	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

EAGLE
EnhAnced Government LEarning

www.fp7-eagle.eu

FP7-ICT-2013-11

Objective 8.2 Technology-enhanced learning;

Target outcome c): Holistic learning solutions for managing, reaching
and engaging learners in the public administrations

Deliverable 5.2

OER Data Registry

WP 5 – OER Learning Platform Development
Lead Participant: Fraunhofer

Approval Panel	Name / Partner short name	Department / Function	Date
Author	Arun Prakash, Santosh Kumar Rajaguru / FOKUS	System Quality Engineering (SQC) / WP5 Leader Researcher	11/03/2016
Reviewer	Muriel Foulonneau/ LIST	Embedded Assessment Research Group / Senior Researcher	08/04/2016
Approval	Sabine Moebs / DHBW	Professor, Quality Manager	28/04/2016



This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement N°310806.



	<p style="text-align: center;">Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p>Contract Number 619347</p>		Version 1.0


Table of Contents

LIST OF FIGURES	II
GLOSSARY	III
1 SOFTWARE DETAILS	4
2 OVERVIEW OF THE PROTOTYPE	5
2.1 High-level System Architecture	5
2.2 Integration within the EAGLE Platform.....	7
2.2.1 Prerequisites.....	7
2.2.2 Building, installing.....	7
3 INSTALLATION GUIDE	8
3.3 Prerequisites.....	8
3.4 Configurations.....	8
3.5 Installation Steps	9
4 USAGE	11
REFERENCES	15

	<p style="text-align: center;">Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p>Contract Number 619347</p>		Version 1.0


List of Figures

Figure 2-1: Overall Stack of Harvestino	6
Figure 2-2: Modular workflow of Harvester controller.....	6
Figure 3-1: Target definition	8
Figure 3-2: Packaging the Harvestino Through Product	9
Figure 4-1: Login Page	11
Figure 4-2: Harvester and its status view	11
Figure 4-3: Harvestino - List of Perspectives	12
Figure 4-4: Harvester Configuration View	12
Figure 4-5: User Management Perspective.....	13
Figure 4-6: User Configuration View	13
Figure 4-7: Analytics View of Cultural Data.....	14

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0


Glossary

FP7	Seventh Framework Programme
EC	European Commission
EAGLE	EnhAnced Government LEarning
EU	European Union
WP	Work Package
OSGi	Open Service Gateway Initiative
RAP	Remote Application Platform
EMF	Eclipse Modelling Framework
MDE	Model Driven Engineering
LOM	Learning Object Meta-data
OER	Open Educational Resource
HCM	Harvester Control Module
JDK	Java Development Kit
ETL	Extract Transform Load
SWORD	Simple Web-service Offering Repository Deposit
OAI-PMH	Open Archives Initiative - Protocol for Metadata Harvesting
FTP	File Transfer Protocol
DC	Dublin Core

	<p style="text-align: center;">Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p>Contract Number 619347</p>		Version 1.0

1 Software Details

Title of the Software	Harvestino
Version	0.7 (internal version)
Release Date	11 March 2016
Relevant Licenses	GNU General Public License (GPL)
Programming Languages	Java
Source code	https://gitlab.fokus.fraunhofer.de/sqc/odp.git
Developers	Arun Prakash, Santosh Kumar Rajaguru

	<p style="text-align: center;">Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p style="text-align: center;">Contract Number 619347</p>		Version 1.0

2 Overview of the Prototype

WP5 task T5.2 deals with the creation of an open-source OER Data Registry. As the EAGLE system is based on Liferay, storage and retrieval of the OERs created within the scope of the EAGLE portal is stored in the content management system (CMS) provided by Liferay. Furthermore, Apache Marmotta is used for knowledge management and to provide semantic services in the EAGLE system. Thus, meta-data information for the OERs created through the EAGLE portal is stored with Apache Marmotta.

In the case of harvesting, only the meta-data information about the OERs stored in external repositories are harvested. This way, the copyright of the content are respected. Additionally, this avoids replication of content across the EAGLE portal servers. Apache Marmotta which serves as the knowledge management module ensures that the data and meta-data are correctly linked. Thus portlets rendering the content only retrieve the content from the original locations.

As the EAGLE Ontology for Local Governments (refer D5.3 and Chapter 3 in D5.5) serves as the binding data / knowledge model for the EAGLE system, OER meta-data schemas such as LOM (Learning Object Metadata) have been integrated directly into the ontology.


