



Technisch ontwerp Routing

1.1

Datum	22-01-2016
Status	Definitief

Inhoudsopgave

1	INLEIDING	4
1.1	Beknopte omschrijving.....	4
1.2	Referenties	4
2	CONTEXT	5
2.1	Omgeving	5
2.2	Ontwerpbeslissingen.....	5
2.2.1	<i>Communicatie</i>	5
2.2.2	<i>Tooling</i>	5
3	ROUTERING CENTRALE	6
3.1	Maven Structuur	6
3.2	Queues en topics	6
3.3	Onverwerkbaar berichten	6
3.3.1	<i>Opnieuw verwerken van DLQ berichten</i>	7
4	OPMERKINGEN	8
5	BIJLAGE A: JMX INTERFACE	9
5.1	Active MQ	10
6	BIJLAGE B: CONFIGURATIE	11
7	BIJLAGE C: GEMEENTE- EN AUTORISATIEREGISTER CLIENT	12
7.1	Maven Structuur	12
7.2	Implementatie	12
7.2.1	<i>Autorisatieregister</i>	12
7.2.2	<i>Gemeenteregister</i>	13

Versiehistorie

Datum	Versie	Omschrijving	Auteur
02-10-2015	1.0	Initiële versie	Operatie BRP
22-01-2016	1.1	Nalopen voor release; kleine fouten bijwerken; autorisatieregister opnemen	Operatie BRP

Reviewhistorie

Datum	Versie	Omschrijving	Reviewers
02-10-2015	1.0	Initiële review	Operatie BRP

1 Inleiding

1.1 Beknopte omschrijving

Het component Routing verzorgt de centrale communicatie tussen de componenten van de migratievoorzieningen ([VOISC], [ISC] en [SYNC]).

1.2 Referenties

#	Document	Organisatie	Versie	Datum
[SAD]	Software Architecture Document Migratiesoftware BRP	OperatieBRP		
[AMQ_JMX]	http://activemq.apache.org/jmx.html			
[VOISC]	TO Voisc	OperatieBRP		
[ISC]	TO ISC	OperatieBRP		
[SYNC]	TO BRP Synchronisatie Service	OperatieBRP		

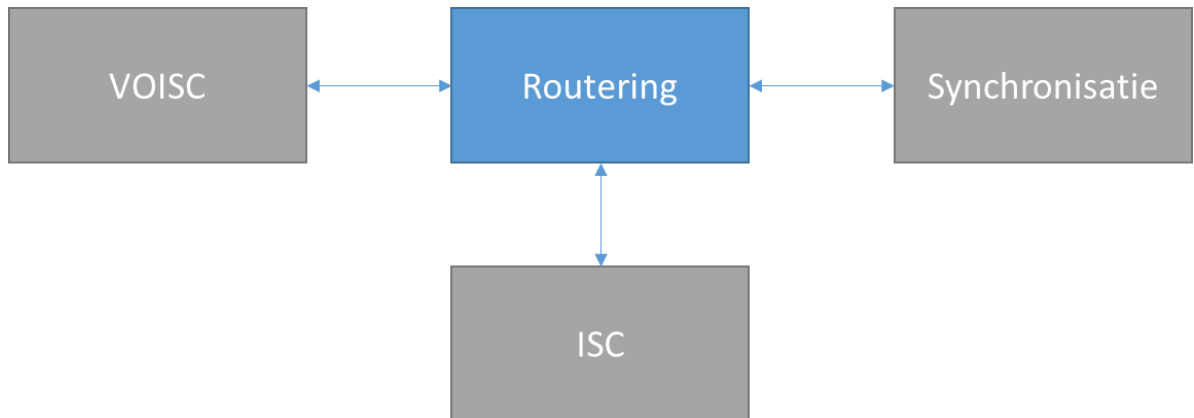
Nota: versienummers van 'interne' documenten worden niet opgenomen. Deze dienen consistent te zijn.

2 Context

De routing centrale is het centrale component binnen de migratievoorzieningen waarmee de verschillende componenten met elkaar communiceren.

2.1 Omgeving

Zoals beschreven in het [SAD] bevindt de routing centrale zich als communicatie kanaal tussen de verschillende componenten van de migratievoorzieningen.



