Smart Integration of Energy Storages in Local Multi Energy Systems for maximising the Share of Renewables in Europe's Energy Mix

Grant Agreement No. 730936
Start Date: 01/12/2016 – Duration 36 Months

Description of D6.5 – Instrumented search index server infrastructure

Work Package No./ Title	WP 6 Shared Data and Information Platform
Task No/title	T6.4: Setting up the search index and implementation of the search interface
Deliverable Number	D6.5
Delivery Date	M24 (shifted to M32)
Date of completion	28/10/2019
Document Type	DEC – Websites, patents, filings
Dissemination Level	PU - Public
Issued by	KIT
Author (s)	Clemens Düpmeier
Contributors	Shadi Shahoud, Isabelle J. Südmeyer

Summary

This document gives a short overview on the work performed in Task 6.4 for setting up and configuring the search infrastructure for SDIP. Liferay allows it to either use an internal or external Elasticsearch installation as the search engine for instrumenting search. Because of performance reason, an external installation of Elasticsearch is used for the SmILES search interface.

The SmILES portal uses an Elasticsearch cluster, which is part of the Energy Lab 2.0 infrastructure which also hosts the SmILES portal. For providing faceted search, several classification systems were added to the SmILES Liferay server. These classify Liferay contents according to the "Content Type", "Organisation" and "Scope" of contents. Then, a search page for SDIP was implemented, which uses these classifications for a facetted search.

The main goal of the setup of the SDIP search interface was to keep it very simple for the users. While testing the system, it turned out that the above mentioned facets together with the search slot for keywords already allow a very comfortable search for SDIP results. But the search infrastructure provides many more advanced features, which could be instrumented, if users aren't satisfied with the current search interface.

Approval

	First Author	1st Revision by	2nd Revision	Final Revision by WP leader	Project Coordinator
	Clemens Düpmeier			Clemens Düpmeier	Isabelle J. Südmeyer
Date	30/07/2019			28/10/2019	04/11/2019

1 Introduction

1.1 Purpose of the document

This document gives a short overview on the work performed in Task 6.4 for setting up and configuring the search infrastructure for SDIP. Liferay allows it to either use an internal or external Elasticsearch installation as the search engine for instrumenting search. Because of performance reason, an external installation of Elasticsearch will be used for the SmILES search interface.

The SmILES portal uses an Elasticsearch cluster, which is part of the Energy Lab 2.0 infrastructure which also hosts the SmILES portal. For providing faceted search, several classification systems were added to the SmILES Liferay server. These classify Liferay contents according to the "Content Type", "Organisation" and "Scope" of contents. Then, a search page for SDIP was implemented, which uses these classifications for a facetted search.

The main goal of the setup of the SDIP search interface was to keep it very simple for the users. While testing the system, it turned out that the above mentioned facets together with the search slot for keywords already allow a very comfortable search for SDIP results. But the search infrastructure provides many more advanced features, which could be instrumented, if users aren't satisfied with the current search interface.

The setup and configuration of the search index server and Liferay search interface mark the delivery of Deliverable D6.5 "Instrumented search index server infrastructure" and was already performed by the end of 2018, the configuration has slightly be actualized several times in 2019.

1.2 Scope of the document

The work description of Task 6.4 of the SmILES project defines the following subtasks:

- Set up of a scalable search index server for implementing the search interface (D6.5).
- Implementing functionality to automatically compile a structured search index from all the data and documents.

This document describes how the search index server was setup and integrated with the SmILES web portal for implementing the search interface, and how the portal server were configured to compile a search index for all SmILES contents.

1.3 Structure of the document

In Section 2, the overall architecture of the Shared Data and Information Platform (SDIP) is explained.

Section 3 describes how the search index server was setup and integrated with the SmILES web portal for implementing the search interface, and how the portal server were configured to compile a search index for all SmILES contents.

The document concludes with a short summary and outlook in Section 4.

2 Overall Architecture of the Shared Data and Information Platform (SDIP)

The Shared Data and Information Platform is fully integrated with the SmILES project web site and is implemented by using a Liferay web portal server as technical platform for delivering the web pages of the user interface. As shown in Figure 1, the SmILES web portal (e.g. the Liferay server and SDIP-UI-application layer) sits on top of the Energy Lab 2.0 ICT infrastructure as runtime infrastructure.