External repositories provide their data / meta-data in a variety of ways (OAI-PMH, FTP, etc) for the same kind of data type (DC or LOM). Thus a mechanism with configurable data providers is needed. Additionally, the choice of the backend (where the harvested meta-data is stored) should be configurable as backend requirements change from time to time and with the introduction of new technology. In the case of EAGLE, the backends for the harvesters are the CMS from Liferay and Apache Marmotta. Thus RESTful API are used to deposit the harvested meta-data resources instead of SWORD (Simple Web-service Offering Repository Deposit).

2.1 High-level System Architecture

Harvestino is an application focusing on the harvesting of open data in a scalable and modular manner. Figure 2-1 illustrates the overall stack diagram of the Harvestino. The state-of-art features of Harvestino eases the ETL process and provides a platform to configure various types of data providers as well as backend providers.

The design and development of the architecture uses the concepts of OSGi [1] and model-driven engineering. For each data providers, a single harvester module is implemented that extends the abstract super harvester. The mechanism of data management illustrated in Figure 2 shows the overall workflow of the harvester controller. As pictured in Figure 2-2, the harvesting mechanism comprises of five simple steps.

1. First, the data is extracted from the data source with the help of data providers.
2. The extracted data is immediately validated for syntactical and semantical error. The errors obtained from the validation process are logged and reported back to the data provider.
3. Upon successful validation, the input data is transformed to a target meta-model that is OER model in the case of EAGLE.
4. The transformed datasets are validated once again to check the transformation errors and
5. Finally pushed to the backend through the help of backend provider, for instance Marmotta knowledge service

