

Eau de Web



**tripledev**  
www.tripledev.ee

# Deployment manual

---

## **Support services for the Digital Agenda Data Tool SMART 2015/1086**

Deployment manual detailing the complete process that would allow a third party to replicate the entire content and functionalities of the website in a new suitable environment

Last update: July 2016



## Table of contents

1.	Current installation .....	5
1.1.	Domain names .....	5
1.2.	Deployment diagram .....	5
1.3.	Operating system, system tools, libraries and prerequisites .....	6
1.3.1.	Libraries, tools, prerequisites .....	6
1.3.2.	Scheduled tasks .....	6
1.4.	Major software components .....	7
1.4.1.	Virtuoso Open-Source Edition .....	7
1.4.2.	Content Registry .....	7
1.4.3.	Plone .....	7
1.4.4.	ELDA .....	7
1.4.5.	Apache Web Server .....	8
1.5.	Source code .....	8
1.5.1.	Data files .....	8
1.5.2.	SSL certificates .....	9
1.5.3.	System services .....	10
2.	Installation from source code .....	11
2.1.	In a virtualized environment .....	11
2.2.	In a pre-configured server .....	11
2.3.	Testing installation .....	11
2.3.1.	Plone .....	11
2.4.	Log files .....	12
2.5.	Application-specific settings .....	13
2.5.1.	Plone .....	13
2.5.2.	Content Registry .....	13



## 1. Current installation

This document describes the hardware and software architecture currently used (as of July 2016) by the production and test instances of Digital Agenda Data Website.

It contains a detailed technical description of the software components, their configuration and installation steps on a clean system.

