



Ministerie van Economische Zaken,
Landbouw en Innovatie

CBSviewer Webapp

Project documentatie

User Guide

Inhoud

1. Inhoud	i
2. Home	1
3. Release notes	2
4. Technisch ontwerp	4
4..1. Layer model	5
5. Ontwikkel handleiding	7
6. Installatie handleiding	10

1 Home

1.1 CBS Viewer Site

Dit is de Maven Site voor CBS Viewer, een [Webrichtlijnen](#) compliant viewer.

2 Release notes

2.1 CBSviewer Webapp 1.2-SNAPSHOT Release Notes

Dit zijn de release notes van CBSviewer Webapp versie 1.2-SNAPSHOT.

- [CBSviewer Webapp 1.2-SNAPSHOT Release Notes](#)
 - [1.2-SNAPSHOT Release Notes](#)
 - [1.1 Release Notes](#)
 - [1.0 Release Notes](#)
 - [0.2 Release Notes](#)
 - [0.1 Release Notes](#)

2.1.1 1.2-SNAPSHOT Release Notes

Een complete lijst van de issues van deze release wordt gegeven in de tracker [milestone 1.2-SNAPSHOT](#)

2.1.1.1 Verbetering

- autocomplete voor adreszoeken
- uitbreiding documentatie

2.1.1.2 Nieuw

- nieuwe huisstijl CBS websites
- gebruik van Compass
- schaalbalk op de kaart
- overzichtskaartje

2.1.2 1.1 Release Notes

Een complete lijst van de issues van deze release wordt gegeven in de tracker [milestone 1.1](#)

2.1.2.1 Verbetering

- printen

2.1.2.2 Nieuw

- feature attribuutnaam mapper
- feature attribuutwaarde mapper
- transparantie voor de voorgrondkaart
- gebruik van Sass

2.1.3 1.0 Release Notes

Een complete lijst van de issues van deze release wordt gegeven in de tracker [milestone 1.0](#)

2.1.3.1 Verbetering

- bugfixes en workarounds voor pdok gazetteer en adreszoeken.

- verbetering in feature informatie parsing en presentatie

2.1.3.2 Nieuw

- wisselen achtergrond kaart

2.1.4 0.2 Release Notes

Milestone twee bevat functie uitbreidingen.

Een complete lijst van de issues van deze release wordt gegeven in de tracker [milestone 0.2](#)

2.1.4.1 Nieuw

- adres zoeken
- feature informatie
- lagen menu

2.1.5 0.1 Release Notes

Initiele oplevering van een basis viewer.

Een complete lijst van de issues van deze release wordt gegeven in de tracker [milestone 0.1](#)

3 Technisch ontwerp

3.1 Technisch Ontwerp

- [Technisch Ontwerp](#)

3.1.1 Open standaard interfaces

De applicatie gebruikt drie OGC standaard interfaces om te communiceren met data services. Deze interfaces zijn WMS (Web Map Service, versie 1.3.0 en 1.1.1) en WMTS (Web Map Tiled Service, versie 1.0) voor de kaart, legenda en attribuut informatie. Voor de adreszoek functie wordt gebruik gemaakt van OpenLS (Open Location Service, versie 1.0, Location Utility Service profiel)

3.1.2 Layer object

Binnen de applicatie is onderscheid tussen een achtergrond en thematische kaart. Voor de voorgrond kaart is een speciaal model gedefinieerd, dit is beschreven in [het layer model](#).

4 Layer model

4.1 Voorgrond kaartlaag model

- [Voorgrond kaartlaag model](#)
 - [Beschrijving](#)
 - [Attribuut namen filtering](#)
 - [Attribuut waarden filtering](#)

4.1.1 Beschrijving

De voorgrond kaart bestaat uit één of meer lagen (layer elementen) van een WMS service, dit maakt het mogelijk om voor alle lagen tegelijk een featureinfo verzoek om attribuut informatie uit te voeren en andere handelingen uit te voeren.

In de applicatie kan telkens één voorgrond laag gebruikt worden.

Feature-informatie wordt opgehaald als GML en geparsed op het serverside deel van de applicatie, op basis van de descriptor wordt een tabel gemaakt met de gevraagde feature attributen.