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

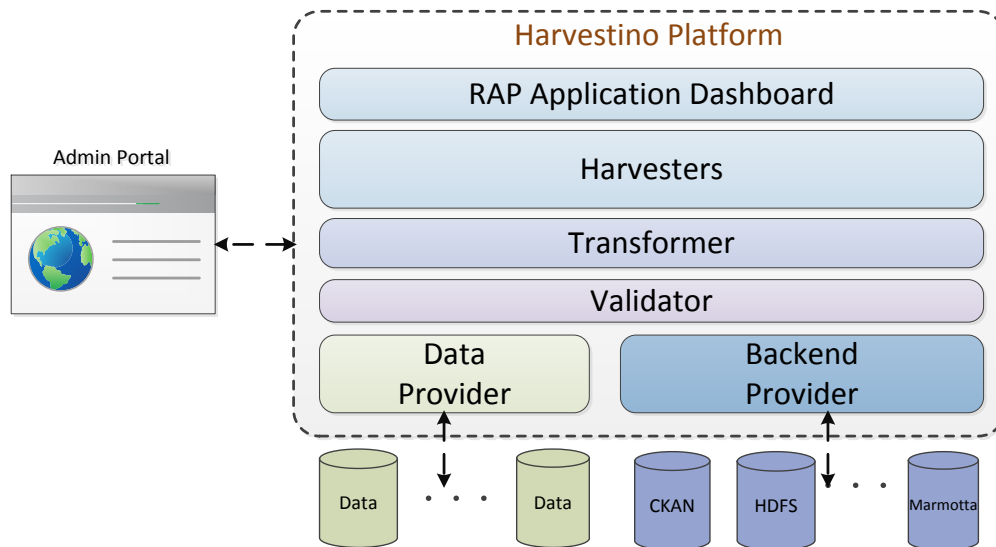


FIGURE 2-1: OVERALL STACK OF HARVESTINO

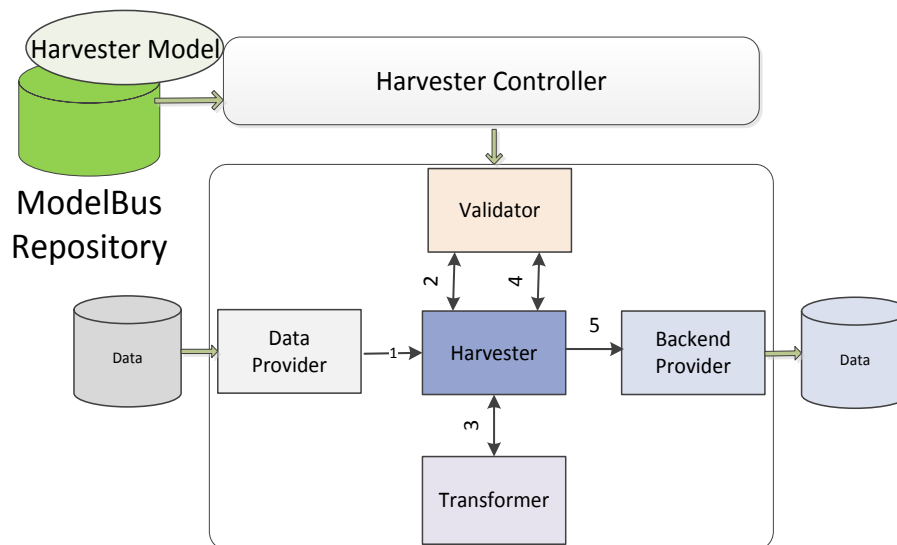



FIGURE 2-2: MODULAR WORKFLOW OF HARVESTER CONTROLLER

All the developed modules are based on OSGi extension mechanism and are operated by the Harvester Controller Module (HCM). HCM controls the configuration, scheduling as well as the operation of Harvesters. Individual Harvester are instantiated by the HCM through the Eclipse Extension Point mechanism [2]. This incorporates the *Dependency Inversion Principle* [3, 4] through the abstract Harvester class. To attain the cohesive as well as scalable nature, all the modules are designed based on SOLID Design principles [5]. Harvestino also provides inbuilt Authentication & Authorization to safeguard the security and integrity of the harvesters. As Harvestino uses Model-Driven Engineering concepts, all modules inherently work on models instead of raw data. The HCM controls the operation of modules based on the configurations (individual Harvester Models) stored in the ModelBus repository. ModelBus [6] serves as a model integration framework that provides the internal storage of harvester related configurations.

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

As Harvestino is built with OSGi, it provides features for hot deployment of modules through the OSGi console. Apart from the OSGi console, Harvestino also realizes a Remote Application Platform (RAP) [7] UI to provide a thin client with a rich widget set to administer the harvesters.

2.2 Integration within the EAGLE Platform

2.2.1 Prerequisites


The following API are used to extract the LOM data for harvesting into EAGLE platform.

- The host of the data provider: <http://monet.informatik.rwth-aachen.de>
- Record path: /openscout-live/repositoryOpenScoutMetadata/services/oai?
- Verb
 - List Record: verb=ListRecords&metadataPrefix=lom
 - List Set: verb=ListSets
 - Records of a particular set: verb=ListRecords&metadataPrefix=lom&set=

2.2.2 Building, installing

Harvestino is realized based on the OSGi platform and compatible for Maven. It also provides support for integrating the third party client through Rest API or configured plugin bundles. Harvestino uses Tycho [8] for managing the dependency. Unlike Harvestino, EAGLE Platform dependency management is done by Maven itself. Hence, to integrate the EAGLE service API jars with Harvestino, the following steps were followed:

- Successfully build the required “*knowledge*” and “*service*” projects.
- Install the projects into the environment.
- Created dependent plugins using the required jars.

	<p style="text-align: center;">Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p>Contract Number 619347</p>		Version 1.0

3 Installation Guide

These are the instructions for building and running the Harvestino Tool.

3.3 Prerequisites

The following software is required for building, installing and running the Harvestino Tool.

- Java version - JDK 1.8u73
- Git distributed version control system.
- Eclipse integrated modelling Tools (<https://eclipse.org/downloads/packages/eclipse-modeling-tools/lunar>)
- Modelbus Server v-1.9.8 , Modelbus TeamProvider v1.9.8 (<https://www.modelbus.org/en/modelbusdownloads.html>)
- Tycho connector for building the OSGi project
- Eclipse Product (<http://www.vogella.com/tutorials/EclipseProductDeployment/article.html>)
- m2e Maven for Eclipse for Development
- For deployment, Linux 64-bit is preferable
- Machine Requirements
 - No. of CPUS # 4
 - RAM memory 8GB

3.4 Configurations

- To setup and configure the Model bus, please go through the below link <https://www.modelbus.org/projects/ModelBus/static/custom/file/docs/modelbus/ModelBus UserGuide 0983 v1.9.9.pdf>.
- Setup and resolve the Target definition file for setting the eclipse environment. By clicking on the “Set as Target Platform” as shown in the Figure 3-1.