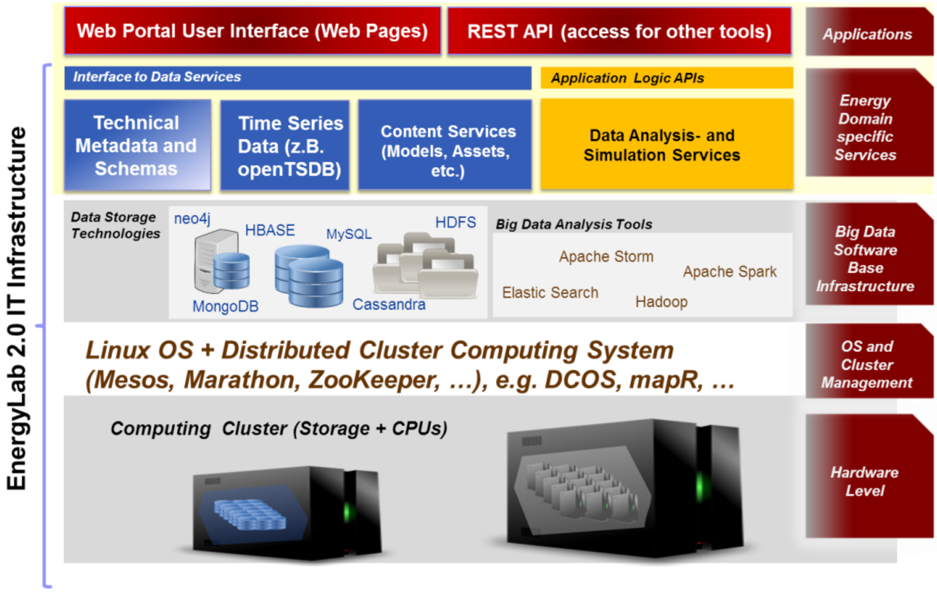

Figure 1: ICT Infrastructure for hosting the SDIP Platform

The Energy Lab 2.0 runtime infrastructure consists of several computing clusters for providing computing and storage capabilities for the software applications of the Energy Lab 2.0 at the KIT Campus North which is managed by the Institute of Automation and Applied Computer Science (IAI) at KIT.
One large storage cluster is used for Big Data Management and Big Data Analytics applications: e.g. it hosts several NoSQL database applications and a distributed HDFS filesystem (e.g. the distributed file system HDFS, MongoDB as document-oriented database, Hbase as column store, the Elasticsearch document-oriented database and index store, a MySQL cluster as relational database cluster, and a neo4J graph database as graph store) for large scale data management and several data analytics frameworks (e.g. Apache Spark, Storm, Hadoop and Elasticsearch) for building and running Big Data analytics applications. Another storage cluster, the Ceph-Cluster provides a large scale distributed filesystem and object (asset) storage by using the Ceph distributed filesystem software. Ceph provides low level API's to operating systems, i.e. a Ceph cluster can be used by virtualized computing nodes on a cluster as low level block devices for providing virtual storage disks for filesystems. It also supports standard network filesystem interfaces for Windows or Linux operating environments (e.g. NFS or SMB). Additionally, it supports cloud storage compatible interfaces (e.g. the Amazon S3 API) for modern Internet applications for storing structured objects and larger binary files in e.g. an own Cloud storage.
Third, the Energy Lab ICT infrastructure contains Linux and Windows computing clusters for server applications. The Linux cluster is equipped with the public domain Openstack cloud operating environment which allows running cloud infrastructure compatible applications on premise on own computing hardware. Especially, the Openstack cluster instruments Docker and Kubernetes for providing software container based automation of server applications (e.g. services, web applications, etc.).
The Openstack cluster is used as runtime environment for the service and application layer of the Energy Lab 2.0 software infrastructures (see the top 2 layers in Figure 1). At the one hand it hosts the Generic Data Services (GDS) for storing Energy related data and information files in the underlying database infrastructure layer. On the other hand, it provides dedicated services for running complex scientific calculations (e.g. forecasting or data analytics workflows) and/or complex simulation workflows (e.g. simulations, co-simulations) as distributed workflows on the underlying container infrastructure with a distributed scientific workflow engine called PROOF (PROcess Operation

Framework). Furthermore, the Docker and Kubernetes infrastructure is used for automating the management of running other applications, such as web-applications, e.g. the SmILES web portal.

