "Standaard berichtafhandeling") Voor elke vrije tekst/bericht wordt een unieke hash berekend dat het haltesysteem kan gebruiken om het bericht te identificeren. Dit veld wordt altijd gevuld.
	message_content	(0..n)	Inhoud van de vrije tekst indien bekend.
	message_start_time	(0..n)	Begintijd vanaf wanneer de vrije tekst getoond moet worden (dit is een timestamp, zie uitleg in bericht 'TravellInfo')
	message_end_time	(0..n)	Eindtijd tot wanneer de vrije tekst getoond moet worden (dit is een timestamp, zie uitleg in bericht 'TravellInfo'). Indien geen eindtijd wordt gespecificeerd moet het bericht getoond worden tot het wordt verwijderd.
	show_overview_display	(0..1)*	Geeft aan of de vrijetekst ook getoond moet worden op een overzichtdisplay. Mogelijke waardes: TRUE: de tekst moet ook op een overzichtdisplay worden getoond. FALSE: de tekst moet niet op een overzichtdisplay getoond worden. ONLY: de tekst moet alleen op een overzichtdisplay

			getoond worden en niet op een haltedisplay.
	message_title	(0..1)*	Titel van de vrije tekst. Afhankelijk van de mogelijkheden van het display, wordt de titel apart (en groter) getoond of wordt de titel opgenomen in de tekst.
	message_priority	(0..1)*	Geeft een indeling naar de inhoud van de tekst. Het geeft daarmee het belang van de tekst aan. Mogelijke waarden: CALAMITY: tekst is gerelateerd aan een acute en ongeplande verstoring waar veel reizigers hinder van ondervinden. PTPROCESS: mededelingen als gevolg van veelvoorkomende afwijkende situaties, zoals omleidingen, vertragingen enz. COMMERCIAL: Aankondigingen van wijzigingen in de toekomst, zoals nieuwe dienstregelingen, enz. MISC: Overige mededelingen.
	general_messages_removes		Bevat een lijst met vrije teksten die verwijderd moeten worden (GeneralMessageRemove)
GeneralMessageRemove	message_hash	(1..n)	(Zie ook "Standaard berichtafhandeling") Voor elke vrije tekst/bericht wordt een unieke hash berekend dat het haltesysteem kan gebruiken om het bericht te identificeren. Dit veld wordt altijd gevuld.
	passing_time_removes		Bevat een lijst met passeertijden die verwijderd moeten worden. (PassingTimeRemove)
PassingTimeRemove	pass_time_hash	(1..n)	De hash ID's van de passeertijden die verwijderd moeten worden.

* Deze velden zijn opgenomen voor toekomstig gebruik.

Indien bij de subscription bij destination_determination is aangegeven dat er gebruik gemaakt wordt van SELF_DETERMINING dan wordt de destinations als volgt gevuld per ritregel.

destination_name = [DestinationName50, DestinationName30, DestinationName24, DestinationName19,
DestinationName16]

destination_detail = [“”, “”, DestinationDetail24, DestinationDetail19, DestinationDetail16]

De eerste 2 waarden van destination_detail zullen dus leeg zijn omdat deze gegevens niet via de huidige
KV7/8 aangeleverd worden.

Berichten Distributiesysteem of Haltesysteem en Dashboardsysteem

SystemStatus

Dit bericht wordt generiek gebruikt door zowel het distributiesysteem als de haltesystemen om te communiceren naar een dashboardsysteem over de status van het systeem en de onderdelen daarvan. De status kan daarbij uitgedrukt worden in een van twee types: meetwaarden (Metric) of logberichten (LogMessage). Indien de informatie betrekking heeft op een haltesysteem, wordt dit in het client id duidelijk.

	Veld	Aantal	Uitleg	
SystemStatus	metrics	(0..n)	Actuele meetwaarden die gecommuniceerd moeten worden. Meetwaarden worden met regelmatige intervallen gepubliceerd door het metende systeem.	
	Metric	Client_id	(1)	Client Id waarmee systeem is aangemeld
		component	(1)	Naam van het betreffende component zoals gedefinieerd door de opdrachtgever (bijvoorbeeld "Behuizing" of "Display" voor haltesystemen of "Planning" voor een distributiesysteem). Toegestane tekens zijn letters en eventueel een lage streep (A-Z, a-z en _)
		component_index	(0..1)	Identificatie van het betreffende component in het systeem voor het geval er meerdere dezelfde componenten zijn. Optioneel, indien er geen tweede component is kan dit leeg worden gelaten.
		property	(1)	Eigenschap van het betreffende component dat wordt gemeten (bijvoorbeeld "deur", "temperatuur", "signaalsterkte" voor een haltesysteem of "ritten" of "berichten" voor een distributiesysteem). Toegestane tekens zijn letters en eventueel een lage streep (A-Z, a-z en _). Voor een betreffende combinatie 'component' en 'property' moet de eenheid waarin de waarde gemeten wordt, altijd hetzelfde zijn.
		value_type	(1)	Geeft aan van welk technisch type de waarde is (string, integer, double, bool, byte). Hiermee kan het ontvangende dashboardsysteem het juiste veld uitlezen.
		string_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde in tekst wordt gegeven.
		int_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde numeriek is.
		double_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde double is.
		bool_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde boolean is.
bytes_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde in bytes is.		

LogMessage	unit	(1)	De eenheid van de waarde. Het mag geen newlines/enters bevatten, verder zijn er geen beperkingen (geldige UTF-8 content).
	timestamp_begin	(1)	Het begintijdstip van de periode waarover de waarde is gemeten. (dit is een timestamp, zie uitleg in bericht 'TravellInfo')
	timestamp_end	(1)	Het eindtijdstip van de periode waarover de waarde is gemeten. Indien de meetwaarde een momentane waarde is, blijft dit veld leeg (dit is een timestamp, zie uitleg in bericht 'TravellInfo')
	logs	(0..n)	Eventuele logberichten die gecommuniceerd moeten worden. (LogMessage)
	client_id	(1)	Client Id waarmee systeem is aangemeld
	type	(1)	Een van de volgende waarden van de enumeratie 'StatusType': <ul style="list-style-type: none"> - <i>ERROR</i>: de boodschap betreft de statusovergang naar een foutsituatie - <i>WARNING</i>: de boodschap betreft de statusovergang naar een waarschuwing - <i>OK</i>: de boodschap betreft statusovergang van een <i>ERROR</i> of <i>WARNING</i> terug naar een geaccepteerde waarde. In dit geval wordt de duur van de storing of waarschuwing in seconden meegegeven. - <i>LOG</i>: de boodschap betreft een andere, meer algemene, melding die geen statusovergang betreft.
	code	(1)	Een numerieke code die het type gebeurtenis aangeeft waarover een logbericht wordt gecommuniceerd.
	message	(1)	De tekstuele boodschap behorende bij de geregistreerde gebeurtenis, deze wordt in een dashboard aan de gebruiker gepresenteerd. Het bericht mag geen newlines/enters bevatten, verder zijn er geen beperkingen (geldige UTF-8 content).
	duration	(0..1)	De duur van een storing (<i>ERROR</i>) of waarschuwing (<i>WARNING</i>) in seconden. Deze waarde wordt alleen gevuld bij een boodschap van het type <i>OK</i> .
	timestamp	(0..1)	Het tijdstip waarop het bericht gegenereerd is (dit is een timestamp, zie uitleg in bericht 'TravellInfo')

CDDInfoRequest

Met dit bericht kan een dashboardstelsel bij het distributiesysteem een lijst opvragen van alle haltesystemen die zich hebben aangemeld bij het distributiesysteem.

De ownercode wordt meegestuurd omdat een dashboard meestal alleen de "eigen" systemen opvraagt.

	Veld	Aantal	Uitleg
CDDInfoRequest	client_id	(1)	Client Id waarmee het (dashboard)stelsel is aangemeld.
	subscriber_owner_code	(0..1)	Als ingevuld stuurt het distributiesysteem alleen de id's van de haltesystemen van de leverancier. Als de leverancier niet is gespecificeerd of nul is, stuurt het distributiesysteem de ID's van alle aangemelde haltesystemen.

Het distributiesysteem bestaat uit twee clusters en een haltesysteem bepaalt zelf op welk van de twee clusters het zich aanmeldt. Op enig moment is dus een deel van alle haltesystemen in Nederland aangemeld op de ene cluster en andere haltesystemen zijn aangemeld op de andere cluster. Voor een volledig overzicht moet een dashboard een CDDInfoRequest sturen naar beide clusters en de informatie die het terugkrijgt combineren.

StopSystems

Op verzoek van een dashboardsysteem stuurt het distributiesysteem, met dit bericht, een lijst van alle haltesystemen die zich (ooit) bij het distributiesysteem hebben aangemeld en nooit permanent hebben afgemeld. In de lijst wordt aangegeven of het haltesysteem actief of inactief is. Een haltesysteem is actief voor het distributiesysteem als het zich heeft aangemeld, verbonden is en reisinformatie ontvangt.

	Veld	Aantal	Uitleg	
StopSystems	client_id	(1)	Client Id waarmee het distributiesysteem is aangemeld.	
	subscribed_systems	(0..n)		
	SubscribedSystem	subscriber_owner_code	(1)	De "owner" van het haltesysteem (meestal de leverancier) Zie het hoofdstuk Client Id
		serial_number	(1)	Uniek volgnummer van het haltesysteem binnen het domein van de "owner". Voor een haltesysteem meestal gelijk aan het serienummer van de hardware.
		active	(1)	Geeft aan of het haltesysteem is aangemeld op het distributiesysteem of niet.