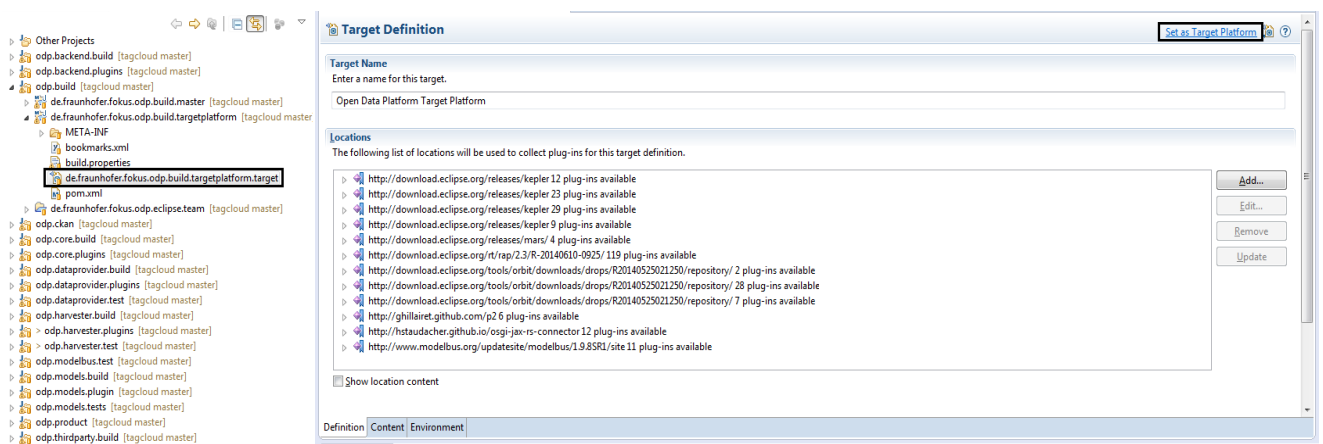



FIGURE 3-1: TARGET DEFINITION

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

- “odp.product” file is responsible for packaging the required Harvestino bundles. All the required bundles are configured into the odp.product configuration. To package these bundles, run the product file launcher in the project “de.fraunhofer.fokus.odp.product.build.master” as shown in the Figure 3-2Figure 3-2.

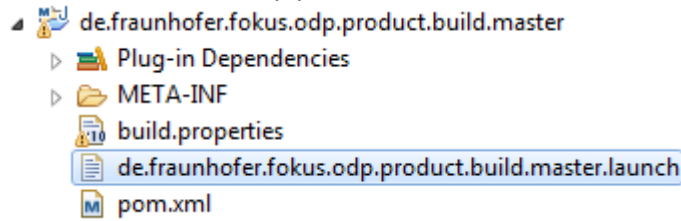


FIGURE 3-2: PACKAGING THE HARVESTINO THROUGH PRODUCT

If the launcher is not available, run the run the POM file with Goals “clean package” and Profiles “product”.

3.5 Installation Steps

After successful packaging of the Harvestino through Product configuration,

- Go to Production Server
- Create a folder named “ODPServer” under “/home/<ServerUserName>/ODPServer”
- Create a folder named “ModelBus” under “/home/<ServerUserName>/ModelBus”
- Follow the below URL for setup the modelbus server 1.9.8
<https://www.modelbus.org/projects/ModelBus/static/custom/file/docs/modelbus/ModelBus UserGuide 0983 v1.9.9.pdf>
- Create a “odp.sh” file under “/home/<ServerUserName>/Desktop” and paste the following commands

```

export MODELBUS_ROOT=/home/<ServerUserName>/ModelBus/Server/ModelBusServer-
Version1.9.8

export ODP_ROOT=/home/<ServerUserName>/ODPServer

echo "Executing from " $<Server_ODP_root>

cd $<Server_ODP_root>

chmod +x odp

./odp


```

- Copy the files under the below specified path to “ODPServer”
..de.fraunhofer.fokus.odp.product\target\products\de.fraunhofer.fokus.odp.product\linux\gtk\x86_64
- Create a “modelbus.sh” file under “/home/<ServerUserName>/Desktop” and paste the following commands

```

export MODELBUS_ROOT=/home/<ServerUserName>/ModelBus/Server/ModelBusServer-

```

	<p style="text-align: center;">Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p>Contract Number 619347</p>		Version 1.0

```
Version1.9.8

echo "Executing from " $<Server_Modelbus_root>

cd $<Server_ModelBus_root>

chmod +x startup.sh

• ./startup.sh
```