2.2 Ontwerpbeslissingen

2.2.1 Communicatie

De communicatie tussen componenten wordt gerealiseerd door middel van JMS-queues en topics om berichten asynchroon te versturen en te ontvangen. Door deze communicatie te ontkoppelen kunnen de verschillende componenten van de migratievoorzieningen (VOISC, ISC en Synchronisatie Service) los van elkaar uitgevoerd worden.

2.2.2 Tooling

Om de routing centrale te realiseren is gekozen om de functionaliteit van Apache ActiveMQ te gebruiken. ActiveMQ wordt reeds gebruikt binnen het BRP systeemlandschap en hiermee sluiten de migratievoorzieningen aan op de reeds gekozen oplossing.

Voor persistentie is gekozen om de berichten op te slaan in een database. Binnen Operatie BRP is gekozen voor de PostgreSQL database. Door de berichten op te slaan in een database kan gebruik worden gemaakt van een uniforme manier van backup over alle persistente gegevens heen.

3 Routing Centrale

3.1 Maven Structuur

Het onderstaande overzicht toont de Maven structuur en geeft aan waar elk project voor wordt gebruikt.

Project	Subproject	Beschrijving
migr-routing		
	migr-routing-runtime	Bevat alle routing centrale functionaliteit

3.2 Queues en topics

De volgende queues en topics worden beschikbaar gemaakt binnen de routing centrale:

Naam	Type	Doel
sync.verzoek	Queue	Berichten van ISC naar de Synchronisatie Service
sync.antwoord	Queue	Berichten van de Synchronisatie Service naar ISC
voisc.ontvangst	Queue	Berichten van VOISC naar ISC
voisc.verzenden	Queue	Berichten van ISC naar VOISC
levering	Queue	Berichten voor Levering (van BRP) naar ISC
archivering	Queue	Berichten voor Archivering (van VOISC) naar de Synchronisatie Service
gemeente.verzoek	Queue	Verzoeken voor het Gemeenteregister (van ISC of VOISC) naar de Synchronisatie Service
gemeente.register	Topic	Publicatie van het Gemeenteregister (voor ISC of VOISC) door de Synchronisatie Service
autorisatie.verzoek	Queue	Verzoeken voor het Autorisatieregister (van ISC) naar de Synchronisatie Service
autorisatie.register	Topic	Publicatie van het Autorisatieregister (voor ISC) door de Synchronisatie Service

De communicatie tussen ISC en VOISC, en ISC en de Synchronisatie Service wordt ondersteund door combinaties van queues. Communicatie waarbij geen antwoord wordt gegeven, zoals voor Levering en Archivering wordt ondersteund door een enkele queue.

Voor het publiceren van een register wordt gebruik gemaakt van een combinatie van een queue en een topic. Van de afnemer van een register wordt verwacht dat naar het topic wordt geluisterd voor de publicatie van het register en dat een verzoek voor het register wordt verstuurd naar de queue. Hierdoor wordt bereikt dat een register centraal kan worden 'gepushed' naar de afnemers en dat de verschillende afnemers van het register allen hetzelfde register gebruiken.

3.3 Onverwerkbare berichten

Indien berichten niet verwerkt kunnen worden (in andere woorden als de transactie teruggedropt wordt) bevat ActiveMQ een automatisch herhalingsmechanisme waardoor het bericht opnieuw wordt aangeboden. Echter indien na een aantal (vijf) herhalingen het bericht opnieuw niet verwerkt kan worden, wordt het bericht, om overbelasting van het systeem te voorkomen,

verplaatst naar een zogenaamde 'dead letter queue' (DLQ). Per queue is een aparte DLQ geconfigureerd met dezelfde naam als de queue met daarachter de suffix '.dlq'.

3.3.1 Opnieuw verwerken van DLQ berichten

De berichten die op de DLQ zijn geplaatst kunnen enkel handmatig opnieuw in het systeem worden gebracht en duiden op een structurele onverwachte fout in het systeem. Om dit proces te ondersteunen kan gebruik worden gemaakt van de JMX operatie 'redeliverDlq' om alle berichten van een DLQ opnieuw op de 'gewone' queue te plaatsen.