Het distributiesysteem bestaat uit twee clusters en een haltesysteem is altijd maar actief op één van de clusters. Als een haltesysteem inactief is op beide clusters betekent dat, dat het haltesysteem wel bestaat en bekend is maar dat het op dat moment niet is aangemeld op één van de clusters en geen reisinformatie ontvangt.

Als een haltesysteem zich permanent heeft afgemeld (`is_permanent = true`) bestaat het systeem niet meer in het distributiesysteem en komt het niet voor in de lijst.

In de lijst staan haltesystemen onafhankelijk van de versie van het koppelvlak waarmee ze communiceren.

Berichten tussen Halte- en Dashboardsysteem

Begrippen en regels

Bij de berichtenstroom tussen halte- en dashboardsystemen gelden de volgende begrippen en regels:

- Een haltesysteem (in de praktijk ook vaak display genoemd) bestaat uit componenten. Voorbeeld: behuizing, scherm, software, ...
- Een component heeft eigenschappen. Voorbeeld: een behuizing heeft een deur, een scherm heeft een aantal defecte LED's, ...
- Eigenschappen hebben een unieke code¹.
- Eigenschappen hebben een waarde. Voorbeeld: een deur is open, het aantal defecte LED's is 4.
- Sommige eigenschappen kennen conditie-overgangen. Deze treden op als de waarde een grens overschrijdt. Voorbeeld: een deur gaat van dicht naar open, het aantal defecte LED's gaat van 4 naar 5, waarbij 5 als grens is gedefinieerd.
- De waardes van alle eigenschappen tezamen vormen de status van het haltesysteem.

Bij de aanschaf van een haltesysteem wordt overeengekomen uit welke componenten een haltesysteem bestaat en welke eigenschappen die componenten hebben. Ook wordt van elke conditie-overgang het statustype bepaald (ERROR, WARNING of OK)

Het haltesysteem is verantwoordelijk voor het publiceren van de overeengekomen conditie-overgangen naar de dashboards zodat gebruikers de status van de haltesystemen kunnen volgen, zonder dat er regelmatig statusinformatie moet worden opgevraagd. Indien een eigenschap van een component van een haltesysteem een conditie-overgang ondergaat, wordt een 'SystemStatus' bericht op het topic van het haltesysteem gepubliceerd. Indien een dashboardsysteem meer informatie nodig heeft kan de gehele status van een haltesysteem worden opgevraagd. Het is binnen het koppelvlak niet mogelijk om de grenswaarden voor conditie-overgangen in te stellen. Dit wordt verondersteld onderdeel uit te maken van een configuratiebestand op het haltesysteem.

- Het haltesysteem **kan** een logboodschap versturen als de waarde van een eigenschap verandert.
- Het haltesysteem verstuurt **altijd** een logboodschap bij een overeengekomen conditie-overgang.
- Logboodschappen bij conditie-overgangen hebben een type ERROR, WARNING of OK.
- Logboodschappen van het type ERROR of WARNING, van een eigenschap (=code) worden **altijd** gevolgd door een logboodschap van het type OK van diezelfde eigenschap. Voorbeeld: deur open (WARNING) moet altijd gevolgd worden door een deur dicht (OK).
- Bij elke logboodschap van het type OK berekent het haltesysteem de duur van de storing of waarschuwing en stuurt dat mee met de logboodschap.

Informatieverzoeken

Vanuit de dashboardsystemen kunnen vier verschillende informatieverzoeken naar een haltesysteem worden verstuurd. Het haltesysteem reageert hierop met een antwoord. De vier verzoeken zijn voor het ophalen van:

- De gegevens over het systeem:
gedefinieerd in naam/waarde combinaties. Bijvoorbeeld: schermtype = LED, of aantal regels = 4.
- De actuele status:
lijst met alle eigenschappen waarvoor een conditie-overgang en een logboodschap is gedefinieerd, met de actuele waarde van de eigenschap plus het type van de laatste logboodschap.

¹ Deze codes worden beheerd door OV-data en zijn uniek over de DRIS implementaties heen.

- De gedetailleerde actuele status:
Een lijst met statusgegevens die de displayleverancier zelf kan definiëren en opvragen.
- De reisinformatie in het haltesysteem:
informatie over actuele en toekomstige ritten, alsmede alle actuele en toekomstige vrije teksten.
- De tekst die op het scherm staat:
informatie die getoond wordt op het scherm of de schermen (snapshot).

Bestandsuitwisseling

Het centrale distributiesysteem heeft een FTP server beschikbaar waarmee bestanden kunnen worden uitgewisseld tussen dashboard- en haltesystemen. Een dashboardsysteem kan een configuratiebestand klaarzetten op de FTP server en stuurt vervolgens een bericht naar één of meer haltesystemen dat het bestand kan worden opgehaald.

Andersom kan een haltesysteem (regelmatig) een tracebestand klaarzetten op de FTP server en een bericht sturen naar een dashboardsysteem. Deze mogelijkheid om tracebestanden te maken in het haltesysteem en deze naar een dashboardsysteem te versturen is alleen bedoeld voor incidentele analyse. Het is uitdrukkelijk niet de bedoeling dat meerdere haltesystemen gedurende langere tijd gebruik maken van deze faciliteit.

De FTP servers tussen de beide clusters worden niet gesynchroniseerd. Een haltesysteem kan op enig moment communiceren met één van beide clusters en het is de verantwoordelijkheid van de dashboardsystemen om de configuratiebestanden op beide FTP servers te plaatsen. Evenzo zet een haltesysteem een tracebestand neer op de cluster waarmee het op dat moment communiceert en het is de verantwoordelijkheid van de dashboardsystemen om te onderzoeken op welke cluster een tracebestand staat.

InfoRequest

	Veld	Aantal	Uitleg
InfoRequest	client_id	(1)	Client Id waarmee het (dashboard)systeem is aangemeld.
	request_type	(1)	Een van de volgende waarden van de enumeratie 'RequestType': <ul style="list-style-type: none"> - <i>SYSTEM_INFO</i>: het dashboard vraagt de algemene informatie van het haltesysteem op. - <i>STATUS</i>: het dashboard vraagt de actuele waarde van de eigenschappen op. - <i>TRAVEL_INFO</i>: het dashboard vraagt de informatie op van actuele en toekomstige ritten en vrije teksten. - <i>SCREEN_CONTENT</i>: <i>het commando vraagt wat getoond wordt op het scherm of op de schermen.</i> - <i>DETAIL_STATUS</i>: het dashboard vraagt hiermee een lijst op van leverancierspecifieke eigenschappen plus de actuele waarde.
	request_parameter	(0..1)	Parameter die de opgevraagde informatie nader definieert. De betekenis is afhankelijk van het request_type. <ul style="list-style-type: none"> - Bij het opvragen van het schermbeeld (<i>SCREEN_CONTENT</i>) geeft de parameter het aantal seconden dat de updates van het schermbeeld verstuurd moeten worden. Als de parameter niet is gespecificeerd of nul is, stuurt het haltesysteem het schermbeeld eenmalig. - Bij de andere request_types wordt de parameter in het haltesysteem genegeerd.

SystemInfo

Met dit bericht stuurt een haltesysteem de lijst met overeengekomen¹ informatie-elementen in de vorm van naam/waarde paren. Het bericht wordt verstuurd in antwoord op een verzoek van een dashboardstelsel.

SystemInfo	client_id	(1)	Client Id waarmee het haltesysteem is aangemeld	
	system_info_lines	(1..n)	Lijst met met Informatieregels over het haltesysteem. Elke regel bestaat uit een naam plus een waarde.	
	SystemInfoLine	Name	(1)	De, niet gestandaardiseerde, naam van het informatie-element.
		value_type	(1)	Geeft aan van welk technisch type de waarde is (string, integer, double, bool, byte). Hiermee kan het ontvangende dashboardstelsel het juiste veld uitlezen.
		string_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde in tekst wordt gegeven.
		int_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde numeriek is.
		double_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde double is.
		bool_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde boolean is.
		bytes_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde in bytes is.

¹ Bij de aanschaf van het haltesysteem wordt door de opdrachtgever een lijst met gegevens opgegeven.

StatusOverview

Met dit bericht stuurt een haltesysteem een lijst met actuele waardes van alle overeengekomen eigenschappen van het haltesysteem, plus het type van de laatste Logboodschap die is verstuurd (LogMessage). Tesaamen vormen deze waardes een korte actuele status van het haltesysteem, bedoeld voor eerstelijns onderhoud, met name door DOVA¹.

	Veld	Aantal	Uitleg
StatusOverview	client_id	(1)	Client Id waarmee (halte)systeem is aangemeld
	status_overview_lines	(1...n)	Lijst met met statusregels met de indeling van StatusOverviewLine.

DetailStatusOverview

Met dit bericht stuurt een haltesysteem een lijst met actuele waardes van andere eigenschappen van het haltesysteem. De eigenschappen kunnen door de displayleverancier worden bepaald en de lijst zal over het algemeen veel gedetailleerder informatie over het haltesysteem geven, bijvoorbeeld voor de tweedelijns monteur.

	Veld	Aantal	Uitleg
DetailStatusOverview	client_id	(1)	Client Id waarmee (halte)systeem is aangemeld
	Detail_status_overview_lines	(1...n)	Lijst met met statusregels met de indeling van StatusOverviewLine.