4.1.1.1 Voorbeeld lagen beschrijving

```
<?xml version="1.0" encoding="UTF-8" ?>
<Layers xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xml.util.cbsviewer.mineleni.nl" xsi:schemaLocation="http://xml.util.c
<Layerdescriptor>
<id>vierkanten500m_oad2000</id>
<name>Vierkant 500m - Omgevingsadressendichtheid 2000</name>
<layers>omgevings_adres_dichtheid_2000</layers>
<url>http://geodata.nationaalgeoregister.nl/cbsvierkanten500m/wms</url>
<styles>cbsvierkanten500m.oad2000</styles>
<attributes>oad2000</attributes>
<aliases>dichtheid</aliases>
</Layerdescriptor>
<!-- START SNIPPET: layermodel_apt_1 -->
<Layerdescriptor>
<id>wijkenbuurten2011_thema_gemeenten2011_aantal_inwoners</id>
<name>CBS Gemeenten 2011 - Aantal inwoners</name>
<layers>gemeenten2011</layers>
<url>http://geodata.nationaalgeoregister.nl/wijkenbuurten2011/wms</url>
<styles>wijkenbuurten2011_thema_gemeenten2011_aantal_inwoners</styles>
<attributes>aantal_inwoners, gemnaam</attributes>
<aliases>aantal_inwoners, gemeente</aliases>
<link>http://geodata.nationaalgeoregister.nl/wijkenbuurten2011/wfs?request=GetFeatu
</Layerdescriptor>
<!-- START SNIPPET: layermodel_apt_1 -->
</Layers>
```

De Layerdescriptor is gedefinieerd in [AvailableLayers.xsd](#)

4.1.2 Attribuuat namen filtering

Attribuutnamen kunnen per laag gefilterd worden, hiervoor is een mapping voorzien in het LayerDescriptor. element.

4.1.2.1 Voorbeeld

```
<Layerdescriptor>
<id>wijkenbuurten2011_thema_gemeenten2011_aantal_inwoners</id>
<name>CBS Gemeenten 2011 - Aantal inwoners</name>
<layers>gemeenten2011</layers>
<url>http://geodata.nationaalgeoregister.nl/wijkenbuurten2011/wms</url>
<styles>wijkenbuurten2011_thema_gemeenten2011_aantal_inwoners</styles>
<attributes>aantal_inwoners, gemnaam</attributes>
<aliases>aantal_inwoners, gemeente</aliases>
<link>http://geodata.nationaalgeoregister.nl/wijkenbuurten2011/wfs?request=GetFeatu
</Layerdescriptor>
</Layers>
```

4.1.3 Attribuuat waarden filtering

Attribuutwaarden in een feature info response kunnen globaal voor de applicatie gefilterd worden middels het attribuut waarden filter beschreven in [AttributeValuesFilter.xml](#). Deze functie is vooral nuttig als er ooit shapefiles zijn gebruikt omdat op deze wijze de null waarden en eventuele code waarden kunnen worden herschreven in de resultaat tabel.

4.1.3.1 Voorbeeld

```
<?xml version="1.0" encoding="UTF-8"?>
<FilterMapping xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xml.wms.cbsviewer.mineleni.nl" xsi:schemaLocation="http://xml.wms.cbs
<Filter>
<inputAttributeValue>-99997</inputAttributeValue>
<outputAttributeValue>geheim</outputAttributeValue>
</Filter>
<Filter>
<inputAttributeValue>-99998</inputAttributeValue>
<outputAttributeValue>nihil</outputAttributeValue>
</Filter>
<Filter>
<inputAttributeValue>-99999</inputAttributeValue>
<outputAttributeValue>onbekend</outputAttributeValue>
</Filter>
</FilterMapping>
```

Het attribuutwaarden filter is gedefinieerd in [AttributeValuesFilter.xsd](#)

5 Ontwikkel handleiding

5.1 Ontwikkel handleiding

- [Ontwikkel handleiding](#)
 - [Broncode ophalen](#)
 - [Beginnen met bouwen](#)
 - [code conventies en formatting](#)
 - [Unit testen](#)
 - [Java](#)
 - [Javascript](#)
 - [Applicatie testen](#)
 - [Documentatie genereren](#)
 - [Documentatie uploaden](#)
 - [Externe Informatie en links](#)

Dit project gebruikt <http://maven.apache.org/run-maven/index.html> als build tool en <https://github.com/MinELenI/CBSviewer> voor broncode beheer.