- From the console, execute the command

```
sudo sh /home/<ServerUserName>/Desktop/modelbus.sh

Console Output:
Starting service consumer:
org.modelbus.dosgi.repository.descriptor.RepositoryService
2016-03-11 10:21:45.438:INFO:oejs.Server:jetty-8.1.10.v20130312
2016-03-11 10:21:45.588:INFO:oejs.AbstractConnector:Started
SelectChannelConnector@0.0.0.0:8080
Service consumer started:
org.modelbus.dosgi.repository.descriptor.RepositoryService, target service at:
http://0.0.0.0:8080/modelbusrepository
Service org.modelbus.dosgi.repository.service.RepositoryServiceImpl started at:
http://10.147.69.15:8080/modelbusrepository
-----
ModelBus is up and running
Registering HTTP-Proxy servlet at /
Registering REST servlet at /modelbusrepository-rest
ModelBus Manager is starting...
```

- Upon successful running of modelbus, execute the odp.sh file using the below command:

```
sudo sh /home/<ServerUserName>/Desktop/odp.sh
```

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

4 Usage

Harvestino can be accessed through the link: <http://harvestino.fokus.fraunhofer.de/harvester>.

Step1: the login page



FIGURE 4-1: LOGIN PAGE

For example: for login ID: eagle, password: eagle

Step 2: The user Perspective

After successful login, the user will navigate to the user perspective. In the harvester view, the registered harvesters per user are shown and its corresponding status in the harvester status view.

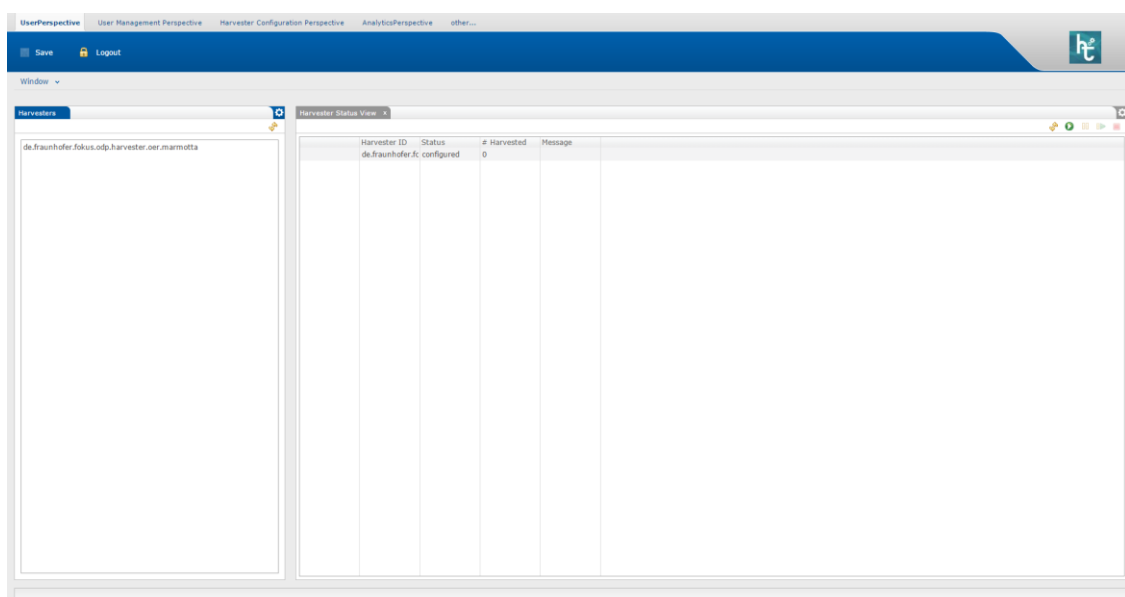



FIGURE 4-2: HARVESTER AND ITS STATUS VIEW

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

Step 3: Click on the “other...” which is placed on the top near user perspective.