¹ Deze lijst is met opzet beperkt gehouden en bevat geen project- of leverancierspecifieke gegevens. Beseft dat DOVA het beheer doet van vele duizenden haltesystemen, van veel verschillende leveranciers. Door de lijst kort en gestandaardiseerd te houden heeft DOVA van alle systemen eenzelfde statusoverzicht.

StatusOverviewLine

Beide bovenstaande statusberichten hebben dezelfde structuur, zodat het vullen en lezen van de berichten met dezelfde code kan gebeuren. Wel zullen bepaalde velden in het ene statusbericht wel en in het andere niet gevuld worden.

	Veld	Aantal	Uitleg
StatusOverviewLine	component	(1)	De naam van een deel van een haltesysteem.
	component_index	(0..1)	Identificatie van het betreffende component in het systeem voor het geval er meerdere dezelfde componenten zijn. Optioneel, indien er geen tweede component is kan dit leeg worden gelaten. Dit veld zal in het standaard statusoverzicht meestal leeg gelaten worden, omdat het voor de eerstelijns niet van belang is.
	property	(1)	De naam van de componenteigenschap waarover gerapporteerd wordt.
	value_type	(1)	Geeft aan van welk technisch type de waarde is (string, integer, double, bool, byte). Hiermee kan het ontvangende dashboardsysteem het juiste veld uitlezen.
	string_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde in tekst wordt gegeven.
	int_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde integer is.
	double_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde double is.
	bool_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde boolean is.
	bytes_value	(0..1)	Dit veld wordt gevuld met de actuele waarde van de eigenschap als de waarde in bytes is.
	status	(0..1)	Het type van de laatst verstuurd logboodschap (ERROR, WARNING, OK, LOG). De enumeratie is beschreven bij de LogMessage. Dit veld zal meestal niet gevuld worden in het gedetailleerde statusoverzicht, omdat de statusgegevens die voorkomen in de logberichten in het gewone statusoverzicht moeten staan. ¹

¹ Het staat een displayleverancier vrij om het veld te gebruiken om bij bepaalde statusgegevens extra aandacht te vragen in het dashboard of statusovergangen te markeren (bijv. ERROR)

TravellInfoResponse

Met dit bericht stuurt het haltesysteem alle vertrektijden en vrije teksten die het heeft.

	Veld	Aantal	Uitleg
TravellInfoResponse	client_id	(1)	Client Id waarmee het (halte)systeem is aangemeld.
	Travel_info_content	(1)	Dit bericht 'TravellInfo' is eerder gedocumenteerd. Het haltesysteem moet alle op dat moment bekende ritten en vrije teksten in het bericht opnemen en publiceren, zodat gecontroleerd kan worden of de juiste informatie opgeslagen is.

ScreenContentResponse

Met dit bericht stuurt het haltesysteem het schermbeeld inclusief eventuele op dat moment niet zichtbare inhoud, bijvoorbeeld bij alterneren of als meerdere blokken vrije tekst worden getoond. Indien op het haltesysteem meerdere schermen zijn aangesloten die ieder het schermbeeld tonen, wordt het beeld 1 keer verstuurd. Indien op het haltesysteem meerdere schermen zijn aangesloten die ieder een deel van het schermbeeld tonen, moet het schermbeeld van ieder scherm separaat gestuurd worden, inclusief een volgnummer van het scherm.

Dit bericht moet eenmalig gestuurd worden als antwoord op een InfoRequest van het type SCREEN_CONTENT en vervolgens steeds als één van de schermbeelden verandert¹, voor de periode die in de request_parameter staat aangegeven.

	Veld	Aantal	Uitleg
ScreenContentResponse	client_id	(1)	Client Id waarmee het (halte)systeem is aangemeld.
	screen_contents	(0..n)	
	ScreenContent		
	screen_index	(0..1)	Indien een haltesysteem meerdere schermen heeft die verschillende informatie tonen, is dit het volgnummer van het betreffende scherm. Indien het haltesysteem maar uit een scherm bestaat of uit meerdere schermen die dezelfde informatie tonen, mag dit veld leegblijven.
	content_type	(1)	Een van de volgende formaten die aangeven hoe het veld 'content' is gevuld: <ul style="list-style-type: none"> - <i>TEXT</i>: De geformatteerde tekst zoals te zien is op het display (op het moment van opname, inclusief het alternerende deel van de tekst) - <i>IMAGE_PNG</i>: Een afbeelding in Portable Network Graphics (PNG) formaat (op het moment van opname, inclusief het alternerende deel van de tekst)

¹ Let op dat alterneren, of het tonen van een ander vrije tekst blok geen verandering is die een nieuw bericht tot gevolg heeft, omdat de scherm inhoud wordt verstuurd inclusief de inhoud die niet zichtbaar is door alterneren of meerdere blokken vrije tekst.

			<ul style="list-style-type: none"> - <i>IMAGE_GIF</i>: Een afbeelding in het Graphics Interchange Format (GIF) formaat. Hierbij is het mogelijk meerdere (bewegende) frames op te nemen om altemerende teksten weer te kunnen geven. De leverancier van het haltesysteem kiest aan de hand van de eisen van opdrachtgever en type scherm het meest geschikte formaat.
	content	(1)	Indien type = TEXT, de ruwe, geformateerde tekst. Indien type = IMAGE_PNG of IMAGE_GIF, base64 encoded data van de afbeelding.
	timestamp	(1)	Tijdstip waarop dit bericht is samengesteld door het haltesysteem (zie uitleg in bericht 'TravellInfo').

FileAvailable

Dashboardsystemen kunnen een configuratiebestand klaarzetten voor een haltesysteem. Op de centrale distributiecluster is daartoe een FTP server beschikbaar. Andersom kunnen haltesystemen tracebestanden op de FTP server klaarzetten voor dashboardsystemen. Het afnemende systeem kan geïnformeerd worden dat er een bestand beschikbaar is, met het hier beschreven bericht.

	Veld	Aantal	Uitleg
FileAvailable	client_id	(1)	Client Id waarmee het (dashboard of halte)systeem is aangemeld.
	file_name	(1)	Naam van het bestand dat klaar staat op de FTP server.

Berichten tussen Haltesystemen onderling

Voor de audiovoorziening op een station is het in uitzonderingsgevallen nodig dat de haltesystemen een bericht naar elkaar versturen¹. Met dit bericht kan een haltesysteem (meestal een overzichtdisplay) aan de andere haltesystemen binnen de stopplace doorgeven dat van een set ritten, bepaalde wijzigingen in de informatie moeten worden uitgesproken.

AudioTracking

	Veld	Aantal	Uitleg
AudioTracking	pass_time_hash	(1...n)	De ritnummers waarvan de wijzigingen moeten worden uitgesproken.
	duration	(1)	De tijd in minuten dat de wijzigingen moeten worden uitgesproken. (max. 15 minuten)

Omdat de Haltesystemen van een stopplace op enig moment verbonden kunnen zijn met verschillende clusters van het distributiesysteem, wordt dit bericht tussen de clusters uitgewisseld.

¹ De Haltesystemen communiceren onderling dus via de centrale MQTT broker.

MQTT

Algemeen

Om data te kunnen versturen en ontvangen, moeten alle systemen (Distributie-, Halte- en Dashboardsystemen) verbonden worden met de centrale MQTT broker, die werkt conform MQTT versie 5.

Verbinden

Bij het verbinden van de clientsystemen met de MQTT broker zijn de volgende instellingen van belang:

- Keep Alive
De Keep Alive periode moet instelbaar zijn in de clientsystemen met een defaultwaarde van 15 seconden voor Distributie- en Dashboardsystemen en 60 seconden voor Haltesystemen.
- Client id
Voor de verbinding met de broker is een naam nodig. De naam wordt samengesteld uit de drie velden van Client Id gescheiden door een ‘_’ (Underscore): de volgorde is als volgt.
 - SubscriberOwnerCode
 - Type
 - Serialnumber

Een voorbeeld van een client id is: DOVA_1_1 of SURTRONIC_2_42741

Een client id dient uniek te zijn over alle systemen heen. Indien het client id wijzigt zal de autorisatieprocedure opnieuw moeten worden doorlopen.

- Clean Start
True voor Distributie- en Haltesystemen en False voor Dashboardsystemen.
Omdat een distributiesysteem alleen informatie stuurt, is een sessie met geschiedenis niet relevant. Een Haltesysteem krijgt bij aanmelden (initieel en na verlies van de verbinding) alle informatie opnieuw opgestuurd. Een sessie met geschiedenis is daarom niet relevant.
Een Dashboardsysteem moet echter altijd een complete historie bijhouden van de aangesloten Halte- en Distributiesystemen en moet daarom alle informatie die gestuurd is, ontvangen. Een sessie met (lange) geschiedenis is daarom van belang. Expiry Interval voor het Dashboard wordt op : 604800 Sec gezet (1Week)
- Last Will
Bij verlies van de verbinding tussen een clientsysteem en de Broker moet altijd een Last Will worden uitgezonden, zodat het verbindingsverlies bekend wordt in de Dashboardsystemen.
Voor Distributiesystemen moet als Last Will op topic `unsubscribe/<version>/0/<SubscriberOwnerCode>/<Serialnumber>` een Unsubscribebericht geconfigureerd staan dat aangeeft dat de verbinding is verbroken.
Voor Haltesystemen moet als Last Will op topic `unsubscribe/<version>/2/<SubscriberOwnerCode>/<Serialnumber>` een Unsubscribe bericht¹ geconfigureerd staan met als enige waarde “is_permanent” = false (dit bericht kan dus eenmalig tijdens initialisatie worden aangemaakt).

