



Ministerie van Binnenlandse Zaken en
Koninkrijksrelaties

Handleiding ordentelijk opstarten

1.0

Datum	23-08-2017
Status	Definitief

Inhoudsopgave

INLEIDING	6
DEEL A	7
1 INLEIDING.....	8
1.1 <i>Benodigde kennis</i>	8
1.2 <i>Technische vereisten</i>	8
2 GEREED MAKEN OMGEVING.....	9
2.1 <i>Installatie VirtualBox</i>	9
2.2 <i>Toevoegen machine</i>	9
2.3 <i>Starten omgeving</i>	9
3 ALGEMENE INFORMATIE OVER DE OMGEVING	10
4 TESTS UITVOEREN.....	11
4.1 <i>Starten SoapUI</i>	11
4.2 <i>Uitvoeren bevraging Zoek persoon</i>	11
4.3 <i>Uitvoeren bevraging Geef details persoon</i>	11
4.4 <i>Uitvoeren bijhouding verhuizing</i>	12
DEEL B	13
5 INLEIDING.....	14
5.1 <i>Benodigde kennis</i>	14
5.2 <i>Versies</i>	14
5.2.1 <i>Overige systeemsoftware</i>	14
6 TOOLING.....	15
6.1 <i>Java</i>	15
6.2 <i>Maven</i>	15
6.2.1 <i>Configuratie maven</i>	15
6.3 <i>NPM (NodeJS)</i>	16
6.4 <i>Docker</i>	16
7 APPLICATIE SAMENSTELLEN	17
7.1 <i>Uitgangssituatie</i>	17
7.2 <i>Database definitie</i>	17
7.3 <i>Koppelvlak definitie</i>	17
7.4 <i>Externe bibliotheken</i>	18
7.5 <i>Operatie BRP</i>	18
7.5.1 <i>Hergebruikte GBA-v code</i>	21
7.5.2 <i>Testdata levering</i>	21
7.5.3 <i>Operatie BRP</i>	22
7.5.4 <i>Geconverteerde GBA gegevens (optioneel)</i>	22
8 APPLICATIE TESTEN	23
8.1 <i>Integratie testen initiële vulling</i>	23
8.2 <i>Integratie testen migratie voorzieningen</i>	23
8.3 <i>Integratie testen bijhouding</i>	23
8.4 <i>Integratie testen levering</i>	24
9 BIJLAGE A – OVERIGE TESTEN	25
9.1 <i>Regressietesten Migratievoorzieningen</i>	25
9.1.1 <i>Regressie component GBA Afnemerindicaties in BRP</i>	25
9.1.2 <i>Regressie component GBA Archivering in BRP</i>	25
9.1.3 <i>Regressie component BRP Beheer</i>	25
9.1.4 <i>Regressie component GBA Bevraging in BRP</i>	25
9.1.5 <i>Regressie component GBA Bijhouding in BRP</i>	25
9.1.6 <i>Regressie component GBA expressies in BRP</i>	25
9.1.7 <i>Regressie component GBA Levering in BRP</i>	25
9.1.8 <i>Regressie component GBA Selectie in BRP</i>	25

9.1.9	Regressie component GBA Sleutelrubrieken in BRP.....	26
9.1.10	Regressie component GBA Vrijbericht in BRP.....	26
9.1.11	Regressie component ISC Beheer.....	26
9.1.12	Regressie component ISC.....	26
9.1.13	Regressie component IV-Logging.....	26
9.1.14	Regressie component IV-Naar BRP.....	26
9.1.15	Regressie component IV-Naar LO3.....	26
9.1.16	Regressie component Synchronisatie (IV-modus).....	26
9.1.17	Regressie component Synchronisatie (synchronisatie-modus).....	27
9.1.18	Regressie component VOISC.....	27
9.2	<i>Integratietesten Migratievoorzieningen</i>	28
9.2.1	Integratie IV Personen (uc101).....	28
9.2.2	Integratie IV Autorisaties (uc102).....	28
9.2.3	Integratie IV Afnemerindicaties (uc103).....	28
9.2.4	Integratie IV Protocollering (uc104).....	28
9.2.5	Integratie IV configuratie (uc105).....	28
9.2.6	Integratie leveringsvergelijking.....	28
9.3	<i>Conversietesten Migratievoorzieningen</i>	28
9.3.1	Regressie component Persoon NaarBrp.....	28
9.3.2	Regressie component Persoon NaarLO3.....	29
9.3.3	Regressie component Persoon Preconditie.....	29
9.4	<i>Bijhouding</i>	29
9.5	<i>Levering</i>	29
10	BIJLAGE B – SONARQUBE	30
10.1	<i>Configuratie SonarQube omgeving</i>	30
10.1.1	Quality Profiles.....	30
10.1.2	Quality Gate.....	31
10.1.3	Uitvoeren kwaliteitscontrole.....	31
10.1.4	Inlezen 'explain'.....	32

Versiehistorie

Datum	Versie	Omschrijving	Auteur
17-08-2017	1.0	Initiële versie	Operatie BRP

Reviewhistorie

Datum	Versie	Omschrijving	Reviewers

Inleiding

Deze handleiding beschrijft de werkwijze om de broncode van Operatie BRP te bouwen, uitrollen en te testen. Omdat het gehele proces om van broncode naar een werkend systeem te komen een groot aantal stappen kent, is de procedure gesplitst in twee delen;

- Deel A: dit deel beschrijft het uitvoeren van een (beperkt) aantal testen op een zogenaamde 'virtual machine'. Deze 'virtual machine' bevat een werkende BRP-omgeving waarin de beschreven stappen van Deel B al zijn uitgevoerd. Met behulp van deze omgeving kan op relatief eenvoudige wijze aangetoond worden dat het systeem werkt.
- Deel B: dit deel beschrijft de stappen die noodzakelijk zijn om van de broncode tot een werkend systeem te komen waarbij de broncode gebouwd, uitgerold en getest wordt.

Deel A

1 Inleiding

Met behulp van de volgende stappen worden een aantal basale tests uitgevoerd op een vooraf geïnstalleerde BRP-omgeving. Deze omgeving is op basis van de opleverde broncode opgebouwd in een virtuele omgeving volgens de werkwijze zoals beschreven in Deel B. Deze virtuele omgeving is in de oplevering meegeleverd als VirtualBox image. Aanvullend bevat de omgeving een gevulde test-database met ongeveer 34.000 personen en hun relaties.

1.1 Benodigde kennis

Voor deze installatiehandleiding wordt uitgegaan dat de lezer beschikt over de benodigde voorkennis waaronder:

- (Het installeren van) VirtualBox en het kunnen configureren hiervan.
- (Beperkte) kennis rond het BRP-berichtenmodel.

1.2 Technische vereisten

Om de BRP-omgeving te kunnen draaien is een computer nodig die voldoet aan de volgende eisen:

- Vrije schijfruimte: minimaal 32 GB
- Vrij werkgeheugen: minimaal 8 GB beschikbaar voor de BRP-omgeving
- De omgeving is geconfigureerd om gebruik te maken van 4 CPU-cores. Dit is geen vereiste maar wel wenselijk.

2 Gereed maken omgeving

2.1 Installatie VirtualBox

Download en installeer Oracle VirtualBox van <https://www.virtualbox.org/>
De versie die gebruikt is voor de installatie is 5.1(.26).

2.2 Toevoegen machine

Start VirtualBox. Kopieer alle bestanden en mappen uit de 02 Software\04 VM map naar een lokale map. Start VirtualBox en kies de menu optie Toevoegen uit het menu Machine. Selecteer de zojuist gekopieerde BRP.vbox.

2.3 Starten omgeving

Selecteer de BRP machine in VirtualBox kies Starten.