Both, the GDS services and PROOF, are Microservice-oriented software applications developed by IAI as server-side backends for developing web-based engineering applications for complex energy system related research on top of these services.

The SmILES web portal and SDIP platform fully instruments the Energy 2.0 runtime infrastructure. The Liferay portal server which is the central web based user interface (see the Web-UI- and Application-layer in Figure 2) to the SmILES web site and SDIP platform runs as automated application on the Openstack cluster in the Docker / Kubernetes environment.

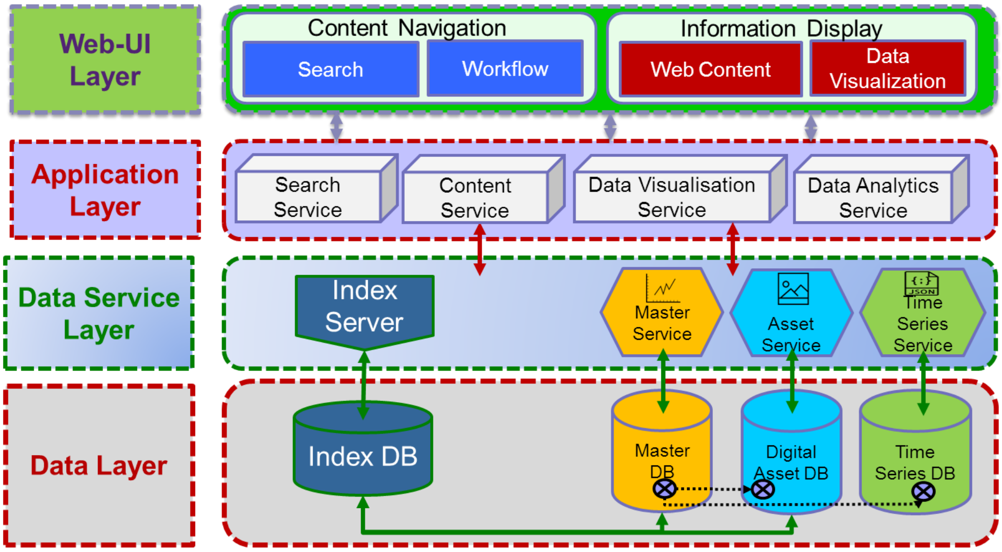

Figure 2: SmILES Portal and SDIP Architecture

An "Application Layer" situated in the application server of the Liferay server provides interfaces to the GDS services of the Energy Lab 2.0 infrastructure (Data Service Layer and Data Layer) for easily accessing and storing data and information in the GDS services. The "Application Layer" implements proxy APIs provided by Liferay for easing a secure and authorized access to the GDS services from web user components and JavaScript applications embedded into the Liferay provided web pages. The Application Layer implemented in the Liferay application server also implements adapter, which provide access to GDS content via standard Liferay APIs and internal interfaces for accessing web content and assets (The content and asset management APIs of Liferay). This allows web authors using the SmiLES Liferay portal server to create and access GDS content and assets as if they were internally stored within the Liferay server storage layer. Therefore, the storage ecosystem of the SmILES Liferay server provides users the opportunity to store and access information content either within the Liferay internal storage layer or within the GDS services using the same authoring and access environment.
Liferay itself provides very good measures for setting up structured content elements and adding arbitrary metadata to assets uploaded to the server via its content management interfaces (e.g. the standard authoring environment of Liferay). Those are used by SDIP to provide structured content and metadata forms to authors for uploading assets (e.g. binary data) and creating content elements according to the structural content specifications for SDIP contents as defined in Deliverable D6.2. The underlying implemented "Application Layer" transparently stores those content and asset elements either in Liferay or the GDS services of the Energy Lavb 2.0 infrastructure as introduced about.