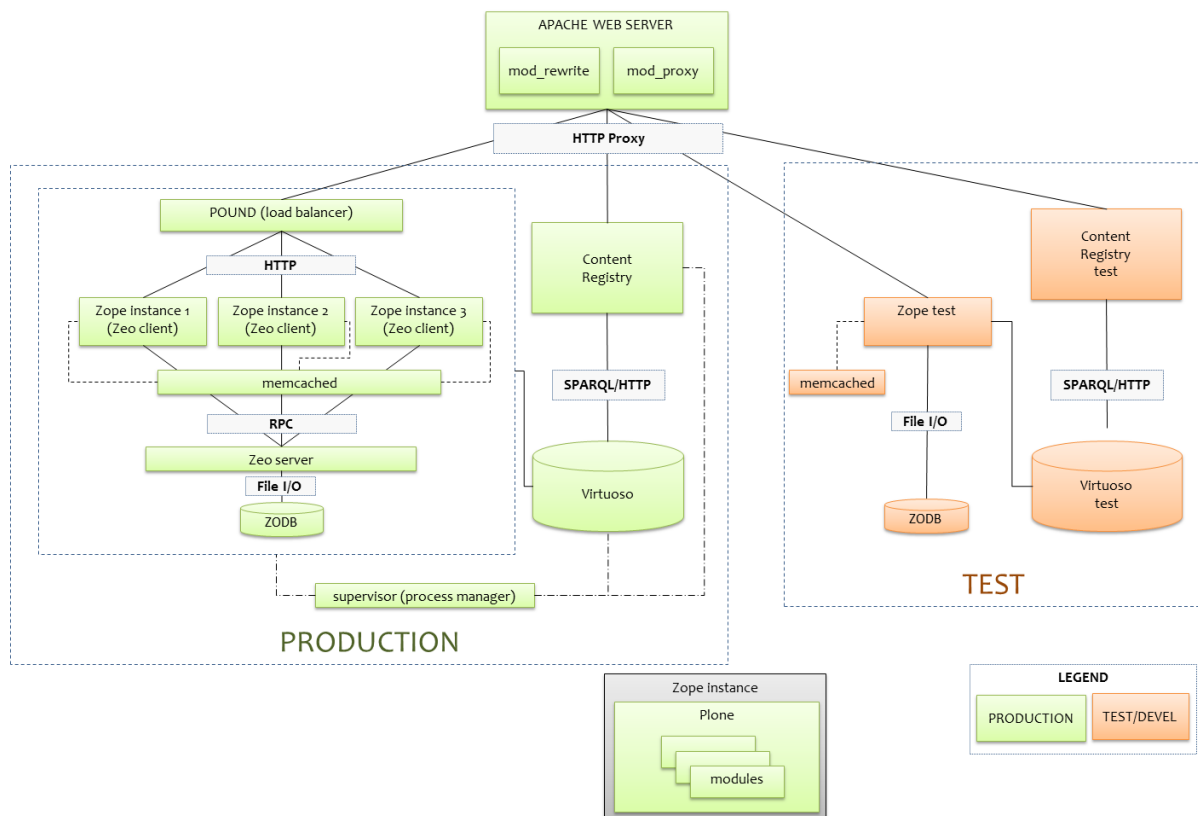
### 1.1. Domain names

PRODUCTION	
digital-agenda-data.eu	Visualisation website (Plone)
www.digital-agenda-data.eu	Alias for digital-agenda-data.eu
digital-agenda-data.eu/data	Content Registry
virtuoso.digital-agenda-data.eu	OpenLink Virtuoso
semantic.digital-agenda-data.eu	ELDA, the linked data API tool also this namespace is used by custom URI's
TEST	
test.digital-agenda-data.eu	Visualisation website (Plone)
test-cr.digital-agenda-data.eu	Content Registry
test-virtuoso.digital-agenda-data.eu	OpenLink Virtuoso

\* All these domains are virtual hosts and point to a single IP address (85.9.22.69) but could be moved to different hosts if required.

### 1.2. Deployment diagram

The main software components and their interaction are depicted in the diagram below:



### 1.3. Operating system, system tools, libraries and prerequisites

All software is running in a Linux server (CPU: Intel® Quad Core™ i7-4770 HyperThreading, RAM: 16 GB DDR3, RAID 1 2\*2 TB SATA):

- Linux CentOS 7.2 64bit
- Kernel 3.10.0-327.4.5.el7.x86\_64 (updated constantly).

#### 1.3.1. Libraries, tools, prerequisites

An updated full installation script is maintained at <https://github.com/digital-agenda-data/scoreboard.vagrant>. The repository contains a Vagrantfile<sup>1</sup> configuration together with necessary scripts and configuration files required to create from scratch a complete development server.

The main installation script is called “[bootstrap.sh](#)” and contains command-line instructions for the download, compilation and installation of each software component.

#### 1.3.2. Scheduled tasks

In addition to the backup and system monitoring jobs, cron jobs are used to generate the RDF, CSV and TSV exports for each datasets. The last modification time of each dataset is verified before exporting. Because of their relative large size, the files are archived (zip) and served as static files directly by Apache http server.

The crontab for user scoreboard shows:

```
30 23 * * * /var/local/plone/export/export_datasets_prod.sh
15 23 * * * /var/local/test-plone/export/export_datasets_test.sh
```

<sup>1</sup> <https://www.vagrantup.com/>

## 1.4. Major software components

The following software components have not been developed specifically for the Digital Agenda Scoreboard project.

### 1.4.1. Virtuoso Open-Source Edition

This component is used as *storage engine* for semantic and relational data.

<b>Vendor</b>	OpenLink Software
<b>Open Source</b>	Yes, GPL v2 ( <a href="http://www.openlinksw.com/dataspace/doc/dav/wiki/Main/VOSLicense">http://www.openlinksw.com/dataspace/doc/dav/wiki/Main/VOSLicense</a> )
<b>Technology</b>	native/mixed
<b>Home page</b>	<a href="http://www.openlinksw.com/dataspace/doc/dav/wiki/Main">http://www.openlinksw.com/dataspace/doc/dav/wiki/Main</a> <a href="http://docs.openlinksw.com/virtuoso">http://docs.openlinksw.com/virtuoso</a>
<b>Source code</b>	<a href="https://github.com/openlink/virtuoso-opensource">https://github.com/openlink/virtuoso-opensource</a>
<b>Current version</b>	07.20.3217-pthreads (release 7.2.4.2)
<b>Management interface</b>	<a href="http://virtuoso.digital-agenda-data.eu">http://virtuoso.digital-agenda-data.eu</a> (production instance) <a href="http://test-virtuoso.digital-agenda-data.eu">http://test-virtuoso.digital-agenda-data.eu</a> (test instance)
<b>Installation details</b>	Home dir: /var/local/virtuoso (production) and /var/local/test-virtuoso (test) Data files (production): /var/local/virtuoso/var/lib/virtuoso/db/ Data files (test): /var/local/test-virtuoso/var/lib/virtuoso/db/  Disk size: 2GB (production) + 2GB (test)