Bij het starten van de omgeving worden ook de BRP (docker) services gestart. Aangezien een groot aantal services op één omgeving draaien kost het enige tijd om alles op te starten. Wacht daarom een aantal minuten voor het verder gaan met het starten van de tests. Als de test gestart worden voordat de omgeving 'idle' wordt, kunnen onderdelen mogelijk nog niet volledig gestart zijn en fouten geven bij het uitvoeren van de tests.

3 Algemene informatie over de omgeving

De omgeving draait CentOS 7 met een Gnome desktop.

Er zijn twee gebruikersaccounts voor de omgeving:

- Gebruiker 'brp' met wachtwoord 'brp'. Dit is de reguliere gebruiker waar de programmatuur onder is geïnstalleerd.
- Gebruiker 'root' met wachtwoord 'root'.

Er zijn twee poorten doorverwezen op de host-machine:

- De poort 2222 is verbonden met de SSH poort (22) op de virtuele omgeving. Hiermee kan je 'remote' inloggen op de terminal
- De poort 2254 is verbonden met de BRP-PostgreSQL database die draait op de virtuele omgeving.

Er zijn geen aanvullende maatregelen getroffen om de omgeving te beveiligen. Zorg dus dat de omgeving afgeschermd is van het internet.

4 Tests uitvoeren

Er zijn een aantal tests opgezet die uitgevoerd worden met SoapUI. Alle tests zijn op het BRP koppelvlak. Hiervoor is gekozen omdat dit het meest interactieve en visuele resultaat geeft.

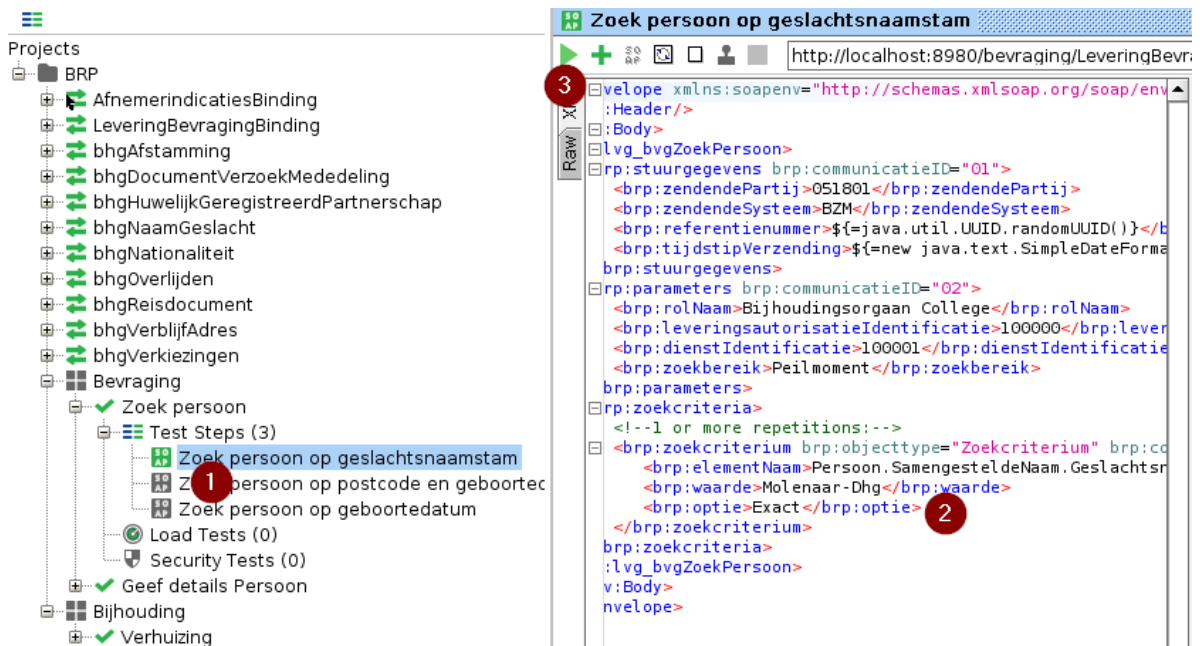
4.1 Starten SoapUI.

Log in op de omgeving met de BRP gebruiker.
Start SoapUI middels het SoapUI icoon op het bureaublad.

Het BRP project opent in SoapUI. In dit project zijn 3 testen gedefinieerd. Twee bevraging-services en een bijhouding-service.

4.2 Uitvoeren bevraging Zoek persoon

1. Open de 'boom' op de test 'Zoek person op geslachtsnaamstam'.
2. Dit testgeval bevraagt de service ZoekPersoon met het zoekcriterium 'Geslachtsnaamstam moet gelijk zijn aan Molenaar-Dhg'.
3. Voer de bevraging uit met de 'afspeel-knop'.



De webservice wordt bevraged en geeft het resultaat terug van de zoekvraag in het resultaatsscherm.

Er zijn twee soortgelijke zoekvragen; op postcode/geboortedatum combinatie en uitsluitend op geboortedatum. Het laatste testgeval zal als antwoord geven dat de zoekvraag niet specifiek genoeg is omdat deze testdatabase veel personen bevat die geboren zijn in 'januari 2017'.

4.3 Uitvoeren bevraging Geef details persoon

Het testgeval Geef details persoon vraagt (1) het volledige beeld op van de person met het bij (2) opgegeven burgerservicenummer. Dit is het burgerservicenummer van de in de eerste test gevonden persoon.

Test suite: bhgDocumentVerzoekMededeling

- bhgHuwelijkGeregistreerdPartnerschap
- bhgNaamGeslacht
- bhgNationaliteit
- bhgOverlijden
- bhgReisdocument
- bhgVerblijfAdres
- bhgVerkiezingen
- Bevraging
- ✓ Zoek persoon
 - Test Steps (3)
 - Zoek persoon op geslachtsnaamstam
 - Zoek persoon op postcode en geboortedatum
 - Zoek persoon op geboortedatum
 - Load Tests (0)
 - Security Tests (0)
- ✓ Geef details Persoon
 - Test Steps (1)
 - Geef details persoon op BSN **1**
 - Load Tests (0)

XML Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <brp:lvb_vbgGeefDetailsPersoon>
      <brp:stuurgegevens brp:communicatieID="01">
        <brp:zendendePartij>051801</brp:zendendePartij>
        <brp:zendendeSysteem>BZM</brp:zendendeSysteem>
        <brp:referentienummer>${=java.util.UUID.randomUUID()}</brp:referentienummer>
        <brp:tijdstipVerzending>${=new java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss")}</brp:tijdstipVerzending>
      </brp:stuurgegevens>
      <brp:parameters brp:communicatieID="02">
        <brp:rolNaam>Bijhoudingsorgaan College</brp:rolNaam>
        <brp:leveringsautorisatieIdentificatie>100000</brp:leveringsautorisatieIdentificatie>
        <brp:dienstIdentificatie>100004</brp:dienstIdentificatie>
      </brp:parameters>
      <brp:identificatiecriteria brp:communicatieID="03">
        <brp:burgerservicenummer>010000739</brp:burgerservicenummer>
      </brp:identificatiecriteria>
    </brp:lvb_vbgGeefDetailsPersoon>
  </soapenv:Body>
</soapenv:Envelope>
```

4.4 Uitvoeren bijhouding verhuizing

De in de vorige stappen gevonden persoon wordt verhuisd middels een binnengemeentelijke verhuizing. Deze verhuizing bestaat uit een aantal stappen:

1. Zoek de persoon om de 'objectsleutel' te achterhalen. Voer deze test uit.
2. Openen van de verhuizing-bijhouding.

Test suite: Bijhouding

- originaaliteit
- bhgOverlijden
- bhgReisdocument
- bhgVerblijfAdres
- bhgVerkiezingen
- Bevraging
- ✓ Zoek persoon
 - Test Steps (3)
 - Zoek persoon op geslachtsnaamstam
 - Zoek persoon op postcode en geboortedatum
 - Zoek persoon op geboortedatum
 - Load Tests (0)
 - Security Tests (0)
- ✓ Geef details Persoon
 - Test Steps (1)
 - Geef details persoon op BSN **5**
 - Load Tests (0)
 - Security Tests (0)
- Bijhouding
 - ✓ Verhuizing
 - Test Steps (2)
 - Zoek persoon op BSN **1**
 - Verhuizing **2**
 - Load Tests (0)

XML Response:

```
<!-- You have a CHOICE of the next 3 items at this level -->
<brp:verhuizingBinnengemeentelijk brp:objecttype="AdministratieveHandeling">
  <brp:partijCode>051801</brp:partijCode>
  <brp:acties>
    <brp:registratieAdres brp:objecttype="Actie" brp:communicatieID="01">
      <brp:datumAanvangGeldigheid>${=new java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss")}</brp:datumAanvangGeldigheid>
      <brp:persoon brp:objecttype="Persoon" brp:objectSleutel="${=java.util.UUID.randomUUID()}</brp:persoon>
      <brp:adressen>
        <brp:adres brp:objecttype="PersoonAdres" brp:communicatieID="02">
          <brp:soortCode>W</brp:soortCode>
          <brp:redenWijzigingCode>P</brp:redenWijzigingCode>
          <brp:aangeverAdreshoudingCode>I</brp:aangeverAdreshoudingCode>
          <brp:datumAanvangAdreshouding>${=new java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss")}</brp:datumAanvangAdreshouding>
          <brp:identificatiecodeAdresseerbaarObject>051801004</brp:identificatiecodeAdresseerbaarObject>
          <brp:identificatiecodeNummeraanduiding>051820004100</brp:identificatiecodeNummeraanduiding>
          <brp:naamOpenbareRuimte>Scharndam</brp:naamOpenbareRuimte>
          <brp:afgekorteNaamOpenbareRuimte>Scharndam</brp:afgekorteNaamOpenbareRuimte>
          <brp:huisnummer>7</brp:huisnummer>
          <brp:huisnummertoevoeging>${=new java.text.DecimalFormat("#.00")}</brp:huisnummertoevoeging>
          <brp:postcode>2566NA</brp:postcode>
          <brp:woonplaatsnaam>s-Gravenhage</brp:woonplaatsnaam>
        </brp:adres>
      </brp:adressen>
    </brp:registratieAdres>
  </brp:acties>
```

3. In de verhuizing wordt de persoon aangewezen met de in stap 1 gevonden objectsleutel. Om de verhuizing meerdere malen te kunnen uitvoeren, wordt automatisch de ingangsdatum (datumAanvangGeldigheid en datumAanvanAdreshouding) en de huisnummertoevoeging van het adres verhoogd. De rest van de adresgegevens kunnen gelijk blijven.

4. Voer de bijhouding uit.¹

5. Controleer met een nieuwe 'Geef details persoon op BSN' of de adreswijziging is doorgevoerd.

Na het muteren van de persoon past de objectsleutel aan. Om de persoon nogmaals te verhuizen moet de nieuwe objectsleutel met stap 1 opgehaald worden voor het uitvoeren van de nieuwe bijhouding.

¹ Geeft de bijhouding als resultaat ALG0001 (algemene fout), dan is het systeem nog niet volledig opgestart. Voer de bijhouding dan nogmaals uit na een aantal minuten.

Deel B

5 Inleiding

Operatie BRP bestaat uit verschillende systemen (Initiële Vulling, ISC en BRP) en verschillende componenten. Om na het 'ordentelijk stoppen' de source code bruikbaar op te leveren is deze handleiding geschreven om, zonder details, te beschrijven hoe het systeem, op basis van de source code, gebouwd, gestart en getest kan worden.

Deze handleiding bestaat uit drie hoofddelen:

1. Het bouwen van de software; hoofdstuk 7
2. Het testen van de software; hoofdstukken 8 en 9)
3. Het meten van de code kwaliteit; hoofdstuk 10

5.1 Benodigde kennis

Voor deze installatiehandleiding wordt uitgegaan dat de lezer beschikt over alle benodigde voorkennis waaronder, maar niet beperkt tot:

- Algemene inhoudelijk kennis van de te installeren componenten en hun functionaliteiten,
- de configuratie van, het werken met en algemene kennis over:
 - het operating systeem,
 - de programmeertaal (Java),
 - de build tools (Maven en NPM),
 - de virtualisatie software (Docker)
 - de code kwaliteit software (SonarQube)

Let op: al worden bij de (meeste) stappen commando's gegeven om de stap uit te voeren, dienen deze commando's enkel als verduidelijking en als voorbeeld bij de gegeven stap. Deze voorbeelden kunnen niet de benodigde voorkennis vervangen of onnodig maken.

5.2 Versies

De volgende softwareversies worden vereist (tussen haakjes de huidige gebruikte versies bij het opstellen van dit document):

- Java 1.8_x (Oracle JDK 1.8_121)
- Maven 3.5.x (3.5.0)
- NodeJS v6.x.x (6.11.1)
- Docker 1.12.x (17.06.0-ce)

5.2.1 Overige systeemsoftware

De volgende standaard programmatuur is vereist op de omgeving:

- unzip
- bzip2
- tar

6 Tooling

Om te controleren dat de verschillende tools correct zijn geïnstalleerd dienen de volgende commando's gebruikt te worden:

6.1 Java

```
java -version
```

Uitvoer:

```
java version "1.8.0_92"
Java(TM) SE Runtime Environment (build 1.8.0_92-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.92-b14, mixed mode)
```

6.2 Maven

```
mvn -v
```

Uitvoer:

```
Apache Maven 3.5.0 (ff8f5e7444045639af65f6095c62210b5713f426; 2017-04-03T21:39:06+02:00)
Maven home: /opt/apache-maven-3.5.0
Java version: 1.8.0_144, vendor: Oracle Corporation
Java home: /usr/java/jdk1.8.0_144/jre
Default locale: nl_NL, platform encoding: UTF-8
OS name: "linux", version: "3.10.0-514.26.2.el7.x86_64", arch: "amd64", family: "unix"
```

6.2.1 Configuratie maven

Om toegang te krijgen tot sommige externe bibliotheken dienen de volgende twee repositories toegevoegd te worden aan de configuratie van Maven (~/.m2/settings.xml). De lichtgrijze tekst is een voorbeeld om aan te geven waar deze repositories toegevoegd kunnen worden.

```
<settings>
  <profiles>
    <profile>
      <id>bro</id>
      <activation>
        <activeByDefault>true</activeByDefault>
      </activation>
      <repositories>
        <repository>
          <id>repository.jboss.org-public</id>
          <name>JBoss.org Maven repository</name>
          <url>https://repository.jboss.org/nexus/content/groups/public</url>
        </repository>
        <repository>
          <id>jitpack.io</id>
          <name>GitHub ghostdriver Maven repository</name>
          <url>https://jitpack.io</url>
        </repository>
      </repositories>
    </profile>
  </profiles>
</settings>
```

6.3 **NPM (NodeJS)**

```
node -v
```

Uitvoer:

```
v6.11.2
```

```
npm -v
```

Uitvoer:

```
3.10.10
```

6.4 **Docker**

```
docker version
```

Uitvoer:

Client:

```
Version:      1.12.1
API version:  1.24
Go version:   go1.6.3
Git commit:   23cf638
Built:
OS/Arch:     linux/amd64
```

Server:

```
Version:      1.12.1
API version:  1.24
Go version:   go1.6.3
Git commit:   23cf638
Built:
OS/Arch:     linux/amd64
```


7 Applicatie samenstellen

De Operatie BRP software is afhankelijk van enerzijds publiek beschikbare (open source) bibliotheken en anderzijds van interne bibliotheken. Enkele van deze bibliotheken dienen gemaakt te worden voordat de Operatie BRP software gemaakt kan worden. Hierbij dient speciale aandacht gegeven te worden aan de database en koppelvlak componenten die wel binnen de Operatie, maar buiten de directe Operatie BRP software vallen. Van deze componenten dient de correcte versie te worden gemaakt. Deze versies zijn op het moment van schrijven:

- Database (BMR) versie 58.0.4
- Koppelvlak (XSD) versie 25.0.3

Nota: bij de commando's staan ten overvloede de GIT (versiebeheer) commando's. Deze commando's zijn niet nodig om de applicatie te maken, maar dienen om aan te geven dat van sommige componenten een specifieke versie gebouwd moet worden.

7.1 Uitgangssituatie

De beschrijving gaat uit van een installatie op een Linux systeem. De software draait ook probleemloos op een Windows omgeving, alleen vereist dit (triviale) aanpassingen aan de uit te voeren commando's en zullen een aantal tests falen omdat de huidige toestand van de broncode Linux geregeleindes verwacht.

Om de software te bouwen is minimaal 8 GB vrij werkgeheugen noodzakelijk en 32 GB aan vrije schrijfruimte. Om alle tests volledig uit te kunnen voeren is een omgeving met 24 GB aan vrij werkgeheugen nodig.

De benodigde broncode-bestanden voor de database definitie, koppelvlak definitie en Operatie BRP staan in de oplevering in 02 Software\01 Broncode\, in de vervolgtekst aangeduid met <release>.

Aanmaken hoofddirectory voor broncode

```
mkdir ~/brp
```

BRP netwerk

Zorg dat Docker een BRP netwerk kent, b.v.

```
docker network create --subnet=10.13.1.0/24 brp
```

7.2 Database definitie

Het database project bevat de definitie van het gegevens model.

```
cd ~/brp
mkdir database-code
cd database-code
unzip <release>/database-code-58.0.4.zip
cd brp-database
mvn clean install
```

7.3 Koppelvlak definitie

Het koppelvlak project bevat de definitie van de externe koppelvlakken die beschikbaar worden gesteld.

```
cd ~/brp
mkdir koppelvlak-code
cd koppelvlak-code
unzip <release>/koppelvlak-code-25.0.3.zip
cd brp-koppelvlak
mvn clean install
```

7.4 Externe bibliotheken

Sommige externe afhankelijkheden dienen gedownload te worden en opgeslagen te worden in de lokale Maven repository.

Download

<http://downloads.jboss.org/hornetq/hornetq-2.2.14.Final.zip>

Installatie

```
mvn install:install-file -Dfile="hornetq-2.2.14.Final.zip" -DgroupId=org.jboss -
DartifactId=hornetq -Dpackaging=zip -Dversion=2.2.14.Final
```

7.5 Operatie BRP

De Operatie BRP software dient in verschillende stappen uitgevoerd te worden.

```
cd ~/brp
mkdir operatiebrp-code
cd operatiebrp-code
unzip <release>/operatiebrp-code-145.3.zip
```

Tijdens het bouwen van de software wordt gebruikt gemaakt van diverse externe afhankelijkheden. Binnen het project werden deze afhankelijkheden opgehaald uit Nexus binnen de Modernodam omgeving. Binnen deze Nexus omgeving werd ook het versiebeheer van deze afhankelijkheden afgehandeld. Om de Docker images om de Docker images toch te kunnen bouwen zonder deze voorzieningen dienen de volgende wijzigingen te worden gedaan. Let wel; dit kan tot gevolg hebben dat er een nieuwere versie van de afhankelijkheid gebruikt wordt, dan waarmee getest is.

Alpine

Vervang in het bestand `distributed/docker-basis-alpine/src/main/docker/Dockerfile` de tekst:

```
# Upgrade alpine
RUN set -x \
    && echo "https://192.168.207.40/nexus3/repository/alpine-proxy/v3.5/main" > /etc/apk/repositories \
    && echo "https://192.168.207.40/nexus3/repository/alpine-proxy/v3.5/community" >> \
    /etc/apk/repositories \
    && echo "https://192.168.207.40/nexus3/repository/alpine-addon" >> /etc/apk/repositories \
    && apk upgrade --update --allow-untrusted \
    && rm -rf /var/cache/apk/*
```

Door:

```
# Upgrade alpine
RUN set -x \
    && apk upgrade --update \
    && rm -rf /var/cache/apk/*
```

Java

Vervang in het bestand `distributie/docker-basis-java/src/main/docker/Dockerfile` de tekst:

```
# Install GNU libc (aka glibc)
RUN ALPINE_GLIBC_PACKAGE_VERSION="2.23-r1" && \
    ALPINE_GLIBC_BASE_PACKAGE="glibc" && \
    ALPINE_GLIBC_BIN_PACKAGE="glibc-bin" && \
    ALPINE_GLIBC_I18N_PACKAGE="glibc-i18n" && \
    apk add --update --no-cache --allow-untrusted \
        "$ALPINE_GLIBC_BASE_PACKAGE" \
        "$ALPINE_GLIBC_BIN_PACKAGE" \
        "$ALPINE_GLIBC_I18N_PACKAGE" && \
    \
    /usr/glibc-compat/bin/localedef --force --inputfile POSIX --charmap UTF-8 C.UTF-8 || true && \
    echo "export LANG=C.UTF-8" > /etc/profile.d/locale.sh && \
    \
    apk del glibc-i18n
```

Door:

```
# Install GNU libc (aka glibc)
RUN ALPINE_GLIBC_BASE_URL="https://github.com/sgerrand/alpine-pkg-glibc/releases/download" && \
    ALPINE_GLIBC_PACKAGE_VERSION="2.25-r0" && \
    ALPINE_GLIBC_BASE_PACKAGE_FILENAME="glibc-$ALPINE_GLIBC_PACKAGE_VERSION.apk" && \
    ALPINE_GLIBC_BIN_PACKAGE_FILENAME="glibc-bin-$ALPINE_GLIBC_PACKAGE_VERSION.apk" && \
    ALPINE_GLIBC_I18N_PACKAGE_FILENAME="glibc-i18n-$ALPINE_GLIBC_PACKAGE_VERSION.apk" && \
    apk add --no-cache --virtual=.build-dependencies wget ca-certificates && \
    wget \
        "https://raw.githubusercontent.com/andyshinn/alpine-pkg-glibc/master/sgerrand.rsa.pub" \
    -O "/etc/apk/keys/sgerrand.rsa.pub" && \
    wget \
        "$ALPINE_GLIBC_BASE_URL/$ALPINE_GLIBC_PACKAGE_VERSION/$ALPINE_GLIBC_BASE_PACKAGE_FILENAME" \
        "$ALPINE_GLIBC_BASE_URL/$ALPINE_GLIBC_PACKAGE_VERSION/$ALPINE_GLIBC_BIN_PACKAGE_FILENAME" \
        "$ALPINE_GLIBC_BASE_URL/$ALPINE_GLIBC_PACKAGE_VERSION/$ALPINE_GLIBC_I18N_PACKAGE_FILENAME" && \
    apk add --no-cache \
        "$ALPINE_GLIBC_BASE_PACKAGE_FILENAME" \
        "$ALPINE_GLIBC_BIN_PACKAGE_FILENAME" \
        "$ALPINE_GLIBC_I18N_PACKAGE_FILENAME" && \
    \
    rm "/etc/apk/keys/sgerrand.rsa.pub" && \
    /usr/glibc-compat/bin/localedef --force --inputfile POSIX --charmap UTF-8 C.UTF-8 || true && \
    echo "export LANG=C.UTF-8" > /etc/profile.d/locale.sh && \
    \
    apk del glibc-i18n && \
    \
    rm "/root/.wget-hsts" && \
    apk del .build-dependencies && \
    rm \
        "$ALPINE_GLIBC_BASE_PACKAGE_FILENAME" \
        "$ALPINE_GLIBC_BIN_PACKAGE_FILENAME" \
        "$ALPINE_GLIBC_I18N_PACKAGE_FILENAME"

