



OKKAM – Enabling a Web Of Entities

Grant Agreement No. 215032

D7.2: Second OKKAM empowered tool suite

Document Number	D7.2
Document Title	Second OKKAM empowered tool suite
Version	3
Status	Final
Work Package	WP7
Deliverable Type	Report
Contractual Date of Delivery	14/05/2010
Actual Date of Delivery	
Responsible Unit	
Contributors	ExpertSystem, UNITN
Keyword List	
Dissemination level	PU

Change History

Version	Date	Status	Author (Company)	Description
1	15/02/2010	Draft	Daniele Cordioli (Expert System)	First draft, document skeleton
2	13/03/2010	Draft	Angela Fogarolli (UNITN)	Input UNITN
4	23/04/2010	Draft	Daniele Cordioli (Expert System)	Input Expert System
5	14/05/2010	Final	Francesco Danza (Expert System)	Approved

Executive Summary

In this document we provide a guide to the second OKKAM empowered tool suite.

The first part of this document describes the OKKAM empowered tool architecture. The objectives are to design and develop OKKAM architecture for supporting the creation of new *OKKAMized* content. In general, this will be achieved by designing and implementing plug-ins for popular content creation tools, like office applications, XML and ontology editors, etc.

In the second part we focus on every single OKKAM empowered tool developed during this second period.

Table of Contents

1. INTRODUCTION.....	6
1.1. WHAT IS OKKAM?.....	6
1.2. THE SCOPE OF THIS DOCUMENT	6
2. OKKAM EMPOWERED TOOL: ARCHITECTURE.....	7
2.1. 3-TIER ARCHITECTURE	7
2.2. THE OKKAMIZERS PIPELINE.....	9
3. SECOND OKKAM EMPOWERED TOOL SUITE.....	12
3.1. GMAIL	12
3.1.1. <i>Environment</i>	12
3.1.2. <i>What it does</i>	12
3.1.3. <i>Features</i>	13
3.1.4. <i>Technology</i>	14
3.1.5. <i>Installation</i>	15
3.2. YAHOO!MAIL	17
3.2.1. <i>Environment</i>	17
3.2.2. <i>What it does</i>	17
3.2.3. <i>Features</i>	18
3.2.4. <i>Technology</i>	19
3.2.5. <i>Installation</i>	20
3.3. HOTMAIL.....	22
3.3.1. <i>Environment</i>	22
3.3.2. <i>What it does</i>	22
3.3.3. <i>Features</i>	23
3.3.4. <i>Technology</i>	24
3.3.5. <i>Installation</i>	25
3.4. BLOGGER	27
3.4.1. <i>Environment</i>	27
3.4.2. <i>What it does</i>	27
3.4.3. <i>Features</i>	28
3.4.4. <i>Technology</i>	29
3.4.5. <i>Installation</i>	30
3.5. TINYMCE	32
3.5.1. <i>Environment</i>	32
3.5.2. <i>What it does</i>	32
3.5.3. <i>Features</i>	33
3.5.4. <i>Technology</i>	34
3.5.5. <i>Installation</i>	34
3.6. NEON.....	35
3.6.1. <i>Environment</i>	35
3.6.2. <i>Features</i>	35
3.6.3. <i>Technology</i>	35
3.7. OPENOFFICE.....	36
3.7.1. <i>Environment</i>	36
3.7.2. <i>Features</i>	36
3.7.3. <i>Technology</i>	36
4. DOWNLOAD	37
5. CONCLUSIONS AND NEXT STEP	38
6. REFERENCES.....	39
7. APPENDIX – INFORMATION ON THE COGITO TECHNOLOGY	40
7.1. SEMANTIC ANALYSIS	40
7.2. COGITO® SEMANTIC TECHNOLOGY.....	40