3 Setting up the search infrastructure for the SmILES Web Portal

This section summarizes the work that has been done in Task 6.4 of work package WP6 for setting up a search service infrastructure for the SmILES web portal and how this is instrumented for use by SDIP. The Liferay server used for the SmILES web portal already contains an internal Elasticsearch server as search engine and index server which can be directly used as search engine for the SmILES project. But Liferay advises users, to set up an external Elasticsearch infrastructure for performance reasons. Because the Energy Lab 2.0 environment, on which the SmILES portal is hosted, already provides a high performance Elasticsearch installation, it was decided to use this installation also for the SmILES web portal.

The configuration of an external Elasticsearch as search engine for Liferay is quite easy. It can be done via the graphical administration user interface of a Liferay instance or by changing the Liferay Elasticsearch configuration file. The following configuration code shows the main part of the configuration file for switching to a remote Elasticsearch instance [3].

```
operationMode="REMOTE"
 # If running Elasticsearch from a different computer:
 #transportAddresses="ip.of.elasticsearch.node:9300"
 # Highly recommended for all non-production usage (e.g., practice, tests, diagnostics):
 #logExceptionsOnly="false"
```

Therefore, the configuration variable “operationMode” should be set to “REMOTE” and the variable “transportAdresses” should be set to a list of foreign IP-address, port pairs, where the remote Elasticsearch server or cluster could be accessed. Also an index dedicated for the Liferay server has to be created on the Elasticsearch cluster (e.g. the index “SmILES”) and the name of the index has to be set in the Elasticsearch configuration file of Liferay.

The Elasticsearch server and Liferay search interface supports a facetted search by defining different kinds of categories for classifying content and assets within Liferay. In context of SmILES, several categories were set up for the facetted search: “Content Types”, “Organisation” and “Scope”.