¹ Zoals beschreven in het hoofdstuk Flows kan dit bericht ook door het haltesysteem worden verstuurd als de software of het haltesysteem moet herstarten om welke reden dan ook.

Voor Dashboardsystemen moet op topic `unsubscribe/<version>/1/<SubscriberOwnerCode>/<Serialnumber>` een Unsubscribebericht geconfigureerd staan dat aangeeft dat de verbinding is verbroken.

Version

Om verschillende versies van het koppelvlak te kunnen ondersteunen bevat elk topic een versienummer. Het versienummer bestaat uit één getal en kent geen subgetallen. Dit koppelvlakdocument wordt geïmplementeerd met versie 4 in de topic's. Er wordt maar één getal gebruikt omdat er verondersteld wordt dat het altijd om een significante wijziging gaat bijvoorbeeld en dat kleinere wijzigingen geclusterd worden. Het versienummer in de topic's is niet gelijk aan het versienummer van de documenten die het koppelvlak beschrijven, zodat daar wel versienummers met subgetallen gebruikt kunnen worden. Het heeft organisatorische voordelen om het versienummer van de documentatie los te koppelen van het versienummer in de software. Op die manier kun je in de documentatie meerdere versies gebruiken zonder dat dit geïmplementeerd hoeft te worden (tekstuele wijzigingen). Ook kun je wijzigingen opsparen en goed administreren en op enig moment beslissen om de collectie aan wijzigingen te implementeren in de software.

Als er een nieuwe versie van het koppelvlak wordt geïntroduceerd, wordt dit eerst in het distributiesysteem geïmplementeerd. Het distributiesysteem abonneert zich op de relevante topic's met het nieuwe versienummer naast de bestaande subscriptions op de topic's met het oude versienummer. Vanaf dat moment kunnen de haltesystemen stuk voor stuk omgebouwd worden. Een omgebouwd haltesysteem gaat dan op de "nieuwe" topic's communiceren. Zodra alle haltesystemen op de nieuwe topic's luisteren en publiceren, hoeft het distributiesysteem niet meer te luisteren en te publiceren op de topic's met het oude versienummer, en kunnen die topic's verwijderd worden.

Bij een migratie is extra aandacht nodig voor de dashboardsystemen. Het centrale DOVA-dashboard zal gelijk met het distributiesysteem de nieuwe topic's gaan gebruiken, maar dashboards van externe partijen hoeven pas te gaan luisteren op de topic's met het nieuwe versienummer zodra het eerste haltesysteem wordt omgebouwd dat wordt gevolgd in het dashboard.

Migratie naar een nieuwe koppelvlakversie is een complex project met risico's op verstoring in de informatievoorziening. Daarom wordt een nieuwe versie alleen geïntroduceerd als het echt niet anders kan en mogen er nooit meer dan twee versies tegelijk actief zijn in het hele systeem.

Topics

Per type bericht topic wordt exact één topic gedefinieerd zodat halte-, distributie en dashboardsystemen zich kunnen abonneren op bepaalde type berichten.

subscribe/<Version>/<Type>/<SubscriberOwnerCode>/<SerialNumber>

Publicatie: Individueel Haltesysteem

Abonnee: Distributiesysteem, Dashboardsystemen

Bericht: Subscribe

QoS: Exactly once (2)

Retain: false

Type, SubscriberOwnerCode, SerialNumber: van het Haltesysteem dat zich aanmeldt.

In het aanmeldproces wordt dit topic gebruikt door haltesystemen om aanmeldinformatie op te publiceren. Hierdoor weten het distributiesysteem en de dashboardsystemen welke haltesystemen op dat moment actief zijn en op welke topics gepubliceerd moet worden om te communiceren met het haltesysteem. Ook krijgt het distributiesysteem eventuele stuurparameters zoals bijvoorbeeld filters, meegestuurd.

subscription_response/<Version>/<Type>/<SubscriberOwnerCode>/<SerialNumber>

Publicatie: Distributiesysteem

Abonnee: Individueel Haltesysteem, Dashboardsystemen

Bericht: SubscriptionResponse

QoS: Exactly once (2)

Retain: false

Type, SubscriberOwnerCode, SerialNumber: van het Haltesysteem dat zich heeft aangemeld.

Resultaten en acties met betrekking tot de subscription worden op dit topic gepubliceerd door het distributiesysteem. Haltesystemen moeten op dit topic abonneren om de status te ontvangen van een aanmelding en eventuele vervolg acties uit te voeren. Er wordt voor elk aanmeldbericht op *subscribe/<Version>/<Type>/<SubscriberOwnerCode>/<SerialNumber>* een antwoord verstuurd op dit topic. De dashboardsystemen kunnen zich op dit topic abonneren, zodat ze het subscriptionproces kunnen volgen en vastleggen.

publicname/<Version>/<Type>/<SubscriberOwnerCode>/<SerialNumber>

Publicatie: Distributiesysteem

Abonnee: Individueel Haltesysteem, Dashboardsystemen

Bericht: PublicName

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, SerialNumber: van het Haltesysteem.

Dit bericht wordt verstuurd na authenticatie en elke keer dat de gegevens van Quay of stopplacecodes veranderen in het CHB. Haltesystemen moeten op dit topic abonneren om de informatie over de Quays en Stopplaces te ontvangen. De dashboardsystemen kunnen zich op dit topic abonneren, zodat ze de wijzigingen kunnen volgen en vastleggen.

unsubscribe/<Version>/<Type>/<SubscriberOwnerCode>/<SerialNumber>

Publicatie: Distributiesysteem, Individueel Haltesysteem

Abonnee: Distributiesysteem, Dashboardsystemen en Individueel Haltesysteem

Bericht: Unsubscribe

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van een Haltesysteem dat zich eerder heeft aangemeld, of van een distributiesysteem dat gepland down wil gaan.

Dit topic wordt gebruikt voor zowel het last-will bericht als permanente afmeldberichten.

travelinfo/<Version>/<Type>/<SubscriberOwnerCode>/<Serialnumber>

Publicatie: Distributiesysteem

Abonnee: Individueel Haltesysteem

Bericht: TravellInfo

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het Haltesysteem dat de informatie moet tonen. Reisinformatieberichten zoals hierboven omschreven worden op dit topic gepubliceerd. Dit kunnen zowel passeertijden als vrije teksten zijn. Passeertijden kunnen op korte termijn (realtime) of langere termijn (planning) verstuurd worden.

De QoS op dit topic is bewust op 1 gehouden omdat de reisinformatieberichten op twee manieren van elkaar te onderscheiden zijn (door middel van het veld 'pass_time_hash' en het optioneel te ontvangen veld 'generated_timestamp'). Hierdoor is er dus geen nadeel als hetzelfde bericht meerdere keren wordt ontvangen door een haltesysteem. Dit heeft performancevoordelen.

systemstatus/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: Individueel Haltesysteem, Distributiesysteem, individueel Dashboardsysteem

Abonnee: Dashboardsystemen (autorisatie wordt afgedwongen op basis van SubscriberOwnerCode)

Bericht: SystemStatus

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het systeem dat de status publiceert. Op dit topic publiceren alle soorten systemen logboodschappen en meetwaarden. Met de logboodschappen kan in een dashboardsysteem de status van een systeem gevolgd worden en de meetwaarden geven (statistische) informatie over de werking van het systeem.

CDDinforequest/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel dashboardsysteem

Abonnee: individueel distributiesysteem.

Bericht: CDDInfoRequest

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het distributiesysteem dat moet antwoorden. Op dit topic publiceren de dashboardsystemen hun verzoek om informatie van een distributiesysteem.

stopsystems/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel distributiesysteem

Abonnee: dashboardsysteem

Bericht: SubscribedSystems

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het vragende Dashboardsysteem.

Op dit topic publiceert een distributiesysteem de lijst met bekende haltesystemen in antwoord op het verzoek van een dashboardsysteem.

inforequest/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel Dashboardsysteem

Abonnee: individueel haltesysteem.

Bericht: InfoRequest

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het Haltesysteem dat moet antwoorden.

Op dit topic publiceren de dashboardsystemen hun verzoek om informatie van een haltesysteem.

systeminfo/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel haltesysteem

Abonnee: het vragende dashboardsysteem

Bericht: SystemInfo

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het vragende Dashboardsysteem.

Op dit topic publiceert een haltesysteem de systeem informatie in antwoord op het verzoek van een dashboardsysteem.

statusoverview/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel haltesysteem

Abonnee: het vragende dashboardsysteem

Bericht: StatusOverview

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het vragende Dashboardsysteem.

Op dit topic publiceert een haltesysteem het statusoverzicht in antwoord op het verzoek van een dashboardsysteem.

detailstatusoverview/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel haltesysteem

Abonnee: het vragende dashboardsysteem

Bericht: DetailStatusOverview

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het vragende Dashboardsysteem.

Op dit topic publiceert een haltesysteem het gedetailleerde statusoverzicht in antwoord op het verzoek van een dashboardsysteem.

travelinforesponse/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel haltesysteem

Abonnee: het vragende dashboardsysteem

Bericht: TravelInfoResponse

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het vragende Dashboardsysteem.

Op dit topic publiceert een haltesysteem de informatie over alle ritten en vrije teksten die het heeft, in antwoord op het verzoek van een dashboardsysteem.

screencontent/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel haltesysteem

Abonnee: het vragende dashboardsysteem

Bericht: InfoRequest

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het vragende Dashboardsysteem.