7.2.1. *Parser* 42
 7.2.2. *Lexicon*..... 42
 7.3. COGITO® DISCOVER..... 44

Glossary

OKKAM	IST 7th Framework Research Project
ENS	Entity Name System (developed by the OKKAM project)
API	Application Programming Interface
URL	Uniform Resource Locator
XML	Extensible Markup Language
RDF	Resource Description Framework
RDFa	Resource Description Framework in attributes
WSDL	Web Service Description Language
XUL	XML User Interface Language
Sensigrafo®	a semantic network is a representation of linguistic knowledge and world knowledge. It is an oriented graph consisting in tags representing concepts, and arcs representing the conceptual relationship between the concepts
COGITO®	Cogito is the pioneer semantic software developed by Expert System. It understands the meaning of words, like a person does when reading. That's why it's a unique technology.
NLP	Natural Language Processing is the extraction process of semantic information from human being expressions (spoken or written) by a PC. This process is particularly complex due to the natural difficulty of the language caused by ambiguity.
REST	is a style of software architecture for distributed hypermedia systems such as the World Wide Web.
OKKAMization	is the process of recognizing the relevant entities in a piece of data and creating an internal annotation which include its global, stable identifier.
OKKAMizer	is a tool which allows the OKKAMization of a specific type of content OKKAM-empowered tool: it is an application which is extended to interact with the ENS for bringing benefits to users based on the occurrence of identifiers in the content.

1. Introduction

1.1. What is OKKAM?

OKKAM is a Large-Scale Integrating Project funded by the European Commission under the 7th Framework Program (FP7), running until **June 2010**.

The **OKKAM** project aims at *enabling* the **Web of Entities**, namely a virtual space where any collection of data and information about any type of entities (e.g. people, locations, organizations, events, products, ...) published on the Web can be integrated into a single virtual, decentralized, open knowledge base (like the Web did for hypertexts.)

OKKAM will contribute to this vision by supporting the convergence towards the use of a *single and globally unique identifier* for any entity which is named on the Web. The intuition of the project is that the concrete realization of the Web of Entities requires us to enable tools and practices for cutting to the root the proliferation of unnecessary new identifiers for naming the entities which already have a public identifier (the *OKKAM's razor*). Therefore, **OKKAM** will make available to content creators, editors and developers a global infrastructure and a collection of new tools and plug-ins which support them, to easily find public identifiers for the entities named in their contents/services, use them for creating annotations, and build new network-based services which make essential use of these identifiers in an open environment (like the Web or large Intranets).

To realize this vision, **OKKAM** proposes the following roadmap:

- Providing a scalable and sustainable infrastructure, called the **Entity Name System (ENS)**, for making the systematic reuse of global and unique entity identifiers not only possible, but easy and straightforward. The ENS will be a distributed service which permanently stores identifiers for entities and provides a collection of core services (e.g. entity matching, ID mapping and resolution) needed to support their pervasive reuse;
- bootstrapping and enabling the fast growth of Web of Entities by fostering the creation of **OKKAMized content** (i.e. content where entities are named or annotated with OKKAM IDs) in **OKKAM-empowered applications** (i.e. applications which can interact with the ENS for getting and reusing identifiers);
- showcasing the benefits of enabling the Web of Entities and, more generally, of an entity-oriented approach to content and knowledge management by building **relevant applications** on top of the new infrastructure in three important areas: *information retrieval and semantic search*, *content authoring* (more specifically, in *scientific publishing* and *news production*) and *organizational knowledge management*.

More up-to-date information can be found on the OKKAM webpage: <http://www.okkam.org>.

1.2. The scope of this document

This deliverable gives an overview of the first OKKAM Empowered tools suite. First we describe the architecture and the APIs which are available for applications to consume the services of the ENS. The second part focuses on every OKKAM Empowered tool developed for this first suite.