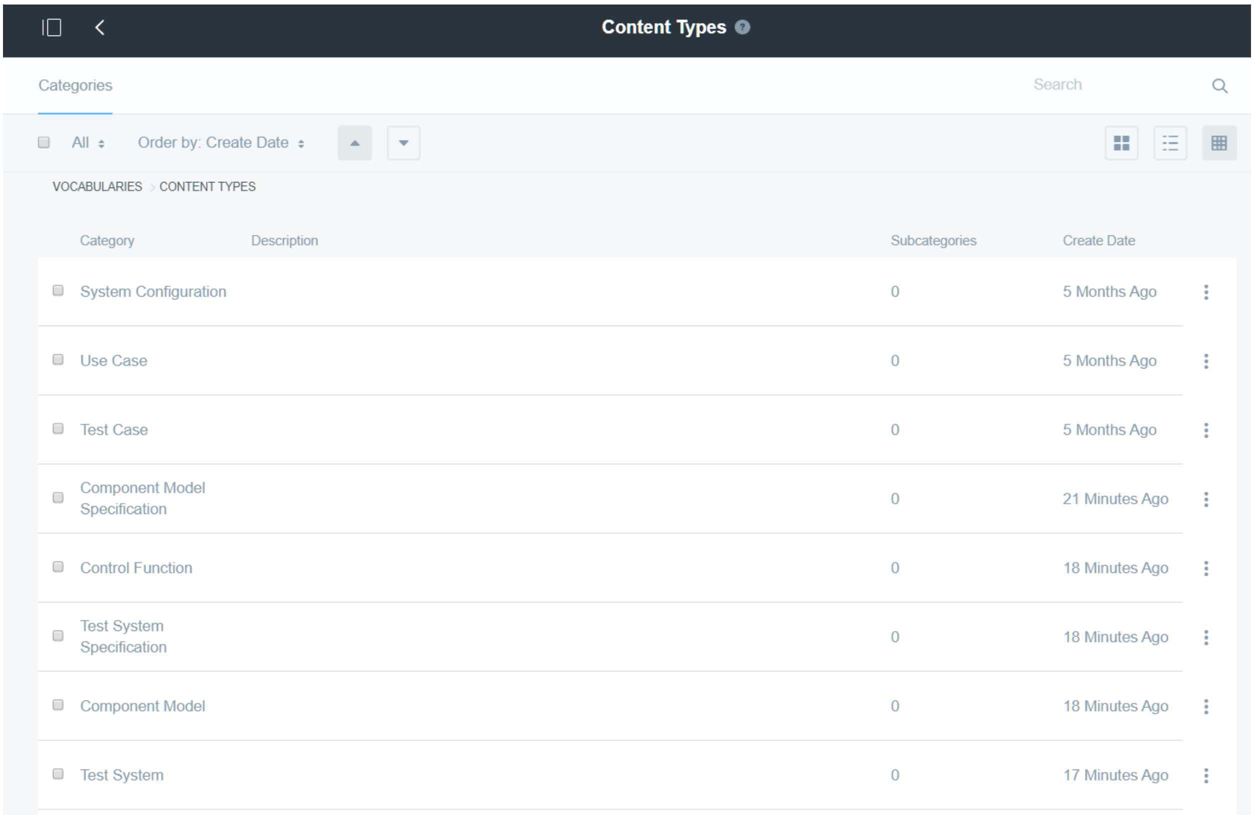

Figure 3: Categories of the classification category "Content Types"

Figure 3 shows exemplarily the editing screen for the “Content Types” classification. The definition of the content types is according to the semi-structured content types for SDIP content as defined in Deliverable 6.2 [2]. The editing forms for contents now allow setting the “Content Type” classification according to the type of content created. Another classification is defined by “Organisation”. It contains as categories short names of all organizations which create content for SDIP (e.g. AIT, KIT, EDF, VITO). This category allows filtering content by organization. The “Scope” classification is used to classify contents according to energy specific categories that classify contents according to their usage of storage technologies (“Electrical energy storage” and/or the aim of a certain use case (e.g. “Demand response”).