### 1.4.2. Content Registry

This component is used to maintain and browse through data and metadata

<b>Vendor</b>	TripleDev, founded by the European Environment Agency and DG-Connect
<b>Open Source</b>	Yes
<b>Technology</b>	Java
<b>Source code</b>	<a href="https://github.com/digital-agenda-data/scoreboard.contreg">https://github.com/digital-agenda-data/scoreboard.contreg</a>
<b>Management interface</b>	<a href="http://digital-agenda-data.eu/data">http://digital-agenda-data.eu/data</a> (production instance) <a href="http://test-cr.digital-agenda-data.eu">http://test-cr.digital-agenda-data.eu</a> (test instance)
<b>Installation details</b>	Home dir (production): /var/local/cr Home dir (test): /var/local/test-cr  Disk size (without staging data files): 100 MB (production) + 100 MB (test)

### 1.4.3. Plone

This component is used as a content management framework and application server for the visualisation website.

<b>Vendor</b>	The Plone Foundation
<b>Open Source</b>	Yes
<b>Technology</b>	Python
<b>Home page</b>	<a href="http://plone.org/">http://plone.org/</a>
<b>Source code</b>	<a href="https://github.com/plone">https://github.com/plone</a>
<b>Current version</b>	Plone 4.3, Zope 2.13.19
<b>Installation details</b>	Home dir (production): /var/local/plone Home dir (test): /var/local/test-plone  Disk size: ~ 700 MB (production) + 700 MB (test)

### 1.4.4. ELDA

An implementation of the Linked Data API

<b>Vendor</b>	Epimorphics Ltd.
<b>Open Source</b>	Yes
<b>Technology</b>	Java
<b>Home page</b>	<a href="http://www.epimorphics.com/web/tools/elda.html">http://www.epimorphics.com/web/tools/elda.html</a>
<b>Source code</b>	<a href="https://code.google.com/p/elda/source/">https://code.google.com/p/elda/source/</a>
<b>Current version</b>	1.2.16

<b>Web interface</b>	<a href="http://semantic.digital-agenda-data.eu/">http://semantic.digital-agenda-data.eu/</a>
<b>Installation details</b>	Home dir (production): /var/local/elda  Disk size: 100 MB

### 1.4.5. Apache Web Server

This is the public web server and reverse proxy that stands in front of all application servers.

<b>Vendor</b>	The Apache Software Foundation
<b>Open Source</b>	Yes
<b>Technology</b>	Native
<b>Home page</b>	<a href="http://httpd.apache.org/">http://httpd.apache.org/</a>
<b>Current version</b>	2.2.15
<b>Web interface</b>	<a href="http://digital-agenda-data.eu/">http://digital-agenda-data.eu/</a> (production) <a href="http://test.digital-agenda-data.eu/">http://test.digital-agenda-data.eu/</a> (test)
<b>Installation details</b>	Web dir: /var/www/  Disk size: 100 MB
<b>Virtual Hosts</b>	Production: <ul style="list-style-type: none"> <li>▪ digital-agenda-data.eu (main website)</li> <li>▪ semantic.digital-agenda-data.eu (Linked Data API)</li> <li>▪ virtuoso.digital-agenda-data.eu (Virtuoso)</li> </ul> Test: <ul style="list-style-type: none"> <li>▪ test.digital-agenda-data.eu (website)</li> <li>▪ test-cr.digital-agenda-data.eu (Content Registry)</li> <li>▪ test-virtuoso.digital-agenda-data.eu (Virtuoso)</li> </ul>

## 1.5. Source code

The following components have been developed specifically for the Digital Agenda Scoreboard project:

Description	Technology	URL
Installation scripts for the entire solution on a development environment using Vagrant and VirtualBox	scripts	<a href="https://github.com/digital-agenda-data/scoreboard.vagrant">https://github.com/digital-agenda-data/scoreboard.vagrant</a>
Build scripts for Plone	Python	<a href="https://github.com/digital-agenda-data/scoreboard.buildout">https://github.com/digital-agenda-data/scoreboard.buildout</a>
Visualisation website	Javascript (96%), Python	<a href="https://github.com/digital-agenda-data/scoreboard.visualization">https://github.com/digital-agenda-data/scoreboard.visualization</a>
Plone theme, CSS stylesheets, front-end widgets (e.g. navigation)	Javascript (55%), CSS (43%), Python	<a href="https://github.com/digital-agenda-data/scoreboard.theme">https://github.com/digital-agenda-data/scoreboard.theme</a>
Backend components (data access, SPARQL queries)	Python	<a href="https://github.com/digital-agenda-data/edw.datacube">https://github.com/digital-agenda-data/edw.datacube</a>
Content Registry with modifications for the Digital Agenda Scoreboard		<a href="https://github.com/digital-agenda-data/scoreboard.contreg">https://github.com/digital-agenda-data/scoreboard.contreg</a>
RDF data model and initial content of the triple store	RDF	<a href="https://github.com/digital-agenda-data/rdf">https://github.com/digital-agenda-data/rdf</a>
Various utility scripts used in day-to-day maintenance	scripts	<a href="https://github.com/digital-agenda-data/scripts">https://github.com/digital-agenda-data/scripts</a>