Op dit topic publiceert een haltesysteem de informatie die op het scherm (de schermen) wordt getoond, in antwoord op het verzoek van een dashboardsysteem.

fileavailable/<Version>/<Type>/<SubscriberOwnerCode>/<serialnumber>

Publicatie: individueel halte- of dashboardsysteem

Abonnee: individueel dashboardsysteem of haltesysteem

Bericht: FileAvailable

QoS: Atleast once (1)

Retain: false

Type, SubscriberOwnerCode, Serialnumber: van het systeem waarvoor het bestand bedoeld is.

Op dit topic geeft een dashboardsysteem aan dat er een configbestand klaar staat voor één of meer haltesystemen. Evenzo geeft een haltesysteem op dit topic aan dat er een tracebestand klaar staat voor één of meer dashboardsystemen.

audiotracking/<Version>/<Stopplace>

Publicatie: individueel haltesysteem waarop een reiziger audio-informatie heeft opgevraagd.

Abonnee: individueel haltesysteem binnen de stopplace van de zender.

Bericht: AudioTracking

QoS: Atleast once (1)

Retain: false

Op dit topic geeft een Haltesysteem aan dat een reiziger audio-informatie heeft opgevraagd waarvan de wijzigingen ook op andere Haltesystemen moeten worden uitgesproken.

Stopplace: de code van de stopplace waartoe het publicerende haltesysteem behoort.

Een haltesysteem moet zich voor de reisinformatie op vier topics abonneren:

- *travel_information/<Version>/2/<SubscriberOwnerCode>/<Serialnumber>*. Het haltesysteem abonneert zich op dit topic met het eigen Type, SubscriberOwnerCode en Serialnumber, zodat het alleen de "eigen" reisinformatie ontvangt.
- *subscription_response/<Version>/2/<SubscriberOwnerCode>/<Serialnumber>*. Het haltesysteem abonneert zich op dit topic met het eigen Type, SubscriberOwnerCode en Serialnumber, zodat het alleen de antwoorden ontvangt op de "eigen" aanmeldingen.
- *unsubscribe/<Version>/2/<SubscriberOwnerCode>/<Serialnumber>*: Het haltesysteem abonneert zich op dit topic met het eigen Type, SubscriberOwnerCode en Serialnumber, zodat het alleen de "eigen" afmeldingen (last will) ontvangt.
- *unsubscribe/<Version>/0/<SubscriberOwnerCode>/<Serialnumber>*: Het haltesysteem abonneert zich op dit topic, waarbij het client id het id is van het distributiesysteem waar het op is aangemeld.

Voor beheerdoeleinden moet een haltesysteem zich verplicht abonneren op de volgende twee topics:

- *inforequest/<Version>/2/<SubscriberOwnerCode>/<serialnumber>* Het haltesysteem abonneert zich op dit topic met het eigen Type, SubscriberOwnerCode en Serialnumber, zodat het alleen de “eigen” informatieverzoeken ontvangt.
- *fileavailable/<Version>/2/<SubscriberOwnerCode>/<serialnumber>* Het haltesysteem abonneert zich op dit topic met het eigen Type, SubscriberOwnerCode en Serialnumber, zodat het alleen de “eigen” configuratiebestanden ontvangt.

Versiebeheer

Dit document en het Protobuf bestand horen bij elkaar en definiëren samen het koppelvlak en het gebruik ervan. Het versienummer bestaat volgens de normale conventie uit drie cijfers. X.Y.Z. Het eerste getal verandert bij een niet backwards compatibele wijziging. Dus als er berichten of velden vervallen, of een ander nummer krijgen. Het tweede cijfer wijzigt als er een backwards compatibele wijziging plaatsvindt, dit omvat ook toevoeging van berichten of velden. In beide gevallen verandert het Protobufbestand. Als er alleen tekstuele wijzigingen plaatsvinden in de koppelvlakbeschrijving zonder dat het Protobufbestand wijzigt, wordt het derde cijfer opgehoogd. De versie van het Protobufbestand bestaat alleen uit de eerste twee cijfers.

versie	datum	wijziging	beschrijving	Protobuf
3.4	14 juni 21	Veld toegevoegd in InfoRequest bericht, waarmee het aantal seconden kan worden opgegeven dat een haltesysteem de wijzigingen van het schermbeeld moet sturen.	✓	✓
		De naam van het PassingTime deel van het bericht is enkelvoud gemaakt (PassingTime ipv PassingTimes), om het consistent te maken met de rest van de berichten.	✓	✓
		Topicnamen systeminfo en statusoverview zonder hoofdletter om het consistent te maken met de andere topicnamen.	✓	
		De vrijeteksttype BLANC is vervallen.	✓	✓
		Het type van het contentdeel van het ScreenContentResponse bericht is veranderd van string naar bytes.		✓
		Hoofdstuk over granulariteit toegevoegd	✓	
3.5	6 aug 21	Het vrijeteksttype OVERRULE is vervallen, waardoor het veld Generalmessage_type geen functie meer heeft. Veld is verwijderd.	✓	✓
4.0	1 maart 22	Bericht AudioTracking toegevoegd om communicatie tussen Haltesystemen binnen een STOPPLACE mogelijk te maken. Inclusief het benodigde Topic. Omdat het haltesysteem moet weten tot welke STOPPLACE het behoort, wordt dit door het distributiesysteem teruggestuurd bij het aanmelden.	✓	✓
		Het PublicNames bericht is uit de Container gehaald naar een apart bericht. Restrictie toegevoegd dat alle stop_codes waarvoor een Haltesysteem zich aanmeldt moeten behoren tot 1 STOPPLACE.	✓	✓
		De enumeratie DATA_CHANGED uit het SubscriptionResponse bericht gehaald. Als het distributiesysteem constateert dat de CHB-gegevens van de halte zijn gewijzigd stuurt het een Unsubscribe bericht. Het haltesysteem meldt zich dan opnieuw aan en ontvangt de gewijzigde haltegegevens.	✓	✓
		Bericht toegevoegd waarmee een dashboard bij het distributiesysteem per leverancier een lijst op kan vragen van alle bekende haltesystemen met de status en tijdstip van laatste wijziging. Ook bericht met de lijst toegevoegd. Incl. topics.	✓	✓
4.1	1 april 22	Veld toegevoegd aan het Containerbericht (flush) dat aangeeft dat de Container een hele nieuwe set aan ritpassages bevat. Het haltesysteem moet dan de oude ritpassages verwijderen.	✓	✓

versie	datum	wijziging	beschrijving	Protobuf
4.2	22 april 22	ShowCancelledTrip van enumeratie naar bool in proto en tekst aangepast in beschrijving.	✓	✓
		In enkele topics de QoS gewijzigd van 2 naar 1.	✓	
		Bericht toegevoegd in de container waarmee eerder verstuurd passeertijden ingetrokken kunnen worden, waarmee het flush veld weer overbodig is geworden.	✓	✓
		Korte toelichting hoe het versienummer in de topic's gebruikt wordt bij migratie naar een nieuwe koppelvlakversie.	✓	
		Beschrijving van Last Will bericht iets verduidelijkt.	✓	
4.3	16 mei 22	De berichtnaam Container en het topic travel_information veranderd naar TravellInfo. Het bericht en topic TravellInfo veranderd naar TravellInfoResponse.	✓	✓
		Het veld generated_timestamp verwijderd uit het bericht GeneralMessageRemove.	✓	✓
		Het veld component_index toegevoegd aan bericht StatusOverview.	✓	✓
		In bericht Metric andere typen waardes (dan int) toegevoegd. Gelijk gemaakt aan bericht SystemInfoLine en StatusOverviewLine	✓	✓
4.4	1 juli 22	In de figuren stond nog "het bericht Container". Dit is veranderd naar TravellInfo	✓	
		In het TravellInfo bericht bij veld trip_stop_status, zin over ritgebonden tekst verwijderd.	✓	
		Bijlage toegevoegd met mapping van treininfo naar TravellInfo bericht.	✓	
		Bij de topics toegevoegd dat antwoorden op het InfoRequest bericht gepubliceerd moeten worden op het topic van het vragende dashboardsysteem.	✓	
		In het Protobuf bestand de velden van het AudioTracking bericht repeated gemaakt.		✓
4.5	8 juli 22	Bericht DetailStatusOverview toegevoegd. Daarmee kunnen leveranciers eigen statusgegevens naar een dashboard sturen. Bijbehorend topic toegevoegd en de enumeratie van request_type uitgebreid.	✓	✓

Bijlage 1: mapping Kv7/Kv8 t.o.v. Openkoppelvlak

In de onderstaande tabel wordt de mapping van de Open koppelvlak velden met de KV7 Planning data beschreven.

De 'pass_time_hash' wordt gegenereerd uit de velden 'DataOwnerCode, LocalServiceLevelCode, LinePlanningNumber, JourneyNumber, FortyfyOrderNummer, UserStopCode, UserStopOrderNumber en OperationDate)

Open Koppelvlak Velden	KV7 planning	Default value
pass_time_hash	<i>Gegenereerde hash</i>	
target_arrival_time	LocalServiceGroupPasstime.TargetArrivalTime	
target_departure_time	LocalServiceGroupPasstime.TargetDepartureTime	
expected_arrival_time	n.v.t.	0
expected_departure_time	n.v.t.	0
number_of_coaches	n.v.t.	0
trip_stop_status	'PLANNED'	
transport_type	Line.TransportType	
wheelchair_accessible	LocalServiceGroupPasstime.WheelChairAccessible	
is_timing_stop	LocalServiceGroupPasstime.IsTimingStop	
stop_code	PassengerStopAssignment.QuayCode	
destinations	Destination.DestinationNameXX (Waarbij XX afhankelijk is van de subscription.) en DestinationDetailXX (Waarbij XX afhankelijk is van de subscription.)	
show_cancelled_trip	n.v.t.	True
block_code	n.v.t.	<lege string>
Occupancy	n.v.t.	0
line_public_number	Line.LinePublicNumber	
side_code	LocalServiceGroupPasstime.SideCode	
line_direction	LocalServiceGroupPasstime.LineDirection	
line_color	LocalServiceGroupPasstime.LineDestColor	
line_text_color	LocalServiceGroupPasstime.LineDestTextColor	
line_icon	LocalServiceGroupPasstime.LineDestIcon	
destination_color	Destination.DestColor	
destination_text_color	Destination.DestTextColor	
destination_icon	Destination.DestIcon	
generated_timestamp	<i>Tijdstip waarop record gegenereerd is.</i>	

De mapping van de openkoppelvlak velden ten opzichte van KV8 berichten wordt in de onderstaande tabel weergegeven.