PROXY

Een aantal van de maven commando's heeft internet toegang nodig, het kan zijn dat je hiervoor http of https proxy configuratie gegevens moet aangeven op de commando regel.

```
-Dhttp.proxyHost=<host> -Dhttp.proxyPort=<poort> -Dhttps.proxyHost=<host> -Dhttps.proxyPort=<poort>
```

bijvoorbeeld: `mvn -Popenlayers -Dhttp.proxyHost=145.12.1.11
-Dhttp.proxyPort=8080 -Dhttps.proxyHost=145.12.1.11 -
Dhttps.proxyPort=8080 site`

5.1.1 Broncode ophalen

De broncode is opgeslagen in het Github systeem. De toegang is beschreven in het [Source Repository](#) document.

```
git clone https://github.com/MinELenI/CBSviewer.git C:\workspace\CBSviewer
```

5.1.2 Beginnen met bouwen

Het project gebruikt Maven 3 als build tool, als IDE raden we Eclipse 4.2 aan omdat deze de beste Maven integratie heeft. Het project is opgezet als een [http://en.wikipedia.org/wiki/WAR_file_format_\(Sun\)](http://en.wikipedia.org/wiki/WAR_file_format_(Sun)) artifact. Voor het maken van een .war of release is alleen een juist geconfigureerde Maven installatie nodig.

Om de ontwikkelomgeving op te zetten is het nodig om een build van OpenLayers te maken, dit gaat met onderstaand commando:

```
mvn -Popenlayers,developer
```

Hiermee wordt een debug versie van OpenLayers in de broncode ondergebracht, een compacte versie voor de webapp en een volledige versie voor de unit tests. De OpenLayers build configuratie is opgeslagen in het bestand `/OL-build.cfg`.

5.1.2.1 code conventies en formatting

Voor formatting van de Javascript code wordt het profiel gebruikt zoals dat te vinden is in de `ide/` directory.

Voor formatting van de Java code wordt het profiel gebruikt zoals dat te vinden is in de `ide/` directory.

5.1.3 Unit testen

5.1.3.1 Java

Er wordt gebruik gemaakt van JUnit 4 tests.

5.1.3.2 Javascript

Er wordt gebruik gemaakt van het <http://pivotal.github.com/jasmine/> framework. De code voor een unit test gaat in `/src/test/js/jasmine/`. Tests worden automatisch gedraaid in een HtmlUnit runtime tijdens de test fase van Maven. Om unit tests met de hand te draaien zijn tijdens het ontwikkelen kan het `jasmineserver` profiel worden gebruikt.

```
mvn -Pjasmineserver jasmine:bdd
```

Start de tests op: <http://localhost:8234/>, de pagina kan worden geladen in een browser naar keuze.

5.1.4 Applicatie testen

De applicatie kan lokaal gebouwd en gerund worden met behulp van Jetty

```
mvn -Pdeveloper jetty:run
```

Jetty start een webserver op op poort 8020 met de <http://localhost:8020/>.

5.1.5 Documentatie genereren

Met Maven kan een site worden gemaakt met het commando:

```
mvn site
```

Tijdens het maken van de site worden ook de unit en integratie tests gedraaid (er vindt dus een volledige build plaats!).

5.1.5.1 Documentatie uploaden

De documentatie moet "met de hand" naar de Github website worden overgedragen, hiervoor is authenticatie nodig.

```
mvn site-deploy -Dgithub.username=<login> -Dgithub.password=<passwd>
```

De website staat dan op: <http://mineleni.github.io/CBSviewer/>

5.1.6 Externe Informatie en links

- <http://help.github.com/>
- <http://help.github.com/send-pull-requests/>
-

6 Installatie handleiding

6.1 Installatie handleiding

- [Installatie handleiding](#)
 - [Tomcat](#)

6.1.1 Tomcat

De applicatie wordt uitgeleverd als .war bestand dat rechtstreeks in een Tomcat servlet container kan worden gebruikt.