### 1.5.1. Data files

The following files need to be copied and replaced in the case of a server migration in order to keep all current data. They cannot be restored by other means except a clean re-upload of all datasets:

<b>PRODUCTION</b>
-------------------



/var/local/virtuoso/var/lib/virtuoso/db/ <b>virtuoso.db</b>	2.2 GB	Semantic data
/var/local/plone/scoreboard.buildout/var/ <b>filestorage/Data.fs</b>	60 MB	ZODB database, contains metadata for all objects (datasets, charts, users, settings, etc.) created in the visualisation website
/var/local/plone-scoreboard/scoreboard.buildout/var/ <b>blobstorage</b>	10 MB	Directory containing binary objects (images, attachments, etc.)
<b>TEST</b>		
/var/local/test-virtuoso/var/lib/virtuoso/db/ <b>virtuoso.db</b>	2.2 GB	Semantic data
/var/local/test-plone/scoreboard.buildout/var/ <b>filestorage/Data.fs</b>	60 MB	ZODB database, contains metadata for all objects (datasets, charts, users, settings, etc.) created in the visualisation website
/var/local/test-plone/scoreboard.buildout/var/ <b>blobstorage</b>	10 MB	Directory containing binary objects (images, attachments, etc.)
/etc/httpd/ssl/*	< 1 MB	SSL certificates

The following data files can be copied, but can also be re-created:

<b>PRODUCTION</b>		
/var/local/cr/apphome/staging/*		Staging files uploaded in Content Registry (e.g. *.mdb)
/var/local/cr/apphome/acl/*		Access Control Lists (ACLs) designating access permissions of users and groups to the functionality of Content Registry.
/var/local/cr/apphome/filestore/*		Files uploaded by users, and also content of the dynamically editable documentation sections of the Content Registry user interface.
/var/www/html/download/*	50 MB	Datasets exported as csv, tsv and ttl (e.g. as listed in page <a href="http://digital-agenda-data.eu/datasets/digital_agenda_scoreboard_key_indicators#download">http://digital-agenda-data.eu/datasets/digital_agenda_scoreboard_key_indicators#download</a> )
<b>TEST</b>		
/var/local/test-cr/apphome/staging/*		Staging files uploaded in CR test (e.g. *.mdb)
/var/local/test-cr/apphome/acl/*		Access Control Lists (ACLs) designating access permissions of users and groups to the functionality of Content Registry.
/var/local/test-cr/apphome/filestore/*		Files uploaded by users, and also content of the dynamically editable documentation sections of the Content Registry user interface.
/var/www/test-html/download/*	50 MB	Datasets exported as csv, tsv and ttl

Some of the above files can be re-created if lost:

- The files in /var/local/(test-)cr/apphome/staging/\* can be downloaded or obtained in other ways from their original providers, such as Eurostat. For example the latest MS Access statistics on ICT survey can be downloaded from [http://epp.eurostat.ec.europa.eu/portal/page/portal/information\\_society/data/comprehensive\\_databases](http://epp.eurostat.ec.europa.eu/portal/page/portal/information_society/data/comprehensive_databases). They can be downloaded directly into the above location in Content Registry file system, or they can be downloaded via the "Staging files" section in Content Registry's "Admin actions".
- The files in /var/local/(test-)cr/apphome/acl/\* can be re-created when building Content Registry from source code and providing the corresponding folder path in build properties (see below).
- The files in /var/local/(test-)cr/apphome/filestore/\* can only be re-created by users re-uploading them via dedicated sections in Content Registry web interface.

### 1.5.2. SSL certificates

The domain digital-agenda-data.eu uses a single wildcard certificate for all HTTPS subdomains. This has to be configured in Apache web server (see section below). The following files (from /etc/httpd/ssl) need to be restored:

- STAR\_digital-agenda-data\_eu.key
- STAR\_digital-agenda-data\_eu.crt
- STAR\_digital-agenda-data\_eu.ca-bundle

The current certificate expires on 13.05.2017.