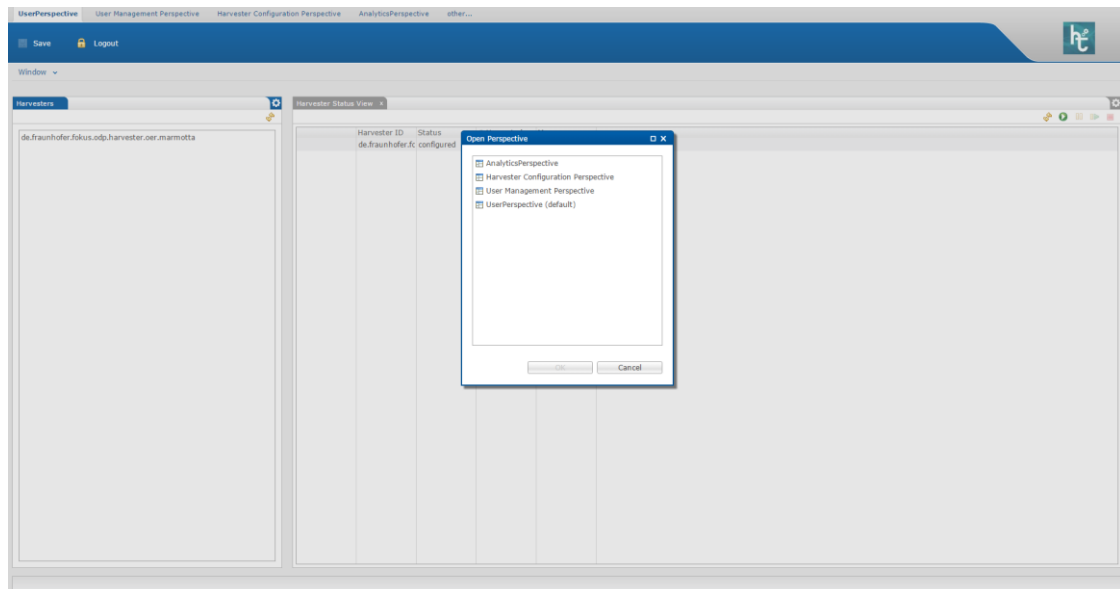


FIGURE 4-3: HARVESTINO - LIST OF PERSPECTIVES

It will show more perspectives that a user is authorized to view. It has Harvester Configuration Perspective, UserManagement Perspective and Analytics Perspective.

Step 4: Harvester Configuration Perspective:

It shows all the configurational parameters of a particular harvester, such as the registered harvester id, data source provider configuration such as host name of the provider as well as the backend provider configuration. For example in Figure 4-4, we show the Marmotta Backend Stub Configuration.

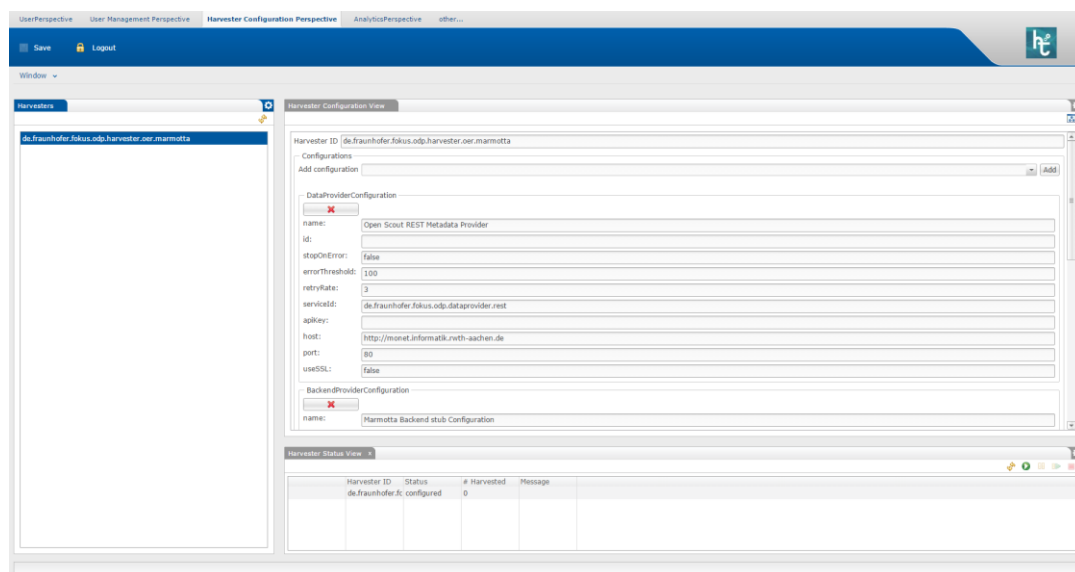


FIGURE 4-4: HARVESTER CONFIGURATION VIEW

 Contract Number 619347	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Version 1.0		