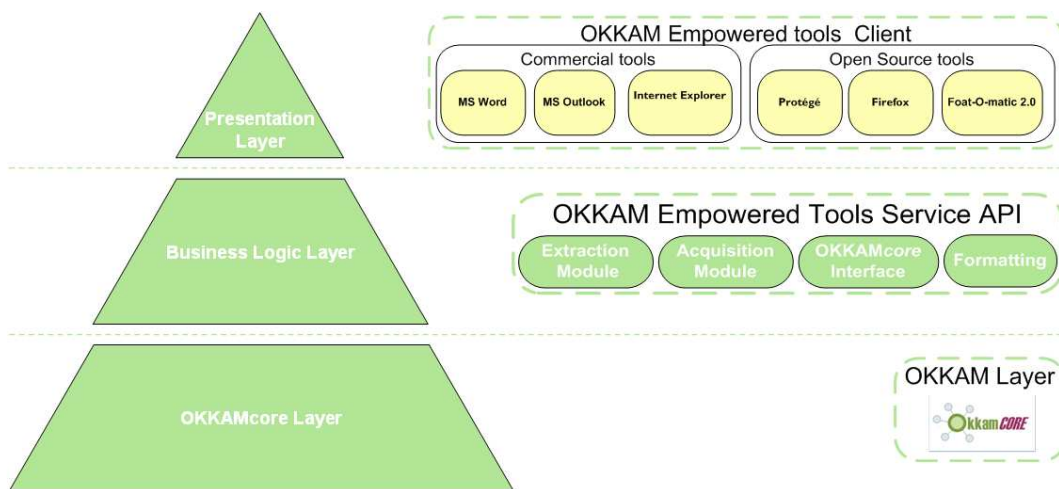
2. OKKAM Empowered tool: Architecture

The OKKAM empowered tools architecture is a typical three tier architecture. This is a client-server architecture in which the user interfaces, business logic, computer data storage and data access are developed and maintained as independent modules, on separate platforms.

In the next paragraph we describe the architecture and the workflow of process.

2.1. 3-tier Architecture

As we can see in the following picture, the system is based on a classic three tier architecture:



- **OKKAMcore Layer**

This layer contains the ENS services, which are available via Web Services. The ENS stores unique identifiers and offers services to applications in order to retrieve them. The information is then passed back to the logic layer for processing, and then eventually back to the user. A description of the Web Services API is publicly available at:

- <http://api.okkam.org>, which contains further examples as well.
- WSDL URL: <http://api.okkam.org/okkam-core/WebServices?wsdl>

- **Business Logic Layer**

This layer contains the Application services API. In particular, it coordinates the application, processes commands, and makes logical decisions and evaluations. The Application Services API automatically extracts entities from documents and Web resources; it also moves and processes data between the two surrounding layers.

The Application Services API allows the following operations starting from a document:

- Recognizing the named entities like People, Companies, Location, and other
- Recognizing the OKKAM entities
- Generating the OKKAM core queries extracting the named entities
- Enriching the document with the OKKAMid
- Enriching the document with the RDFa

WSDL URL:

<http://host.expertsystem.it/OkkamWebService/services/OkkamWebService?wsdl>

- **Presentation Layer**

This is the topmost level of the application. The presentation tier displays information related to such services for each tool, in commercial and open source environments. The first OKKAM empowered tool suite contains plug-ins for the following tools:

- Gmail
- Yahoo!mail
- Hotmail
- Blogger
- TinyMCE
- Neon
- Open Office

Using this approach we are able to create a set of OKKAM Empowered tools based on a shared platform.