### 1.5.3. System services

The following systemd services start automatically on reboot:

Service name Production	Service name Test	Purpose	Configuration
httpd		Apache, the web server	default + SSL + <a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/scoreboard-prod.conf">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/scoreboard-prod.conf</a> <a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/scoreboard-test.conf">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/scoreboard-test.conf</a>
supervisord	supervisord-test	Orchestrates sub-processes (plone instances, memcached, pound, zeo)	<a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/supervisord.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/supervisord.service</a> <a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/supervisord-test.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/supervisord-test.service</a>
virtuoso7	virtuoso7-test	Virtuoso, the semantic database	<a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/virtuoso7.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/virtuoso7.service</a> <a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/virtuoso7-test.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/virtuoso7-test.service</a>
cr	cr-test	Content Registry (data maintenance tool)	<a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/cr.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/cr.service</a> <a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/cr-test.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/cr-test.service</a>
mariadb		Used by Piwik, the web analytics tool	default
elda		Linked data API	<a href="https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/elda.service">https://github.com/digital-agenda-data/scoreboard.vagrant/blob/master/etc/elda.service</a>

## 2. Installation from source code

### 2.1. In a virtualized environment

In order to install all applications in a virtual machine using Vagrant and VirtualBox, use the scripts maintained at <https://github.com/digital-agenda-data/scoreboard.vagrant> (connection to the Internet is required):

- Install Vagrant <https://www.vagrantup.com/downloads.html> with support for VirtualBox <https://www.virtualbox.org/wiki/Downloads>
- git clone <https://github.com/digital-agenda-data/scoreboard.vagrant.git>
- then follow the instructions from the README file:
  - `vagrant box add centos/7 --provider virtualbox`
  - `vagrant plugin install vagrant-vbguest`
  - `vagrant plugin install vagrant-cachier`
  - `vagrant up`

The installation script can be used with small adaptations on a physical server (without Vagrant).

### 2.2. In a pre-configured server

In order to install the DAD applications in an existing environment (assumed to be CentOS7 or compatible), use the example installation scripts for Plone, Virtuoso and Content Registry:

- `install_plone_novagrant.sh`
- `install_virtuoso_novagrant.sh`
- `install_cr_novagrant.sh` (requires Virtuoso to be installed first)

The scripts should be executed as root.

Each script contains at the beginning of the file various configuration settings (e.g. home directory, ports, etc.) which can be used to switch between production and test configurations.

#### Important note:

After installation, either edit your local hosts file and map digital-agenda-data.eu domain to the IP of your test server, or, alternatively, edit `/etc/httpd/conf.d/scoreboard-prod.conf` and change all occurrences of “digital-agenda-data.eu” to the actual hostname of the server, including in lines similar to the ones below:

```
ProxyPass / http://localhost:8440/VirtualHostBase/http/digital-agenda-data.eu:80/Plone/VirtualHostRoot/ timeout=300 retry=0
```

```
RewriteRule ^/(.*) http://localhost:8448/VirtualHostBase/http/test.digital-agenda-data.eu:80/Plone/VirtualHostRoot/$1 [P]
```

### 2.3. Testing installation

To test that all applications were correctly installed, use the following commands:

#### 2.3.1. Plone

- a. check the output of `supervisorctl`:

```
[root@localhost]# /var/local/plone/bin/supervisorctl status
instance1          RUNNING pid 32179, uptime 0:00:10
```

```
instance2          RUNNING  pid 32182, uptime 0:00:09
instance3          RUNNING  pid 32183, uptime 0:00:08
memcached          RUNNING  pid 31277, uptime 0:56:29
pound              RUNNING  pid 31808, uptime 0:25:58
zeo                RUNNING  pid 31805, uptime 0:25:59
```

for test environment use:

```
[root@localhost]# /var/local/test-plone/bin/supervisorctl status
instance          RUNNING  pid 9800, uptime 5:57:50
memcached         RUNNING  pid 9799, uptime 5:57:50
```

b. Check open ports:

```
[root@localhost]# netstat -tln | grep -E "(844[0-9])|(1121[0-1])|(889[0-1])|(111[1-2])|(808[0-1])"
```