Step 5: User Management Perspective

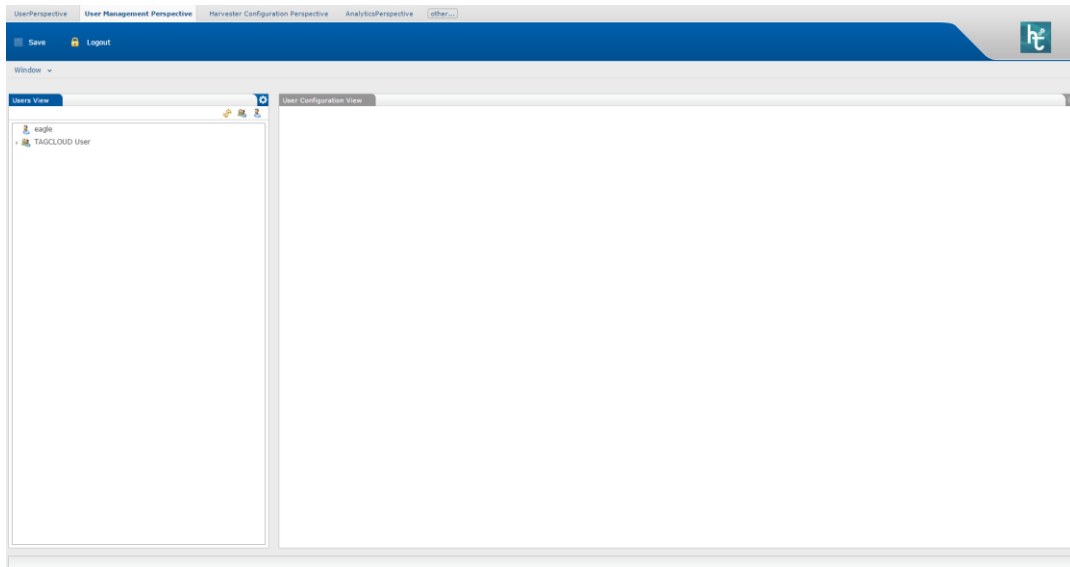


FIGURE 4-5: USER MANAGEMENT PERSPECTIVE

In the users view, the registered users are shown and if you click on a user, the user configuration view will show the respective user details.

User configuration view is displayed in the Figure 4-6

It contains all the information such as user name, user password email, rules and groups.

If a user wants to change the user credential and rules, he can do so. To save it into the persistence layer, the save button needs to be clicked.

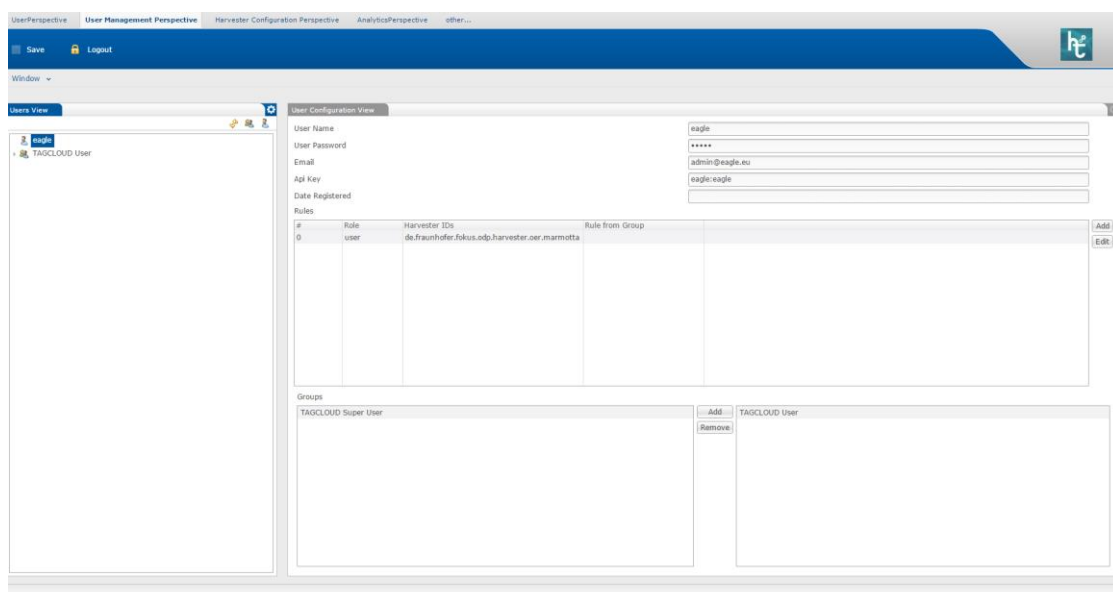



FIGURE 4-6: USER CONFIGURATION VIEW

	<p>Document Title OER Data Registry</p>	Deliverable Nature D
		Dissemination level PU
<p>Contract Number 619347</p>		Version 1.0