4 Opmerkingen

Instellingen wat betreft fail-over en clustering dienen nog te worden bepaald.

5 Bijlage A: JMX interface

Ten behoeve van externe toegang is een JMX interface beschikbaar waarmee enkele handelingen uitgevoerd kunnen worden

De standaard JMX voorziening van Java wordt gebruikt om de JMX interface beschikbaar te stellen. Om hier mee te kunnen verbinden worden bij het opstarten van de applicatie de volgende systeem instellingen (properties) gebruikt.

Instelling	Waarde	Omschrijving
com.sun.management.jmxremote		Indicatie om aan te geven dat de JMX interface wordt gebruikt
com.sun.management.jmxremote.port	1099	Poort voor JMX
com.sun.management.jmxremote.rmi.port	1099	Poort voor RMI
com.sun.management.jmxremote.ssl	false	Indicatie om aan te geven dat geen SSL wordt gebruikt
com.sun.management.jmxremote.authenticate	false	Indicatie om aan te geven dat geen authenticatie wordt gebruikt
com.sun.management.jmxremote.local.only	false	Indicatie om aan te geven dat niet enkel lokaal verbonden mag worden

Nota: de poort instellingen kunnen veranderd worden indien deze poorten reeds gebruikt worden door een andere applicatie.

5.1 Routing

nl.bzk.migratiebrp.routing:name=ROUTERING		
Operatie	Parameters	Omschrijving
afsluiten	-	Applicatie afsluiten.
redeliverDlq	queueNaam (String)	Alle berichten op een DLQ (queueNaam + '.dlq') verplaatsen naar de 'gewone' queue (queueNaam).

Class
Jmx
JmxImpl

5.2 Active MQ

ActiveMQ stelt ook een JMX interface beschikbaar om informatie te verschaffen over de queues en topics (zie [AMQ_JMX]).

Deze wordt binnen de Routing centrale beschikbaar gesteld onder de naam:

org.apache.activemq:type=Broker,brokerName=routingCentrale

Met deze interface kan onder andere worden bepaald hoeveel berichten er op een queue of topic staan:

org.apache.activemq:type=Broker,brokerName=routingCentrale, destinationType=Queue,destinationName=<<Queue naam>>	
Attribuut	Omschrijving
QueueSize	Aantal berichten op de queue

org.apache.activemq:type=Broker,brokerName=routingCentrale, destinationType=Topic,destinationName=<<Topic naam>>	
Attribuut	Omschrijving
QueueSize	Aantal berichten op het topic

6

Bijlage B: Configuratie

De volgende instellingen zijn beschikbaar in het configuratie bestand 'routing-runtime.properties':

Instelling	Standaard waarde	Omschrijving
routing.database.driver	org.postgresql.Driver	JDBC database driver
routing.database.url	jdbc:postgresql://localhost:5432/routing	JDBC database verbinding (host, port en database naam)
routing.database.username	postgres	Database gebruikersnaam
routing.database.password	postgres	Database wachtwoord
routing.database.minpool	3	Minimum aantal database verbindingen
routing.database.maxpool	20	Maximum aantal database verbindingen
routing.activemq.url	nio://localhost:61616	Active MQ poort (waarop de andere componenten kunnen verbinden)
routing.activemq.data.directory	work/activemq-data	Active MQ data directory (relatief)
routing.activemq.scheduler.directory	work/activemq-scheduler	Active MQ scheduler directory (relatief)
routing.activemq.memory.usage	32 mb	Active MQ geheugen gebruik (memoryUsage)
routing.activemq.store.usage	32 mb	Active MQ disk gebruik (storeUsage)
routing.activemq.temp.usage	32 mb	Active MQ temp gebruik (tempUsage)
routing.queue.sync.verzoek	sync.verzoek	Naam voor de sync.verzoek queue
routing.queue.sync.antwoord	sync.antwoord	Naam voor de sync.antwoord queue
routing.queue.voisc.ontvangst	voisc.ontvangst	Naam voor de voisc.ontvangst queue
routing.queue.voisc.verzenden	voisc.verzenden	Naam voor de voisc.verzenden queue
routing.queue.levering	levering	Naam voor de levering queue
routing.queue.archivering	archivering	Naam voor de archivering queue
routing.queue.gemeente.verzoek	gemeente.verzoek	Naam voor de gemeenteregister verzoek queue
routing.topic.gemeente.register	gemeente.register	Naam voor het gemeenteregister topic
routing.queue.autorisatie.verzoek	autorisatie.verzoek	Naam voor de autorisatieregister verzoek queue
routing.topic.autorisatie.register	autorisatie.register	Naam voor het autorisatieregister topic