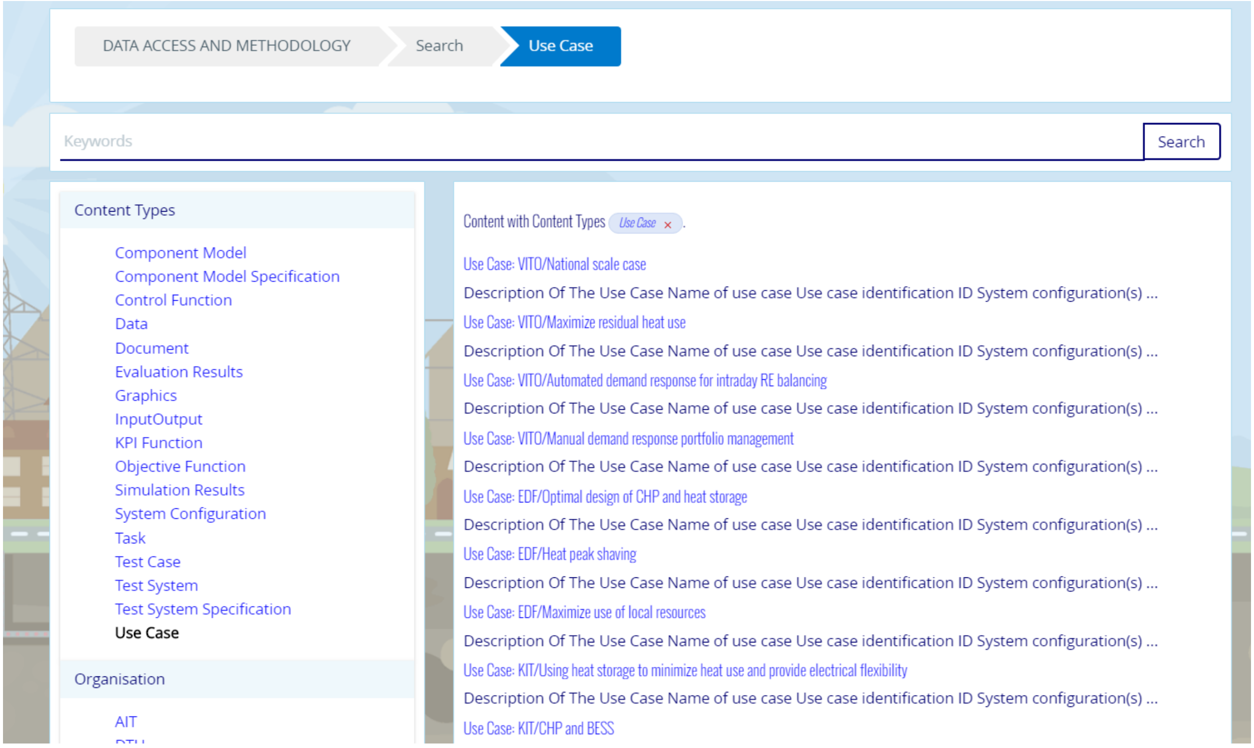

Figure 4: SDIP search filtered by the category "Use Case"

Figure 4 shows the search page of SDIP filtered by the “Content Types” category “Use Case”. On the left side of Figure 4, a column with the mentioned facets for restricting the search is displayed. As can be seen from the screenshot, the category “Use Case” was chosen by the user and is therefore highlighted. Because no other keywords were put into the search slot on top of the figure, the result box on the right side of the search page lists all “Use Case Descriptions” which are available in SDIP. Putting in some search keywords into the search box would filter the list of use cases even more according to the given keywords.

Clicking on the link of a search hit in this box will directly lead to a web page describing the use case in more detail.

The setup of the search page for SDIP was kept to be very simple. Performing some tests, it was find out that the mentioned facets are enough to quickly arrive at a wanted search result. Anyway, the Liferay / Elasticsearch interface provide much more functionalities for more advanced search configurations, if needed. E.g. authors can attach freely chosen “tag” keywords to their content which can be used for searching by using a “tag cloud”. The system allows it also to create more facets if needed. Facets could be either defined as further category classifications or Elasticsearch could even provide categories on the fly [4] by classifying certain fields of the underlying data structure (e.g. documents could be classified by authors if needed, or system models could be classified by type of implementation “e.g. co-simulation” or “monolithic”.

But typically the search semantics of these further classifications is already handled by the “search slot” on top of the page well, because search keywords like e.g. “co-simulation” or the name of an author will already provide filtering in a similar manner. Therefore, to not overdesign the search interface, only the few mentioned classifications are used at the moment.

4 Summary and Outlook

This document gives a short overview on the work performed in Task 6.4 for setting up and configuring the search infrastructure for SDIP. Liferay allows it to either use an internal or external Elasticsearch installation as the search engine for instrumenting search. Because of performance reasons, an external installation of Elasticsearch will be used for the SmILES search interface.

The SmILES portal uses an Elasticsearch cluster, which is part of the Energy Lab 2.0 infrastructure which also hosts the SmILES portal. For providing faceted search, several classification systems were added to the SmILES Liferay server. These classify Liferay contents according to the “Content Type”, “Organisation” and “Scope” of contents. Then, a search page for SDIP was setup, which uses these classifications for a facetted search (See Section 3).

The main goal of the SDIP search setup was to keep it very simple for the users. While testing the system, it turns out that the above mentioned facets together with a search slot for keywords already allow a very comfortable search for SDIP results. But anyway, the search infrastructure provides many more advanced features, which could be instrumented, if users aren’t satisfied with the current search interface. E.g. authors can attach freely chosen “tag” keywords to their content which can be used for searching by using a “tag cloud”. The system allows it also to create more facets if needed. Facets could be either defined as further category classifications, or Elasticsearch could even provide categories on the fly by classifying certain fields of the underlying data structure (e.g. documents could be classified by authors if needed, or system models could be classified by type of implementation e.g. “co-simulation” or “monolithic”. This gives many more options for improvement of the search interface in the future.

Bibliography

[1] Deliverable D6.1, "D6.1: First version of web portal which contains a 1st version of the Website," H2020 SmILES project, 2017.

[2] Deliverable D6.2, " D6.2: Specifications of data formats and metadata schemes for the data and information management platform", H2020 SmILES project, 2019.

[3] Liferay DXP7.1 Server Documentation Online, https://help.liferay.com/hc/en-us/categories/360000872531, visited last on 10/31/2019.

[4] Elasticsearch Documentation Online, https://www.elastic.co/guide/en/elasticsearch/reference/7.x/index.html, visited last on 10/31/2019.