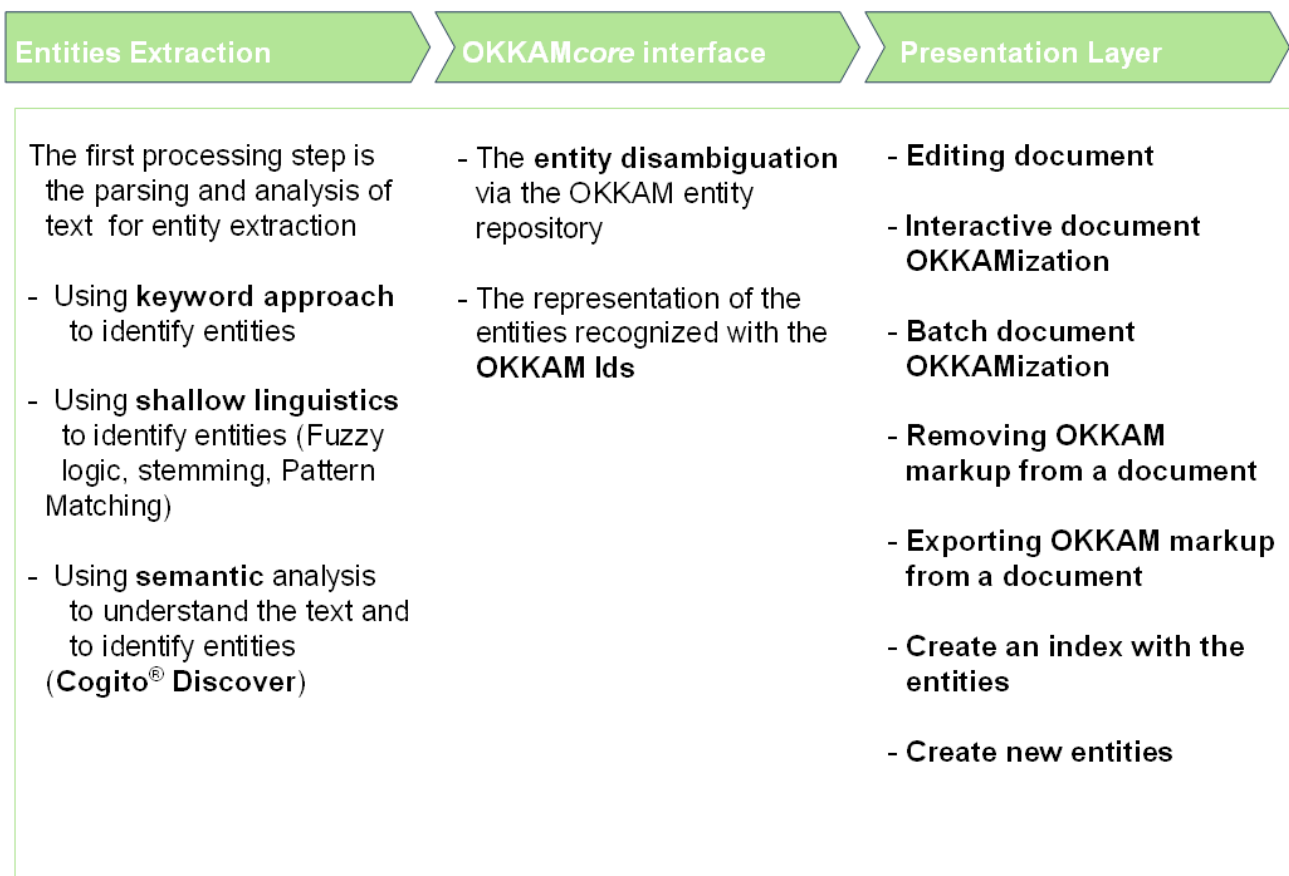
This means that:

- The solution allows integrating different tools in the architecture in a non-invasive way, because the connection is related only to the presentation layer. Changes, updates and new features involve only the presentation layer, not the logic layer.
- Using RESTful web service (simple web service implemented using HTTP and the principles of REST) or SOAP methodology the solution can be integrated with other different platforms.
- The system is able to manage different kind of documents: pdf, txt, html, xml, office.
- Using this architecture it is possible to provide integration capability based on standard technology like Java Message Service, Web Service, HTTP interface.

2.2. The OKKAMizers Pipeline

This paragraph focuses on the different steps of the OKKAMization process. In particular, we will analyze how the different tier moves and processes data, in order to better understand the OKKAM Empowered tool functionality.