De 'pass_time_hash' wordt gegenereerd uit de velden 'DataOwnerCode, LocalServiceLevelCode, LinePlanningNumber, JourneyNumber, FortyfyOrderNummer, UserStopCode, UserStopOrderNumber en OperationDate)

Enkele velden worden gevuld uit de CDD_APPLICATION_INFORMATION database. Let op: de time velden worden gevuld met Unix timestamp. Oftewel het aantal seconden sinds 1 januari 1970 00:00:00 in UTC

Open Koppelvlak Velden	KV8 DATEDPASSTIME	Default value
pass_time_hash	<i>Gegenereerde hash</i>	
target arrival time	(KV7)LocalServiceGroupPasstime.TargetArrivalTime	
target departure time	(KV7)LocalServiceGroupPasstime.TargetDepartureTime	
expected arrival time	(KV8)ExpectedArrivalTime	
expected departure time	(KV8)ExpectedDepartureTime	
number of coaches	(KV8)NumberOfCoaches	
trip stop status	(KV8)TripStopStatus	
transport type	(KV7)Line.TransportType	
wheelchair accessible	(KV8)WheelChairAccessible	
is timing stop	(KV8)IsTimingStop	
stop_code	PassengerStopAssignment.QuayCode	
destinations	Destination.DestinationNameXX (Waarbij XX afhankelijk is van de subscription.) en DestinationDetailXX (Waarbij XX afhankelijk is van de subscription.)	
show_cancelled_trip	(KV8)ShowCancelledTrip	
block_code	n.v.t.	<lege string>
Occupancy	n.v.t.	0
line_public_number	(KV7)Line.LinePublicNumber	
side_code	(KV8)SideCode	
line_direction	(KV8)LineDirection	
line_color	(KV8)LineDestColor	
line text color	(KV8)LineDestTextColor	
line icon	(KV8)LineDestIcon	
destination_color	(KV7)Destination.DestColor	
destination text color	(KV7)Destination.DestTextColor	
destination icon	Destination.DestIcon	
generated_timestamp	<i>Tijdstip waarop record gegenereerd is.</i>	

De mapping van de general message is als volgt.

De 'message_hash' wordt gegenereerd uit de velden 'DataOwnerCode, MessageCodeDate, MessageCodeNumber, TimingPointDataOwnerCode, TimingPointCode)

Let op: de time velden worden gevuld met unix timestamp. Oftewel het aantal seconden sinds 1 januari 1970 00:00:00 in UTC

Open Koppelvlak Velden	KV8 Generalmessage	Default value
message_hash	<i>Ge genereerde hash</i>	
message_content	KV8.MessageContent	
message_start_time	KV8.MessageStartTime	
message_end_time	KV8.MessageEndTime	Max unixtimestamp als leeg aangeleverd
generated_timestamp	<i>Tijdstip waarop record gegenereerd is.</i>	
show_overview_display	n.v.t.	True
message_title	n.v.t.	<lege string>
message_priority	Altijd CALAMITY	

Bijlage 2: mapping treininfo naar TravellInfo bericht

Het Distributiesysteem kan ook informatie leveren over vertrekkende treinen. Deze informatie wordt via het standaard TravellInfo bericht verstuurd.

In het Distributiesysteem heeft elk treinstation een (Stopplace)code. Indien een Haltesysteem in zijn subscriptie zo'n code stuurt, zal het Distributiesysteem om de minuut (instelbaar) de vertrektijden van de treinen van dat station opvragen via de NS API.

De vertrektijden van de treinen worden omgezet naar een TravellInfo bericht en verstuurd naar het Haltesysteem. Aangezien de NS API alleen 'DRIVING' en 'ARRIVED' berichten aanlevert, zorgt het Distributiesysteem voor de 'PASSED' berichten. Het Distributiesysteem vergelijkt daartoe de ritten uit een uitvraag met de ritten uit de voorgaande uitvraag. Indien een rit niet meer voorkomt, dan wordt er voor deze rit een 'PASSED' gestuurd.

Invulling velden TravellInfo bericht:

TravellInfo velden	NS API omschrijving
pass_time_hash	Ge genereerde code voor unieke rit.
target_arrival_time	Geplande datum en tijd. (De NS API maakt geen verschil tussen aankomst en vertrektijd. De aankomst en vertrektijd zullen dus altijd hetzelfde zijn.)
target_departure_time	Geplande datum en tijd. (De NS API maakt geen verschil tussen aankomst en vertrektijd. De aankomst en vertrektijd zullen dus altijd hetzelfde zijn.)
expected_arrival_time	Actuele datum en tijd. (De NS API maakt geen verschil tussen aankomst en vertrektijd. De aankomst en vertrektijd zullen dus altijd hetzelfde zijn.)
expected_departure_time	Actuele datum en tijd. (De NS API maakt geen verschil tussen aankomst en vertrektijd. De aankomst en vertrektijd zullen dus altijd hetzelfde zijn.)
trip_stop_status	'INCOMING' uit de API wordt 'DRIVING'. 'ON_STATION' uit de API wordt 'ARRIVED'. Als een rit vervalt, wordt er CANCELED gestuurd. Indien een rit niet meer in de NS API lijst voorkomt dan wordt er een PASSED gestuurd.
transport_type	Is altijd 'TRAIN'
is_timingstop	Is altijd 'true'
stop_code	Bevat de aangevraagde stationscode.
destinations	Zie uitleg 'Bestemming en ritgebonden mededelingen'
show_cancelled_trip	Is altijd 'true'. Dit is omdat de NS API altijd een 'ritgebonden' tekst stuurt als een rit vervalt. Deze moet getoond worden.
line_public_number	Hierin wordt het type trein vermeld 'Sprinter', 'Intercity' en 'Stoptrein'
side_code	Geplande spoornummer
line_icon	De 'OperatorCode' uit de NS-API kan gebruikt worden om een icoontje op het scherm te tonen.
generated_timestamp	Tijdstip van de aanvraag aan de NS API

Bestemming en rit gebonden meldingen.

De NS API geeft bij elke rit een eindbestemming en een lijst met tussenliggende stations. In geval van een verstoring wordt er ook een mededeling gestuurd.

De eindbestemming wordt altijd op de eerste positie in 'destination_name' gezet. De tweede positie van 'destination_name' is leeg.

Indien er geen ritgebonden mededeling is dan zal de eerste positie van 'DestinationDetail' gevuld zijn met een tekst met de tussenliggende stations. De tweede positie van 'destination_name' is dan gevuld met 'DEST'.

Indien er wel een ritgebonden mededeling is dan zal de eerste positie van 'destination_detail' gevuld zijn met de ritgebonden tekst. De tweede positie van 'destination_detail' is dan gevuld met de stijl van het bericht zoals de NS API deze doorgeeft. (De API documentatie geeft helaas geen lijst met welke waarden er mogelijk zijn.)

Lijst met stationscodes:

In de onderstaande lijst zijn de codes weergegeven waarvoor reisinformatie opgevraagd kan worden.