ENV LANG=C.UTF-8
```

en

```
&& zurl=https://192.168.207.40/nexus3/repository/alpine-addon/$JAVA_PACKAGE-$JAVA_VERSION.tar.gz \
```

door:

```
&& curl=http://cdn.azul.com/zulu/bin/$JAVA_PACKAGE-$JAVA_VERSION.tar.gz \
```

PostgreSQL

Vervang in het bestand `distributed/docker-basis-postgres/src/main/docker/Dockerfile` de tekst:

```
&& wget --no-check-certificate -O /usr/local/bin/gosu
"https://192.168.207.40/nexus3/repository/alpine-addon/gosu-amd64-1.sh" \
```

door

```
&& wget --no-check-certificate -O /usr/local/bin/gosu
"https://github.com/tianon/gosu/releases/download/1.10/gosu-amd64" \
```

Tomcat

Vervang in het bestand `distributed/docker-basis-tomcat/src/main/docker/Dockerfile` de tekst:

```
&& wget --no-check-certificate https://192.168.207.40/nexus3/repository/alpine-addon/apache-tomcat-
$TOMCAT_VERSION.tar.gz \
```

door:

```
&& wget --no-check-certificate http://archive.apache.org/dist/tomcat/tomcat-
7/v$TOMCAT_VERSION/bin/apache-tomcat-$TOMCAT_VERSION.tar.gz \
```

ElasticSearch

Vervang in het bestand `distributed/docker-logging-elasticsearch/src/main/docker/Dockerfile` de tekst:

```
&& wget -nv --no-check-certificate -O /tmp/elasticsearch-$ELASTICSEARCH_VERSION.tar.gz
https://192.168.207.40/nexus3/repository/elastic-proxy/downloads/elasticsearch/elasticsearch-
$ELASTICSEARCH_VERSION.tar.gz \
```

door:

```
&& wget -nv --no-check-certificate -O /tmp/elasticsearch-$ELASTICSEARCH_VERSION.tar.gz
https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-$ELASTICSEARCH_VERSION.tar.gz \
```

Kibana

Vervang in het bestand `distributed/docker-logging-kibana/src/main/docker/Dockerfile` de tekst:

```
&& wget -nv --no-check-certificate -O /tmp/kibana-$KIBANA_VERSION.tar.gz
https://192.168.207.40/nexus3/repository/elastic-proxy/downloads/kibana/kibana-$KIBANA_VERSION-linux-
x86_64.tar.gz \
```

door:

```
&& wget -nv --no-check-certificate -O /tmp/kibana-$KIBANA_VERSION.tar.gz
https://artifacts.elastic.co/downloads/kibana/kibana-$KIBANA_VERSION-linux-x86_64.tar.gz \
```

Logstash

Vervang in het bestand `distributed/docker-logging-logstash/src/main/docker/Dockerfile` de tekst:

```
&& wget -nv --no-check-certificate -O /tmp/logstash-$LOGSTASH_VERSION.tar.gz
https://192.168.207.40/nexus3/repository/elastic-proxy/downloads/logstash/logstash-
$LOGSTASH_VERSION.tar.gz \
```

door

```
&& wget -nv --no-check-certificate -O /tmp/logstash-$LOGSTASH_VERSION.tar.gz
https://artifacts.elastic.co/downloads/logstash/logstash-$LOGSTASH_VERSION.tar.gz \
```

Beheer-selectie

Verwijder het bestand `brp/beheer-selectie/beheer-ng2/.npmrc`.

7.5.1 *Hergebruikte GBA-v code*

Uit de GBA-v is de code hergebruikt die de Teletex set definieert.

Deze code bevindt zich binnen de Operatie BRP software structuur, maar maakt geen deel uit van de 'normale' applicatie bouw en moet dus apart worden uitgevoerd.

```
cd ~/brp/operatiebrp-code/migratie/migr-extern/gbav/migr-util-gbav
mvn clean install
```

7.5.2 *Testdata levering*

Er is een afhankelijkheid tussen verschillende projecten op het gebied van test data. BRP Leveren is afhankelijk van de bijgehouden personen die resulteren uit de testgevallen van de bijhouding en de conversie.

Deze afhankelijkheid ziet er binnen dit document vreemd uit, maar dit werd tijdens de ontwikkeling ondervangen door het geautomatiseerde build proces. Binnen dit proces draaien eerst de bijhouding testen en op een later moment werden de resultaatdata uit de bijhouding testen geïntegreerd in de levering testen.

Het betreft de volgende drie artifacten welke allen persoonsgegevens bevatten in geserialiseerd JSON formaat, de "blob".

- `bijhouding-blob.jar`; Bevat de persoon output van alle bijhouding testgevallen
- `oranje-blob.jar`; Bevat de persoon output van alle conversie testgevallen
- `special-blob.jar`; Bevat de persoon output van speciale BRP testgevallen

Voor het gemak zijn de release versies meegeleverd in de oplevering in `02 Software\02 Test-blobs` en kunnen deze direct in de repository geïnstalleerd worden. Voer hiervoor de volgende commando's uit

```
mvn install:install-file -Dfile=blob-oranje-365.jar -
DgroupId=nl.bzk.brp.excel2blob -DartifactId=blob-oranje -Dversion=365 -
Dpackaging=jar -Dclassifier=blob-oranje
mvn install:install-file -Dfile=blob-specials-735.jar -
DgroupId=nl.bzk.brp.excel2blob -DartifactId=blob-specials -Dversion=735 -
Dpackaging=jar -Dclassifier=blob-specials
mvn install:install-file -Dfile=blob-bijhouding-135.jar -
DgroupId=nl.bzk.brp.excel2blob -DartifactId=blob-bijhouding-BIJHOUDING -
Dversion=135 -Dpackaging=jar
```

De versies komen hier overeen met laatst geselecteerde release versies waarvan bepaald is dat

er voor geleverd moet worden. Deze versienummers zijn te vinden in het bestand algemeen/alg/pom.xml in de properties blob.oranje.version (huidige versie 365), blob.specials.version (huidige versie 735) en blob.bijhouding.bijhouding.version (huidige versie 135).

Zie de instructies in de paragrafen 7.5.4 en 9.4 om de test-artifacten zelf te builden.

7.5.3 *Operatie BRP*

Na het uitvoeren van bovenstaande stappen kan de Operatie BRP software worden gebouwd².

```
cd ~/brp/operatiebrp-code
```

```
mvn clean install -Pdocker,deploy-frontend -T3
```

Om de code te bouwen zonder het uitvoeren van de unittests kan de optie -DskipTests toegevoegd worden. Dit versnelt het bouw-proces.

7.5.4 *Geconverteerde GBA gegevens (optioneel)*

Na het bouwen van de Operatie BRP software kunnen (eventueel) de eerste twee blobs 'zelf' gebouwd worden. De eerste set aan gegevens is gebaseerd op geconverteerde GBA gegevens.