The following picture describes the three main steps of this workflow:



- **Entities extraction**

It is clear that every tool analyzes different kinds of information. Depending on the situation, the gathering of data (ACQUISITION) can be carried out in various ways. After acquisition, the first processing step is the parsing and analysis of a text for entities extraction. In particular, the system contains three different approaches:

- Keyword based
- Shallow linguistic
- Semantic Analysis

The main module is the semantic analysis based on COGITO[®] Discover semantic capabilities, which provides Semantic Analysis for texts in English and Italian. Once acquired, data are cached and immediately analyzed using the semantic approach. Following the NPL, the whole text is disambiguated, where disambiguation is the process defining the meaning of one or more words in a given text, when words have distinct meanings. Disambiguation is solved through adequate algorithms. This process is based on the Sensigrafo[®], a semantic network that contains a representation of linguistic knowledge and world knowledge. It is an oriented graph consisting in tags representing concepts, and arcs representing the conceptual relationship between concepts.

See the end of the document for more information about the semantic approach.

The extractor used by the system is architecture independent, the pipeline currently contains not only COGITO[®] Discover, but also:

- the Sap extractor, in order to recognize Sap products
- Protein's extractor based on Keyword Approach

- **OKKAMcore Interface**

The Entity Matching stage generates an OKKAM ID for entities, where this is required. This process is based on the following steps:

- **Querying the system**
A query is created for each identified named entity (e.g. people, location). It contains the main features useful to better identify the potential entity in a unique way. The query is sent to the OKKAM Engine.
- **Query processing**
The OKKAM Engine analyzes the query, searches the repository and returns, if present, the correct OKKAM ID for the named entity. If the named entity is not in the repository, another task starts, creating a new persistent web identifier for entities which don't have one yet.
- **The correct OKKAM ID identification**
The OKKAM ID resulting from matching is a value ready to be included as a new field in the document.

The use of this module is the basis for building a Web of Entities, where information about the same entity is consolidated in such a way that aggregation, integration and mash up become easier and faster.

- **Presentation Layer**

At this stage, the system is able to perform different operations depending on the type of tool that is used. The main features are:

- Editing a document
- Interactive document OKKAMization
- Batch document OKKAMization
- Removing OKKAM mark up from a document
- Exporting OKKAM mark up from a document
- Creating an entities index
- Creating a new entity
- Searching for entities in the repository or in a document

In the next paragraph we will analyze the different tools developed.

3. Second OKKAM Empowered tool suite

The second OKKAM Empowered tool suite contains the tools described in the following paragraphs.