Aanmeld Code	Station Naam	Aanmeld Code	Station Naam
NL:S:NS_HT	's-Hertogenbosch	NL:S:NS_APDM	Apeldoorn De Maten
NL:S:NS_HTO	's-Hertogenbosch Oost	NL:S:NS_APDO	Apeldoorn Osseveld
NL:S:NS_HDE	't Harde	NL:S:NS_APG	Appingedam
NL:S:NS_ATN	Aalten	NL:S:NS_AKL	Arkel
NL:S:NS_AC	Abcoude	NL:S:NS_ARN	Arnhemuiden
NL:S:NS_AKM	Akkrum	NL:S:NS_AH	Arnhem Centraal
NL:S:NS_AMR	Alkmaar	NL:S:NS_AHPR	Arnhem Presikhaaf
NL:S:NS_AMRN	Alkmaar Noord	NL:S:NS_AHP	Arnhem Velperpoort
NL:S:NS_AML	Almelo	NL:S:NS_AHZ	Arnhem Zuid
NL:S:NS_AMRI	Almelo de Riet	NL:S:NS_ASN	Assen
NL:S:NS_ALMB	Almere Buiten	NL:S:NS_BRN	Baarn
NL:S:NS_ALM	Almere Centrum	NL:S:NS_NSCH	Bad Nieuweschans
NL:S:NS_ALMM	Almere Muziekwijk	NL:S:NS_BF	Baflo
NL:S:NS_ALMO	Almere Oostvaarders	NL:S:NS_BRD	Barendrecht
NL:S:NS_ALMP	Almere Parkwijk	NL:S:NS_BNC	Barneveld Centrum
NL:S:NS_AMPO	Almere Poort	NL:S:NS_BNN	Barneveld Noord
NL:S:NS_APN	Alphen a/d Rijn	NL:S:NS_BNZ	Barneveld Zuid
NL:S:NS_AMF	Amersfoort	NL:S:NS_BDM	Bedum
NL:S:NS_AMFS	Amersfoort Schothorst	NL:S:NS_BK	Beek-Elsloo
NL:S:NS_AVAT	Amersfoort Vathorst	NL:S:NS_BSD	Beesd
NL:S:NS_ASA	Amsterdam Amstel	NL:S:NS_BL	Beilen
NL:S:NS_ASB	Amsterdam Bijlmer Arena	NL:S:NS_BGN	Bergen op Zoom
NL:S:NS_ASD	Amsterdam Centraal	NL:S:NS_BET	Best
NL:S:NS_ASHD	Amsterdam Holendrecht	NL:S:NS_BV	Beverwijk
NL:S:NS_ASDL	Amsterdam Lelylaan	NL:S:NS_BHV	Bilthoven
NL:S:NS_ASDM	Amsterdam Muiderpoort	NL:S:NS_BR	Blerick
NL:S:NS_RAI	Amsterdam RAI	NL:S:NS_BLL	Bloemendaal
NL:S:NS_ASSP	Amsterdam Science Park	NL:S:NS_BDG	Bodegraven
NL:S:NS_ASS	Amsterdam Sloterdijk	NL:S:NS_BN	Borne
NL:S:NS_VMW	Amsterdam Van der Madeweg	NL:S:NS_BSK	Boskoop
NL:S:NS_ASDZ	Amsterdam Zuid	NL:S:NS_BSKS	Boskoop Snijdelwijk
NL:S:NS_ANA	Anna Paulowna	NL:S:NS_BHDV	Boven-Hardinxveld
NL:S:NS_APD	Apeldoorn	NL:S:NS_BKF	Bovenkarspel Flora

Aanmeld Code	Station Naam	Aanmeld Code	Station Naam
NL:S:NS_BKG	Bovenkarspel-Grootebroek	NL:S:NS_DRP	Dronryp
NL:S:NS_BMR	Boxmeer	NL:S:NS_DRON	Dronten
NL:S:NS_BTL	Boxtel	NL:S:NS_DVN	Duiven
NL:S:NS_BD	Breda	NL:S:NS_DVD	Duivendrecht
NL:S:NS_BDPB	Breda-Prinsenbeek	NL:S:NS_EC	Echt
NL:S:NS_BKL	Breukelen	NL:S:NS_EDC	Ede Centrum
NL:S:NS_BMN	Brummen	NL:S:NS_ED	Ede-Wageningen
NL:S:NS_BP	Buitenpost	NL:S:NS_EEM	Eemshaven
NL:S:NS_BDE	Bunde	NL:S:NS_EDN	Eijsden
NL:S:NS_BNK	Bunnik	NL:S:NS_EHV	Eindhoven
NL:S:NS_BSMZ	Bussum Zuid	NL:S:NS_EHS	Eindhoven Strijp-S
NL:S:NS_CPS	Capelle Schollevaar	NL:S:NS_EST	Elst
NL:S:NS_CAS	Castricum	NL:S:NS_EMN	Emmen
NL:S:NS_CVM	Chevremont	NL:S:NS_EMNZ	Emmen Zuid
NL:S:NS_CO	Coevorden	NL:S:NS_EKZ	Enkhuizen
NL:S:NS_CK	Cuijk	NL:S:NS_ES	Enschede
NL:S:NS_CL	Culemborg	NL:S:NS_ESE	Enschede De Eschmarke
NL:S:NS_DA	Daarlerveen	NL:S:NS_ESK	Enschede Kennispark
NL:S:NS_DLN	Dalen	NL:S:NS_EML	Ermelo
NL:S:NS_DL	Dalfsen	NL:S:NS_ETN	Etten-Leur
NL:S:NS_DVNK	De Vink	NL:S:NS_EGH	Eygelshoven
NL:S:NS_DWE	De Westereen	NL:S:NS_EGHM	Eygelshoven Markt
NL:S:NS_DEI	Deinum	NL:S:NS_WTM	Eys-Wittem
NL:S:NS_DDN	Delden	NL:S:NS_FWD	Feanwâlden
NL:S:NS_DT	Delft	NL:S:NS_FN	Franeker
NL:S:NS_DTZ	Delft Zuid	NL:S:NS_GDR	Gaanderen
NL:S:NS_DZ	Delfzijl	NL:S:NS_GDM	Geldermalsen
NL:S:NS_DZW	Delfzijl West	NL:S:NS_GP	Geldrop
NL:S:NS_DLD	Den Dolder	NL:S:NS_GLN	Geleen Oost
NL:S:NS_GVC	Den Haag Centraal	NL:S:NS_LUT	Geleen-Lutterade
NL:S:NS_GV	Den Haag HS	NL:S:NS_GZ	Gilze-Rijen
NL:S:NS_LAA	Den Haag Laan v NOI	NL:S:NS_GBR	Glanerbrug
NL:S:NS_GVM	Den Haag Mariahoeve	NL:S:NS_GS	Goes
NL:S:NS_GVMW	Den Haag Moerwijk	NL:S:NS_GO	Goor
NL:S:NS_YPB	Den Haag Ypenburg	NL:S:NS_GR	Gorinchem
NL:S:NS_HDR	Den Helder	NL:S:NS_GD	Gouda
NL:S:NS_HDRZ	Den Helder Zuid	NL:S:NS_GDG	Gouda Goverwelle
NL:S:NS_DN	Deurne	NL:S:NS_GBG	Gramsbergen
NL:S:NS_DV	Deventer	NL:S:NS_GK	Grijpskerk
NL:S:NS_DVC	Deventer Colmschate	NL:S:NS_GN	Groningen
NL:S:NS_DID	Didam	NL:S:NS_GERP	Groningen Europapark
NL:S:NS_DMN	Diemen	NL:S:NS_GNN	Groningen Noord
NL:S:NS_DMNZ	Diemen Zuid	NL:S:NS_GW	Grou-Jirnsum
NL:S:NS_DR	Dieren	NL:S:NS_WIJ	Gulpen-Wijlre
NL:S:NS_DTC	Doetinchem	NL:S:NS_HLM	Haarlem
NL:S:NS_DTCH	Doetinchem De Huet	NL:S:NS_HLMS	Haarlem Spaarnwoude
NL:S:NS_DDR	Dordrecht	NL:S:NS_HWZB	Halfweg-Zwanenburg
NL:S:NS_DDRS	Dordrecht Stadspolders	NL:S:NS_HDB	Hardenberg
NL:S:NS_DDZD	Dordrecht Zuid	NL:S:NS_HD	Harderwijk
NL:S:NS_DB	Driebergen-Zeist	NL:S:NS_HBZM	Hardinxveld Blauwe Zoom
NL:S:NS_DRH	Driehuis	NL:S:NS_GND	Hardinxveld-Giessendam