```
cd ~/brp/operatiebrp-code/brp/test/excel-blob-conversie
```

```
mvn install -pl :excel-blob-conversie -am
```

```
mvn install -pl :excel-blob-conversie -DskipTests -Pconversie-oranje -
  Denv.BUILD_NUMBER=365
```

```
mvn install -pl :excel-blob-conversie -DskipTests -Pconversie-specials -
  Denv.BUILD_NUMBER=735
```

² Het bouw proces is afhankelijk van diverse externe repositories waar publieke software bibliotheken uit opgehaald worden. Als deze bibliotheken niet beschikbaar zijn op het moment van bouwen, dan zal de bouw poging falen. In de praktijk blijkt de beschikbaarheid voor de Javascript bibliotheken ten behoeve van de beheer projecten niet 100% te zijn. Mocht er een foutmelding optreden bij het ophalen van de bibliotheken, probeer het dan later nog eens. In de Operatie BRP-ontwikkelstraat was dit ondervangen door lokale mirror van de benodigde repositories.

8 Applicatie testen

De applicatie testen kunnen uitgevoerd worden met de hiervoor gebouwde Docker images. Voor het uitvoeren is het nodig om de variabele DOCKER_SERVER te zetten.

```
set DOCKER_SERVER=localhost
```

Bij alle scripts wordt uitgegaan dat deze uitgevoerd worden vanaf het pad:

```
cd ~/brp/operatiebrp-code
```

Voor het uitvoeren van iedere test moet een 'remove' script gedraaid worden om alle (draaiende) Docker containers te verwijderen:

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-remove.pom.xml -Pdocker-remove
```

8.1 Integratie testen initiële vulling

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=iv
```

```
mvn test -U -f migratie/migr-test-isc/regressie-iv.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

8.2 Integratie testen migratie voorzieningen

Let op; deze test vereist een omgeving met minimaal 24GB geheugen.

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=integratie
```

```
mvn test -U -f migratie/migr-test-isc/regressie-int.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

8.3 Integratie testen bijhouding

In de bijhoudingstests staat nog een verwijzing naar modernodam servers. Om de tests voor bijhouding te kunnen uitvoeren moeten de hosts test-bij-hg-dok.modernodam.nl, test-bij-as-dok.modernodam.nl en test-bij-va-dok.modernodam.nl resolvable zijn naar 127.0.0.1. Voeg hiervoor aan de /etc/hosts file de volgende regel toe:

```
127.0.0.1 test-bij-hg-dok.modernodam.nl test-bij-as-dok.modernodam.nl test-bij-va-dok.modernodam.nl
```

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=brp-bijhouding
```

Voor het vullen van de database met testdata;

```
cd ~/brp/operatiebrp-code/brp/test/deployment-art-omgeving/
```

en voer uit:

```
mvn verify -P nieuwste,create-test-db-docker-bijhouding -Ddbusername=brp -Ddbpassword=brp -Ddbprefix=
```

Om de testcases te draaien:

```
cd ~/brp/operatiebrp-code/brp/test/testcases/nieuwestijl
```

Huwelijk en geregistreerdpartnerschap:

```
mvn install -Dit.test=Koppelvlaktest_docker
```

Afstamming:

```
mvn install -Dit.test=Koppelvlak_AS_docker
```

Verblijf en Adres:

```
mvn install -Dit.test=Koppelvlak_VA_docker
```

8.4

Integratie testen levering

Navigeer naar de directory:

```
cd ~/brp/operatiebrp-code/brp/test/brp-docker-test
```

Voer het volgende commando uit waarbij <DOCKER_IP> vervangen is door het IP van de Docker daemon (zie output ip a, ipconfig of ifconfig). Onderstaande test werkt alleen als de firewall de verbinding van het interne docker naar de exposed ports niet blokkeert. Stop eventueel de firewall en start docker met de optie `--iptables=false`.

```
mvn verify -Dit.test=EndToEndSuite -DDOCKER_IP=<DOCKER_IP>
```

Resultaten zijn na de run te vinden in `brp/test/brp-docker-test/target/jbehave`

9 Bijlage A – Overige testen

Naast de uitgevoerde basis set aan integratie testen zijn er nog andere component en integratie testen.

9.1 Regressietesten Migratievoorzieningen

9.1.1 *Regressie component GBA Afnemerindicaties in BRP*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-afnemerindicatie-gba
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-afnemerindicatie-gba.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.2 *Regressie component GBA Archivering in BRP*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-archivering-gba
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-archivering-gba.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.3 *Regressie component BRP Beheer*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-beheer
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-beheer.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.4 *Regressie component GBA Bevraging in BRP*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-bevraging-gba
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-bevraging-gba.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.5 *Regressie component GBA Bijhouding in BRP*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-bijhouding-gba
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-bijhouding-gba.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.6 *Regressie component GBA expressies in BRP*

```
cd ~/brp/operatiebrp-code/brp/test/lo3conversie/conversie-expressies
mvn test -Pregressie -DfailIfNoTests=false
```

9.1.7 *Regressie component GBA Levering in BRP*

```
cd ~/brp/operatiebrp-code/brp/test/lo3conversie/conversie-levering
mvn test -Pregressie -DfailIfNoTests=false
```

9.1.8 *Regressie component GBA Selectie in BRP*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-selectie
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-selectie-gba.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.9 ***Regressie component GBA Sleutelrubrieken in BRP***

```
cd ~/brp/operatiebrp-code/brp/test/lo3conversie/conversie-sleutelrubrieken
mvn test -Pregressie -DfailIfNoTests=false
```

9.1.10 ***Regressie component GBA Vrijbericht in BRP***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=brp-vrijbericht-gba
```

```
mvn test -f migratie/migr-test-isc/regressie-brp-vrijbericht-gba.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.11 ***Regressie component ISC Beheer***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=isc-beheer
```

```
mvn test -f migratie/migr-test-isc/regressie-isc-beheer.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.12 ***Regressie component ISC***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=isc
```

```
mvn test -f migratie/migr-test-isc/regressie-isc.pom.xml -DDOCKER_SERVER=localhost
-Dregressietest.properties=docker.properties -Pregressie
```

9.1.13 ***Regressie component IV-Logging***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=iv-logging
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-logging.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.14 ***Regressie component IV-Naar BRP***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=iv-naarbrp
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-naarbrp.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.15 ***Regressie component IV-Naar LO3***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=iv-naarlo3
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-naarlo3.pom.xml -
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.16 ***Regressie component Synchronisatie (IV-modus)***

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-
start -Dprofiel=sync-init
```

```
mvn test -f migratie/migr-test-isc/regressie-sync-init.pom.xml -  
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.17 *Regressie component Synchronisatie (synchronisatie-modus)*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-  
start -Dprofiel=sync-sync
```

```
mvn test -f migratie/migr-test-isc/regressie-sync.pom.xml -  
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.1.18 *Regressie component VOISC*

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-  
start -Dprofiel=voisc
```

```
mvn test -f migratie/migr-test-isc/regressie-voisc.pom.xml -  
DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.2 Integratietesten Migratievoorzieningen

9.2.1 Integratie IV Personen (uc101)

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=iv
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-101.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.2.2 Integratie IV Autorisaties (uc102)

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=iv
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-102.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.2.3 Integratie IV Afemerindicaties (uc103)

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=iv
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-103.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.2.4 Integratie IV Protocollering (uc104)

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=iv
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-104.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.2.5 Integratie IV configuratie (uc105)

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=iv
```

```
mvn test -f migratie/migr-test-isc/regressie-iv-105.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.2.6 Integratie leveringsvergelijking

Let op; deze test vereist een omgeving met minimaal 24GB geheugen.

```
mvn -f deployment/deployer-docker/src/main/deployer/docker-start.pom.xml -Pdocker-start -Dprofiel=proefsynchonisatie
```

```
mvn test -f migratie/migr-test-isc/regressie-proefsync.pom.xml -DDOCKER_SERVER=localhost -Dregressietest.properties=docker.properties -Pregressie
```

9.3 Conversietesten Migratievoorzieningen

9.3.1 Regressie component Persoon NaarBrp

```
cd ~/brp/operatiebrp-code/migratie/migr-test-persoon-naarbrp
```

```
mvn test -Pregressie -Dspring.profiles.active=memoryDS -Dsynchonisatie.deltaAan=true -Dsynchonisatie.relaterenAan=false -DfailIfNoTests=false
```

Resultaten van de test staan in regressie-output/rapport.html.