Step 6: Analytics Perspective

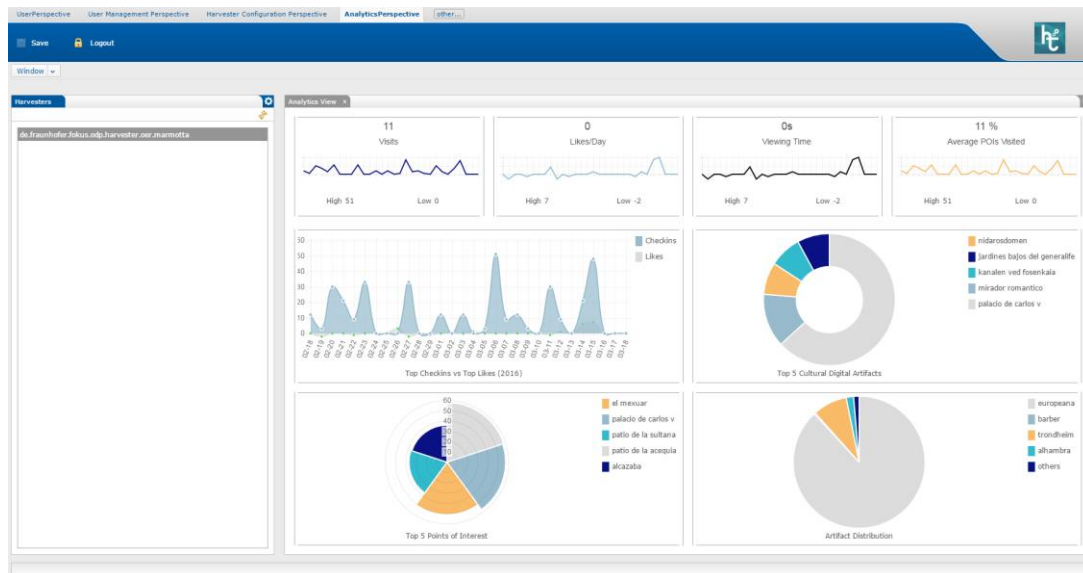



FIGURE 4-7: ANALYTICS VIEW OF CULTURAL DATA

This view shows the usage of the data / meta-data harvested by the various portal. Currently, no statistics are available for the EAGLE portal. As an example we showcase the analytics view for cultural meta-data harvested by the tool, as well as the artefact distributions.

Step 7: To logout from the Harvestino, click the “logout” button.

	Document Title OER Data Registry	Deliverable Nature D
		Dissemination level PU
Contract Number 619347		Version 1.0

References

- [1] Wikipedia. (2016) OSGi. [Online]. Available: <http://en.wikipedia.org/wiki/OSGi>
- [2] Vogella. (2014, Dec.) Eclipse Extension Mechanism. [Online]. Available: <http://www.vogella.com/tutorials/EclipseExtensionPoint/article.html>
- [3] L. Vogel. (2012, May) Using dependency injection in java - introduction - tutorial. [Online]. Available: <http://www.vogella.com/tutorials/DependencyInjection/article.html>
- [4] Wikipedia. (2016) Dependency Injection. [Online]. Available: https://en.wikipedia.org/wiki/Dependency_injection
- [5] R. C. M. U. BOB”, “Design principles and design patterns,” Last verified 2009-01-14. [Online]. Available: http://www.objectmentor.com/resources/articles/Principles_and_Patterns.pdf
- [6] A. Aldazabal, T. Baily, F. Nanclares, A. Sadovykh, C. Hein, M. Esser, and T. Ritter, *Automated Model Driven Development Processes*, ser. ISBN: 978-3-8167-7645-1, in Proceedings of the ECMDA workshop on Model Driven Tool and P. Integration, Eds. Fraunhofer IRB Verlag, Stuttgart, 2008.
- [7] Eclipse. (2016) Remote application platform. [Online]. Available: <http://www.eclipse.org/rap/>
- [8] ——. (2014, Dec.) Tycho. [Online]. Available: <https://eclipse.org/tycho/>