Aanmeld Code	Station Naam	Aanmeld Code	Station Naam
NL:S:NS_HRN	Haren	NL:S:NS_KMW	Koudum-Molkwerum
NL:S:NS_HRNOVT	Haren OV Transferium	NL:S:NS_KBD	Krabbendijke
NL:S:NS_HLG	Harlingen	NL:S:NS_KMA	Krommenie-Assendelft
NL:S:NS_HLGH	Harlingen Haven	NL:S:NS_KW	Kropswolde
NL:S:NS_HK	Heemskerk	NL:S:NS_KRG	Kruiningen-Yerseke
NL:S:NS_HAD	Heemstede-Aerdenhout	NL:S:NS_ZLW	Lage Zwaluwe
NL:S:NS_HR	Heerenveen	NL:S:NS_LG	Landgraaf
NL:S:NS_HRY	Heerenveen IJstadion	NL:S:NS_LLZM	Lansingerland-Zoetermeer
NL:S:NS_HWD	Heerhugowaard	NL:S:NS_LDM	Leerdam
NL:S:NS_HRL	Heerlen	NL:S:NS_LW	Leeuwarden
NL:S:NS_HRLW	Heerlen Woonboulevard	NL:S:NS_LWC	Leeuwarden Camminghaburen
NL:S:NS_HZE	Heeze	NL:S:NS_LEDN	Leiden Centraal
NL:S:NS_HLO	Heiloo	NL:S:NS_LDL	Leiden Lammenschans
NL:S:NS_HNO	Heino	NL:S:NS_LEDT	Leiden Transferium
NL:S:NS_HM	Helmond	NL:S:NS_LLS	Lelystad Centrum
NL:S:NS_HMH	Helmond 't Hout	NL:S:NS_LTV	Lichtenvoorde-Groenlo
NL:S:NS_HMBV	Helmond Brandevoort	NL:S:NS_LC	Lochem
NL:S:NS_HMBH	Helmond Brouwhuis	NL:S:NS_LP	Loppersum
NL:S:NS_HMN	Hemmen-Dodewaard	NL:S:NS_LTN	Lunteren
NL:S:NS_HGL	Hengelo	NL:S:NS_MZ	Maarheeze
NL:S:NS_HGLG	Hengelo Gezondheidspark	NL:S:NS_MRN	Maarn
NL:S:NS_HGLO	Hengelo Oost	NL:S:NS_MAS	Maarssen
NL:S:NS_HIL	Hillegom	NL:S:NS_MT	Maastricht
NL:S:NS_HVS	Hilversum	NL:S:NS_MTN	Maastricht Noord
NL:S:NS_HVSM	Hilversum Media Park	NL:S:NS_MTR	Maastricht Randwyck
NL:S:NS_HVSP	Hilversum Sportpark	NL:S:NS_MG	Mantgum
NL:S:NS_HNP	Hindeloopen	NL:S:NS_MRB	Mariënberg
NL:S:NS_HB	Hoensbroek	NL:S:NS_MTH	Martenshoek
NL:S:NS_HVL	Hoevelaken	NL:S:NS_MES	Meerssen
NL:S:NS_HOR	Hollandsche Rading	NL:S:NS_MP	Meppel
NL:S:NS_HON	Holten	NL:S:NS_MDB	Middelburg
NL:S:NS_HFD	Hoofddorp	NL:S:NS_MMLH	Mook Molenhoek
NL:S:NS_HGV	Hoogeveen	NL:S:NS_NDB	Naarden-Bussum
NL:S:NS_HGZ	Hoogezand-Sappemeer	NL:S:NS_NA	Nieuw Amsterdam
NL:S:NS_HKS	Hoogkarspel	NL:S:NS_NVP	Nieuw Vennep
NL:S:NS_HN	Hoorn	NL:S:NS_NWK	Nieuwerkerk a/d IJssel
NL:S:NS_HNK	Hoorn Kersenboogerd	NL:S:NS_NKK	Nijkerk
NL:S:NS_HRT	Horst-Sevenum	NL:S:NS_NM	Nijmegen
NL:S:NS_HTN	Houten	NL:S:NS_NMD	Nijmegen Dukenburg
NL:S:NS_HTNC	Houten Castellum	NL:S:NS_NMGO	Nijmegen Goffert
NL:S:NS_SGL	Houthem-St. Gerlach	NL:S:NS_NMH	Nijmegen Heyendaal
NL:S:NS_HDG	Hurdegaryp	NL:S:NS_NML	Nijmegen Lent
NL:S:NS_IJT	IJlst	NL:S:NS_NVD	Nijverdal
NL:S:NS_KPN	Kampen	NL:S:NS_NS	Nunspeet
NL:S:NS_KPNZ	Kampen Zuid	NL:S:NS_NH	Nuth
NL:S:NS_BZL	Kapelle-Biezelingse	NL:S:NS_OBD	Obdam
NL:S:NS_KRD	Kerkrade Centrum	NL:S:NS_OT	Oisterwijk
NL:S:NS_KTR	Kesteren	NL:S:NS_ODZ	Oldenzaal
NL:S:NS_KBK	Klarenbeek	NL:S:NS_OST	Olst
NL:S:NS_KMR	Klimmen-Ransdaal	NL:S:NS_OMN	Ommen
NL:S:NS_KZ	Koog aan de Zaan	NL:S:NS_OTB	Oosterbeek

Aanmeld Code	Station Naam	Aanmeld Code	Station Naam
NL:S:NS_OP	Opheusden	NL:S:NS_STV	Stavoren
NL:S:NS_O	Oss	NL:S:NS_STM	Stedum
NL:S:NS_OW	Oss West	NL:S:NS_SWK	Steenwijk
NL:S:NS_ODB	Oudenbosch	NL:S:NS_SRN	Susteren
NL:S:NS_OVN	Overveen	NL:S:NS_SM	Swalmen
NL:S:NS_PMR	Purmerend	NL:S:NS_TG	Tegelen
NL:S:NS_PMO	Purmerend Overwhere	NL:S:NS_TBG	Terborg
NL:S:NS_PMW	Purmerend Weidevenne	NL:S:NS_TL	Tiel
NL:S:NS_PT	Putten	NL:S:NS_TPSW	Tiel Passewaaij
NL:S:NS_RAT	Raalte	NL:S:NS_TB	Tilburg
NL:S:NS_RVS	Ravenstein	NL:S:NS_TBR	Tilburg Reeshof
NL:S:NS_RV	Reuver	NL:S:NS_TBU	Tilburg Universiteit
NL:S:NS_RH	Rheden	NL:S:NS_TWL	Twello
NL:S:NS_RHN	Rhenen	NL:S:NS_UTG	Uitgeest
NL:S:NS_RSN	Rijssen	NL:S:NS_UHZ	Uithuizen
NL:S:NS_RSW	Rijswijk	NL:S:NS_UHM	Uithuizermeeden
NL:S:NS_RB	Rilland-Bath	NL:S:NS_UST	Usquert
NL:S:NS_RM	Roermond	NL:S:NS_UT	Utrecht Centraal
NL:S:NS_RD	Roodeschool	NL:S:NS_UTLR	Utrecht Leidsche Rijn
NL:S:NS_RSD	Roosendaal	NL:S:NS_UTLN	Utrecht Lunetten
NL:S:NS_RS	Rosmalen	NL:S:NS_UTM	Utrecht Maliebaan
NL:S:NS_RTA	Rotterdam Alexander	NL:S:NS_UTO	Utrecht Overvecht
NL:S:NS_RTB	Rotterdam Blaak	NL:S:NS_UTT	Utrecht Terwijde
NL:S:NS_RTD	Rotterdam Centraal	NL:S:NS_UTVR	Utrecht Vaartsche Rijn
NL:S:NS_RLB	Rotterdam Lombardijen	NL:S:NS_UTZL	Utrecht Zuilen
NL:S:NS_RTN	Rotterdam Noord	NL:S:NS_VK	Valkenburg
NL:S:NS_RTST	Rotterdam Stadion	NL:S:NS_VSV	Varsseveld
NL:S:NS_RTZ	Rotterdam Zuid	NL:S:NS_VDM	Veendam
NL:S:NS_RL	Ruurlo	NL:S:NS_VNDC	Veenendaal Centrum
NL:S:NS_SPTN	Santpoort Noord	NL:S:NS_VNDW	Veenendaal West
NL:S:NS_SPTZ	Santpoort Zuid	NL:S:NS_KLP	Veenendaal-De Klomp
NL:S:NS_SPM	Sappemeer Oost	NL:S:NS_VP	Velp
NL:S:NS_SSH	Sassenheim	NL:S:NS_VL	Venlo
NL:S:NS_SWD	Sauwerd	NL:S:NS_VRY	Venray
NL:S:NS_SGN	Schagen	NL:S:NS_VLB	Vierlingsbeek
NL:S:NS_SDA	Scheemda	NL:S:NS_VTN	Vleuten
NL:S:NS_SDM	Schiedam Centrum	NL:S:NS_VS	Vlissingen
NL:S:NS_SOG	Schin op Geul	NL:S:NS_VSS	Vlissingen Souburg
NL:S:NS_SN	Schinnen	NL:S:NS_VDL	Voerendaal
NL:S:NS_SHL	Schiphol Airport	NL:S:NS_VB	Voorburg
NL:S:NS_SPV	Simpelveld	NL:S:NS_VH	Voorhout
NL:S:NS_STD	Sittard	NL:S:NS_VST	Voorschoten
NL:S:NS_SDT	Sliedrecht	NL:S:NS_VEM	Vorst-Empe
NL:S:NS_SDTB	Sliedrecht Baanhoek	NL:S:NS_VD	Vorden
NL:S:NS_SK	Sneek	NL:S:NS_VZ	Vriezenveen
NL:S:NS_SKND	Sneek Noord	NL:S:NS_VHP	Vroomshoop
NL:S:NS_ST	Soest	NL:S:NS_VG	Vught
NL:S:NS_STZ	Soest Zuid	NL:S:NS_WAD	Waddinxveen
NL:S:NS_SD	Soestdijk	NL:S:NS_WADN	Waddinxveen Noord
NL:S:NS_SBK	Spaubeek	NL:S:NS_WADT	Waddinxveen Triangel
NL:S:NS_SPH	Spekholzerheide	NL:S:NS_WFM	Warffum
NL:S:NS_RTNG	SSN Stoomdepot	NL:S:NS_WT	Weert

Aanmeld Code	Station Naam	Aanmeld Code	Station Naam
NL:S:NS_WP	Weesp		
NL:S:NS_WL	Wehl		
NL:S:NS_WTV	Westervoort		
NL:S:NS_WZ	Wezep		
NL:S:NS_WDN	Wierden		
NL:S:NS_WC	Wijchen		
NL:S:NS_WH	Wijhe		
NL:S:NS_WS	Winschoten		
NL:S:NS_WSM	Winsum		
NL:S:NS_WW	Winterswijk		
NL:S:NS_WWW	Winterswijk West		
NL:S:NS_WD	Woerden		
NL:S:NS_WF	Wolfheze		
NL:S:NS_WV	Wolvega		
NL:S:NS_WK	Workum		
NL:S:NS_WM	Wormerveer		
NL:S:NS_ZD	Zaandam		
NL:S:NS_ZDK	Zaandam Kogerveld		
NL:S:NS_ZZS	Zaandijk Zaanse Schans		
NL:S:NS_ZBM	Zaltbommel		
NL:S:NS_ZVT	Zandvoort aan Zee		
NL:S:NS_ZA	Zetten-Andelst		
NL:S:NS_ZV	Zevenaar		
NL:S:NS_ZVB	Zevenbergen		
NL:S:NS_ZTM	Zoetermeer		
NL:S:NS_ZTMO	Zoetermeer Oost		
NL:S:NS_ZB	Zuidbroek		
NL:S:NS_ZH	Zuidhorn		
NL:S:NS_ZP	Zutphen		
NL:S:NS_ZWD	Zwijndrecht		
NL:S:NS_ZL	Zwolle		
NL:S:NS_ZLSH	Zwolle Stadshagen		