9.3.2 Regressie component Persoon NaarLO3

```
cd ~/brp/operatiebrp-code/migratie/migr-test-persoon-naarlo3
```

```
mvn test -Pregressie -Dspring.profiles.active=memoryDS -DfailIfNoTests=false
```

Resultaten van de test staan in regressie-output/rapport.html.

9.3.3 Regressie component Persoon Preconditie

```
cd ~/brp/operatiebrp-code/migratie/migr-test-persoon-preconditie
```

```
mvn test -Pregressie -Dspring.profiles.active=memoryDS -DfailIfNoTests=false -
Dsynchonisatie.relaterenAan=false
```

Resultaten van de test staan in regressie-output/rapport.html.

9.4 Bijhouding

Voor het draaien van de standaard bijhoudings tests welke direct inspringen op de business laag Is wel een database nodig, hiervoor wordt in deze handleiding een Docker image gebruikt.

```
docker run -d --name brp-database -p 5432:5432 -e POSTGRES_DATABASE=art-brp -e
NAME=art-brp brp/database:latest
```

```
mvn verify -f brp/test/deployment-art-omgeving/pom.xml -Pnieuwste,create-test-db-
docker-bijhouding -Ddbusername=brp -Ddbpassword=brp -Ddbprefix=art-
```

Voor het uitvoeren van de testen kan nu de standard configuratie worden gebruikt die staat geconfigureerd:

```
cd ~/brp/operatiebrp-code/brp/test/testcases/nieuwestijl
```

```
mvn verify -Dit.test=BijhoudingAllesTest
```

Voor het genereren van blobs ten behoeve van levering:

```
mvn install -pl :brp-art-testcases-nieuwestijl -Pblobjar -
Dit.test=BijhoingBLOBTest -Denv.BUILD_NUMBER=365
```

9.5 Levering

```
cd ~/brp/operatiebrp-code/brp/test/brp-api-test
```

Voer het volgende commando uit:

```
mvn verify -Dit.test=IT_UsecaseSuite
```

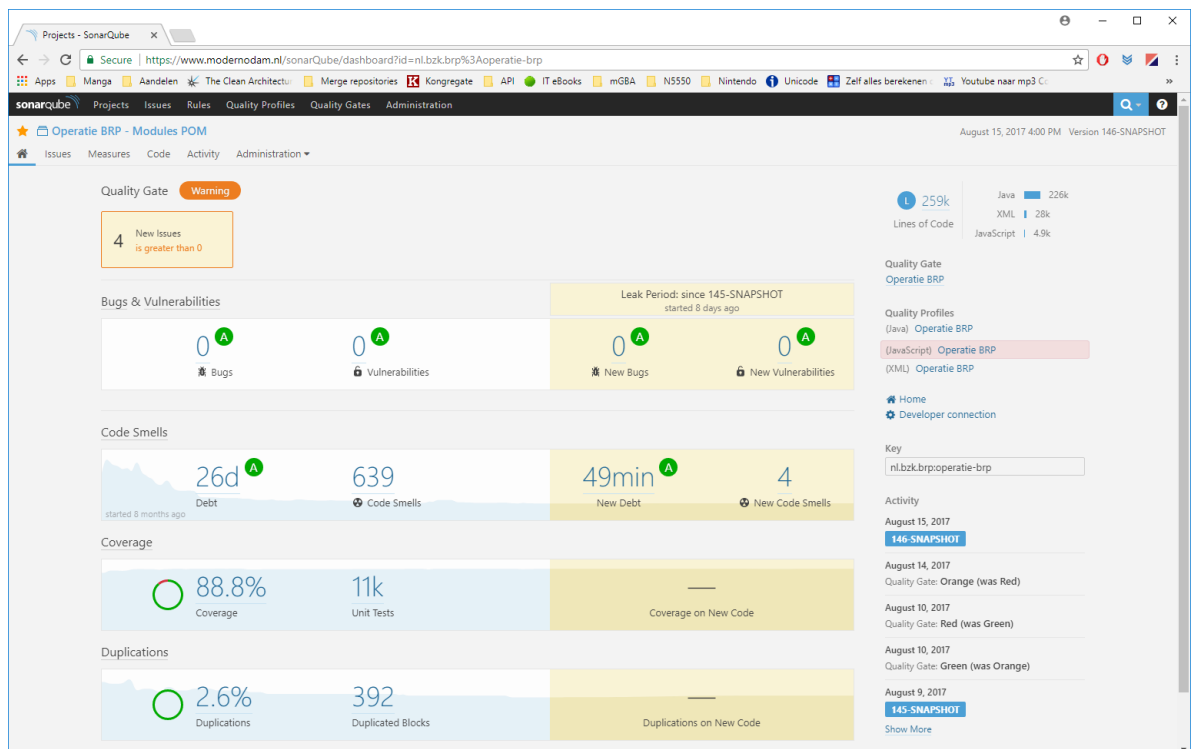
Resultaten zijn na de run te vinden in:

```
~/brp/operatiebrp-code/brp/test/brp-api-test/target/jbehave
```

10 Bijlage B – SonarQube

SonarQube wordt gebruikt om de kwaliteitseisen aan de source code te controleren. De Operatie BRP specifieke bestanden zijn opgenomen in de oplevering in de 02 Software\03 SonarQube. Om deze controle uit te kunnen voeren dienen de volgende stappen uitgevoerd te worden die in de volgende paragrafen nader worden toegelicht:

- Configuratie SonarQube omgeving
 - Configuratie kwaliteitsprofielen (Quality Profiles)
 - Configuratie kwaliteitseisen (Quality Gate)
- Uitvoeren kwaliteitcontrole
 - Inlezen 'explain'



10.1 Configuratie SonarQube omgeving

Er wordt gebruik gemaakt van SonarQube versie 6.3 met de volgende plugins:

- Issue resolver (versie 1.0.2)
- Rules Compliance Index (versie 1.0.1)
- SonarJS (versie 3.0; build 4962)
- SonarJava (versie 4.9.0.9858)
- SonarXML (versie 1.4.2; build 885)
- jDepend (versie 1.1.1)

Nota: indien niet exact dezelfde versie wordt gebruikt, kunnen de (implementaties van de) kwaliteitsregels anders zijn en kunnen afwijkende resultaten worden verkregen.

Installeer SonarQube met de plugins.