production:

```
# Virtuoso
tcp    0    0 0.0.0.0:1111        0.0.0.0:*          LISTEN  21431/virtuoso-t
tcp    0    0 0.0.0.0:8890        0.0.0.0:*          LISTEN  21431/virtuoso-t
# Plone
tcp    0    0 127.0.0.1:8440      0.0.0.0:*          LISTEN  31808/pound
tcp    0    0 127.0.0.1:8441      0.0.0.0:*          LISTEN  32179/python
tcp    0    0 127.0.0.1:8442      0.0.0.0:*          LISTEN  32182/python
tcp    0    0 127.0.0.1:8443      0.0.0.0:*          LISTEN  32183/python
tcp    0    0 0.0.0.0:11210       0.0.0.0:*          LISTEN  31282/memcached
# Content Registry
tcp6   0    0 :::8080             :::*                LISTEN  21557/java
```

test:

```
# Virtuoso
tcp    0    0 0.0.0.0:1112        0.0.0.0:*          LISTEN  21452/virtuoso-t
tcp    0    0 0.0.0.0:8891        0.0.0.0:*          LISTEN  21452/virtuoso-t
# Plone
tcp    0    0 0.0.0.0:11211       0.0.0.0:*          LISTEN  9802/memcached
tcp    0    0 0.0.0.0:8448        0.0.0.0:*          LISTEN  9800/python
# Content Registry
tcp6   0    0 :::8081             :::*                LISTEN  22129/java
```

c. Check service status - all should print “active (running)”:

**Production:**

```
systemctl status httpd
systemctl status supervisor
systemctl status cr
systemctl status virtuoso7
```

**Test:**

```
systemctl status supervisor-test
systemctl status cr-test
systemctl status virtuoso7-test
```

## 2.4. Log files

For troubleshooting, check the following log files (assuming the default home directories have been used):

Category	Application	Log file(s)
----------	-------------	-------------

Production	HTTPD web server	/var/log/httpd/digital-agenda-data_log
	Plone	/var/local/plone/var/log/instance{1,2,3}.log
	Content Registry	/var/local/cr/tomcat/logs/catalina.out
	Virtuoso	/var/local/virtuoso/var/lib/virtuoso/db/virtuoso.log
Test	Plone	/var/local/test-plone/var/log/instance.log
	Content Registry	/var/local/test-cr/tomcat/logs/catalina.out
	Virtuoso	/var/local/virtuoso/var/lib/virtuoso/db/virtuoso.log

## 2.5. Application-specific settings

### 2.5.1. Plone

The following settings are explained in more details in deliverable D2 - Technical Report:

- Open <https://digital-agenda-data.eu> and login using an administrator account using the Login link in the footer
- Check the SMTP settings in page <http://digital-agenda-data.eu/mail-controlpanel>
- Check the reCAPTCHA keys in page <http://digital-agenda-data.eu/recaptcha-settings>
- Check the email addresses that receive notifications when comments are posted, in page [http://digital-agenda-data.eu/ploneboard\\_notification](http://digital-agenda-data.eu/ploneboard_notification)
- Check the properties in page [http://digital-agenda-data.eu/portal\\_registry/](http://digital-agenda-data.eu/portal_registry/) (select prefix IDataCubeSettings from the dropdown list). In particular, the test instance must have different values for parameters DEFAULT\_CR\_URL, DEFAULT\_SPARQL\_ENDPOINT, DEFAULT\_USER\_SPARQL\_ENDPOINT

### 2.5.2. Content Registry

**NB!** An important property to set is *application.homeDir*. This is the root directory of CR's resource files that are needed at run-time. The default value of the production instance is “/var/local/cr/apphome”

In addition to *application.homeDir*, you should normally only need to check these properties:

- *application.homeURL* (further commented in the file)
- *deployment.host* (further commented in the file)
- *mail.\** (further commented in the file)
- *virtuoso.db.url* (check Virtuoso host name and port number)
- *virtuoso.db.usr*, *virtuoso.db.pwd*, *virtuoso.db.rouser*, *virtuoso.db.ropwd* (use the user names and passwords in the *1\_create\_users.sql* file that you will import into Virtuoso below)

The users and access permissions of CR are stored in "acl" directory under app-home. If the value of *useCentralAuthenticationService* build property was set to "false", then users and their passwords are configured in the self-explanatory *users.xml* file. Users' membership in groups can be configured in *cr.groups.xml* file. For example, to grant certain users access to all admin-level functionality in CR, simply add them into the *cr\_admin* group.