7 Bijlage C: Gemeente- en autorisatieregister client

Om het gebruik van het gemeente- en autorisatieregister te ondersteunen is een algemene bibliotheek beschikbaar.

7.1 Maven Structuur

Het onderstaande overzicht toont de Maven structuur en geeft aan waar elk project voor wordt gebruikt.

Project	Subproject	Beschrijving
migr-register		
	migr-register-client	Algemene functionaliteit voor het afnemen van een register

7.2 Implementatie

Package	Omschrijving
<code>nl.bzk.migratiebrp.register.client</code>	Package met de algemene register client functionaliteit

Voor het afnemen van een register dient gebruik gemaakt te worden van een queue en een topic. Om deze specifieke werkwijze te abstraheren is een algemene implementatie beschikbaar waarin dit transparant wordt opgelost.

Class
<code>AbstractRegisterServiceImpl</code> <code>RegisterService</code>

Er wordt gebruik gemaakt van de standaard Spring functionaliteit om te communiceren met queues en topics (`JmsTemplate` en `DefaultMessageListenerContainer`).

7.2.1 Autorisatieregister

Class
<code>AutorisatieService</code> <code>AutorisatieServiceImpl</code> <code>AutorisatieServiceJMX</code>

De autorisatieregister client wordt ondersteund door een spring configuratie (`classpath:/register-client-autorisatie.xml`). Met deze configuratie wordt een instantie van de `AutorisatieService` interface geconfigureerd onder de naam 'autorisatieRegisterService'. Hierbij worden de volgende geconfigureerde beans verwacht: `autorisatieConnectionFactory` (JMS `ConnectionFactory`), `autorisatieVerzoekQueue` (JMS `Destination`) en `autorisatieRegisterTopic` (JMS `Destination`).

Het autorisatieregister wordt ook ondersteund door een JMX interface (let op: de naam kan afwijkend worden geconfigureerd indien nodig).

nl.bzk.migratiebrp.register:name=AUTORISATIE		
Operatie	Parameters	Omschrijving
refreshRegister	-	Register opnieuw opvragen
clearRegister	-	Cache legen (zodat bij het volgende gebruik het register opnieuw wordt opgevraagd)
isGestart	-	Geeft aan of register berichten worden ontvangen
start	-	Start met het ontvangen van register berichten
stop	-	Stop met het ontvangen van register berichten

7.2.2 Gemeenteregister

Class
GemeenteService GemeenteServiceImpl GemeenteServiceJMX

De gemeenteregister client wordt ondersteund door een spring configuratie (classpath:/register-client-gemeente.xml). Met deze configuratie wordt een instantie van de GemeenteService interface geconfigureerd onder de naam 'gemeenteRegisterService'. Hierbij worden de volgende geconfigureerde beans verwacht: gemeenteConnectionFactory (JMS ConnectionFactory), gemeenteVerzoekQueue (JMS Destination) en gemeenteRegisterTopic (JMS Destination).

Het autorisatieregister wordt ook ondersteund door een JMX interface (let op: de naam kan afwijkend worden geconfigureerd indien nodig).

nl.bzk.migratiebrp.register:name=GEMEENTE		
Operatie	Parameters	Omschrijving
refreshRegister	-	Register opnieuw opvragen
clearRegister	-	Cache legen (zodat bij het volgende gebruik het register opnieuw wordt opgevraagd)
isGestart	-	Geeft aan of register berichten worden ontvangen
start	-	Start met het ontvangen van register berichten
stop	-	Stop met het ontvangen van register berichten