10.1.1 Quality Profiles

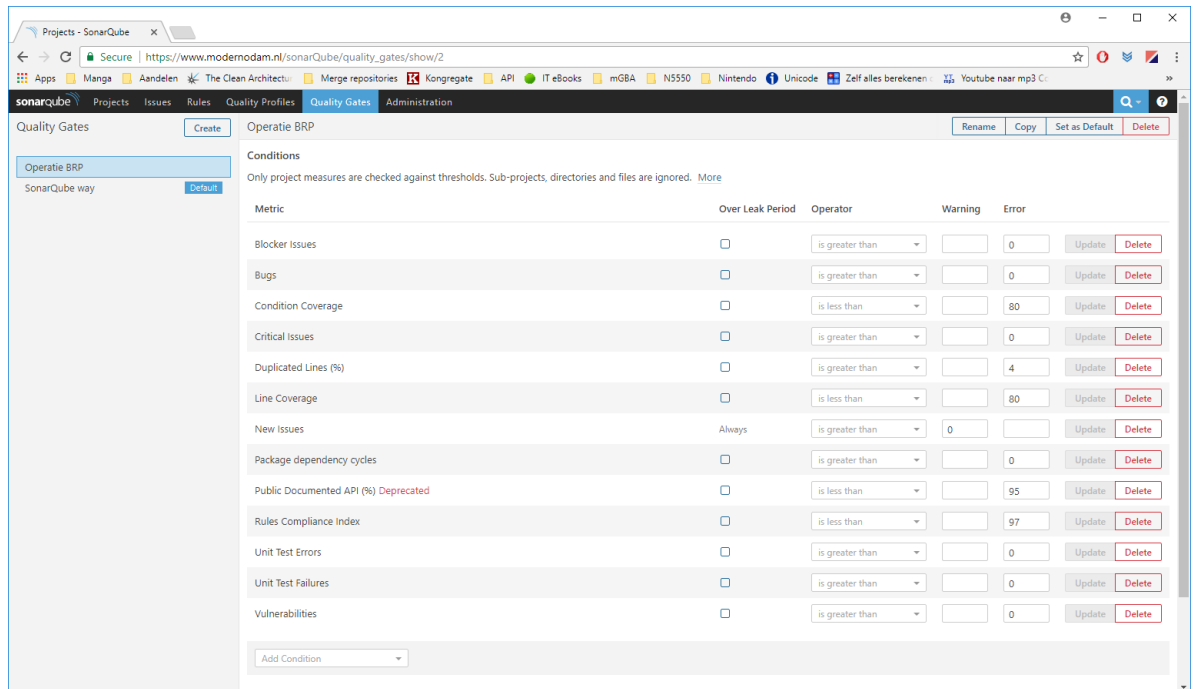
De volgende backups van Quality Profiles dienen te worden ingelezen:

- Java (java-operatie-brp-47749.xml)
- JavaScript (js-operatie-brp-32685.xml)
- XML (xml-operatie-brp-22558.xml)

Nota: om de correcte resultaten te krijgen dienen de kwaliteitsprofielen gekoppeld te zijn aan het project in SonarQube.

10.1.2 Quality Gate

De volgende Quality Gate dient te worden geconfigureerd:



Nota: om de correcte resultaten te krijgen dienen de kwaliteitsprofielen gekoppeld te zijn aan het project in SonarQube.

10.1.3 Uitvoeren kwaliteitscontrole

Om het volledige beeld van de kwaliteit van de source code te krijgen dient eerst de unittests te worden uitgevoerd. (let op: er dient eerst een volledig build te zijn uitgevoerd, zie paragraaf 7.5.3):

```
cd ~/brp/operatiebrp-code
```

```
mvn -pl :jacoco-clean clean:clean
```

```
mvn -Dmaven.test.failure.ignore verify -P!modules-brp,!modules-build,!modules-
deployment,!modules-distributie,!modules-migratie,coverage
mvn -Dmaven.test.failure.ignore verify -P!modules-algemeen,!modules-
build,!modules-deployment,!modules-distributie,!modules-migratie,coverage,inmemory
mvn -Dmaven.test.failure.ignore verify -P!modules-algemeen,!modules-brp,!modules-
build,!modules-distributie,!modules-migratie,coverage
mvn -Dmaven.test.failure.ignore verify -P!modules-algemeen,!modules-brp,!modules-
build,!modules-deployment,!modules-migratie,coverage
mvn -Dmaven.test.failure.ignore verify -P!modules-algemeen,!modules-brp,!modules-
build,!modules-deployment,!modules-distributie,coverage
```

Om de testresultaten van de unittesten aan te vullen met regressie testen dienen de volgende stappen te worden uitgevoerd:

```
mvn -Dmaven.test.failure.ignore verify -Pregressie-coverage,coverage,inmemory -pl
:migr-test-afnemersindicatie
mvn -Dmaven.test.failure.ignore verify -Pregressie,coverage,inmemory -pl :migr-
test-autorisatie
mvn -Dmaven.test.failure.ignore verify -Pregressie,coverage,inmemory -pl :migr-
test-persoon-database
mvn -Dmaven.test.failure.ignore verify -Pregressie-coverage,coverage,inmemory -pl
:migr-test-persoon-naarbrp
mvn -Dmaven.test.failure.ignore verify -Pregressie-coverage,coverage,inmemory -pl
:migr-test-persoon-preconditie
```

Om de resultaten van de testen samen te voegen zodat deze ingelezen kunnen worden door SonarQube dient de volgende stap te worden uitgevoerd:

```
mvn -f build/jacoco/merge.pom.xml -Pcoverage jacoco:merge
```

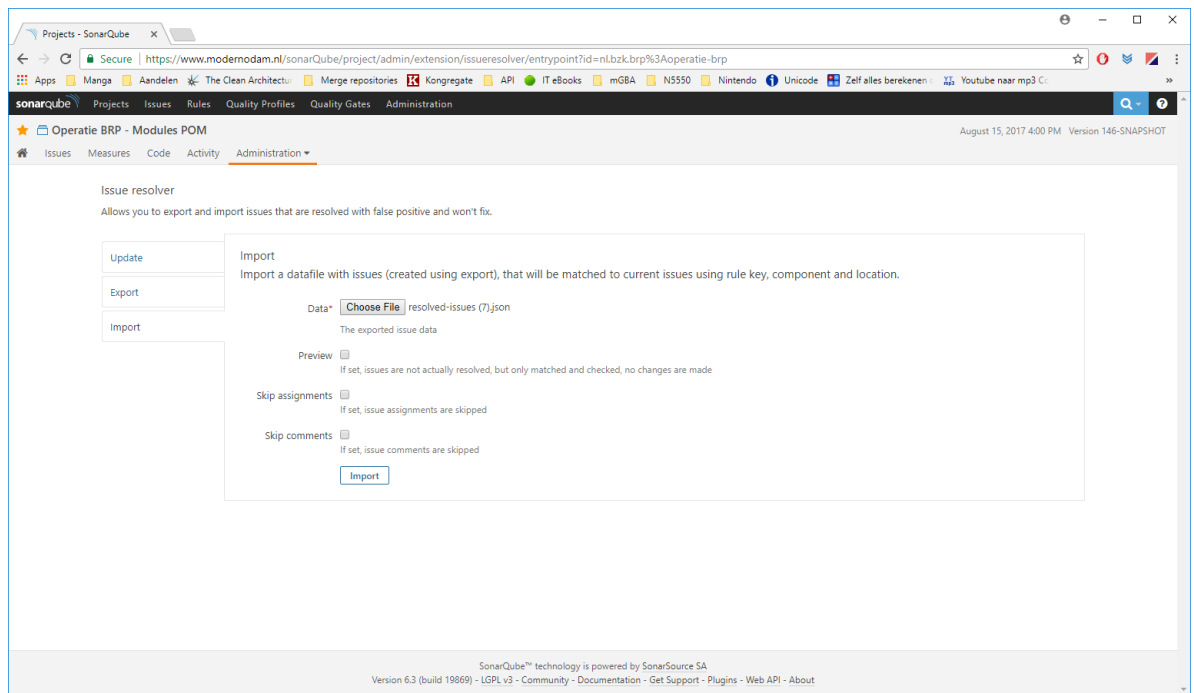
Om tenslotte source code te controleren met SonarQube dient het volgende te worden uitgevoerd (*let op: gebruik de juiste configuratie om SonarQube te benaderen*):

```
mvn -P!modules-tooling sonar:sonar -Dsonar.host.url=http://localhost:9000/
```

10.1.4 Inlezen 'explain'

Sommige regels uit SonarQube worden met een reden overtreden. Om deze redenen te documenteren en vervolgens het juiste resultaat in SonarQube te tonen, dient (nadat het project een keer is gecontroleerd) deze data inlezen te worden (met de Issue resolver plugin):

- resolved-issues.json



Let op: door de volgorde van verwerking binnen SonarQube zal na het inlezen van de 'resolved issues' de Quality Gate niet zijn aangepast. Deze zal pas na een volgende controle worden aangepast.