3.1. GMAIL

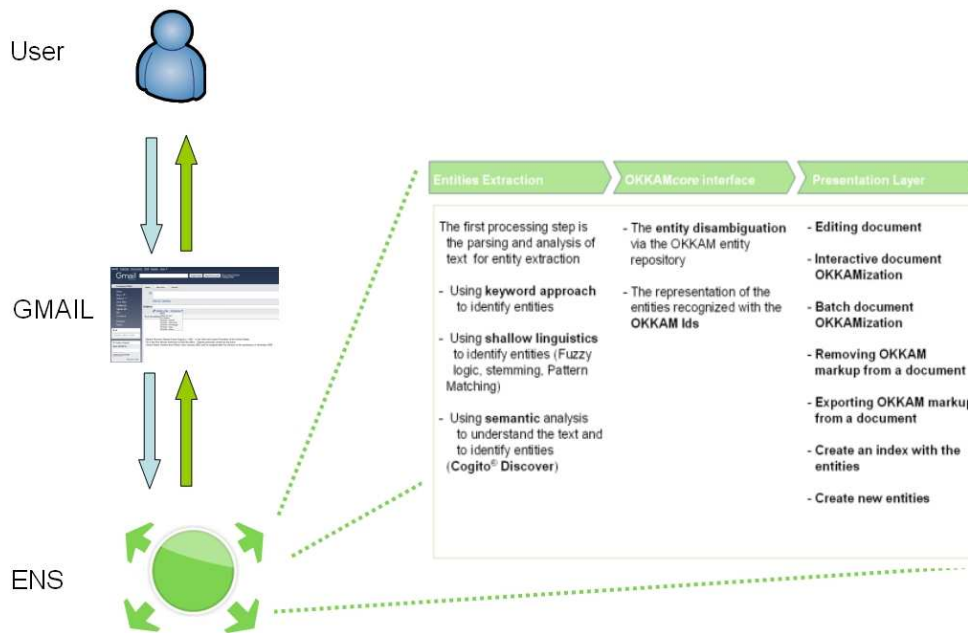
3.1.1. Environment

Gmail is the web interface for Google mail service which, after creating a free account, gives access to common operation as read incoming emails, compose emails and more

3.1.2. What it does

This paragraph focuses on the different steps of the **OKKAMization** process. In particular, we will analyze how OKKAM moves and processes data, in order to better understand the OKKAM Empowered tool functionality.

The following picture describes the three main steps of this workflow:



1. The user starts writing an email
2. The GMAIL OKKAM empowered tool takes the email’s body and sends it to the OKKAM System. To recognize the named entities inside the email it is fundamental to analyze the

email using the OKKAM technology. This technology is very complex, based on a semantic approach and is impossible to use on the client side, so using a server side approach is needed. Before starting, the process requires the user to view and accept the OKKAM privacy statement.

3. OKKAM starts to analyze the plain text coming from the email. The first step is the Entities extraction. It is clear that every tool analyzes different kinds of information. Depending on the situation, the gathering of data (ACQUISITION) can be carried out in various ways. After acquisition, the first processing step is the parsing and analysis of a text for entities extraction. In particular, the system contains three different approaches: Keyword based, Shallow linguistics, **Semantic Analysis**

The main module is the semantic analysis based on COGITO[®] Discover semantic capabilities, which provides Semantic Analysis for texts in English and Italian. A query is created for each identified named entity (e.g. people, location). It contains the main features useful to better identify the potential entity in a unique way. The query is sent to the OKKAM Engine. The Entity Matching stage generates an OKKAM ID for entities, where this is required. This process is based on the following steps:

1. **Query processing**

The OKKAM Engine analyzes the query, searches the repository and returns, if present, the correct OKKAM ID for the named entity.

2. **The correct OKKAM ID identification**

The OKKAM ID resulting from matching is a value ready to be included as a new field in the document.

The use of this module is the basis for building a Web of Entities, where information about the same entity is consolidated in such a way that aggregation, integration and mash up become easier and faster. The result you have is your email will be tagged with the OKKAMids.

3.1.3. Features

- **RDFa**

This function will write RDF fragment around words representing an entity.

- **Okkamize text**

Selecting this your email will be enhanced with a little report showing, for each entity inside your text, their Okkam identifier.

- **Full Okkam**

This will display three little widgets summarizing all the available information about entities inside your text. The one entitled 'Extraction' shows the name of the recognized entities and their semantic type, the 'Okkamize' one reports the information returned by OKKAM for each entity, and the 'Enrichment' one shows the text enriched with OKKAM semantic information.

- **Search**
Links to Okkam webpage where to search for entities.
- **Add entity**
Links to Okkam webpage where to add an entity. It requires registration.
<http://api.okkam.org/EnsWebToolKit>
- **Homepage**
Keeps links to OKKAM homepage and the pages where to search for and add entities.
- **Help**
If you need additional help, here you will find user's manual.
- **About**
Automatic upgrade detection. It keeps links to OKKAM feedback page to provide feedback.

3.1.4. Technology

This Rich Internet Application is created using:

- **Javascript**
JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.
- **Greasemonkey**
Greasemonkey is a Mozilla Firefox add-on that allows users to install scripts that make on-the-fly changes to HTML web page content on the DOMContentLoaded event, which happens immediately after it is loaded in the browser (also known as augmented browsing). As Greasemonkey scripts are persistent, the changes made to the web pages are executed every time the page is opened, making them effectively permanent for the user running the script. Greasemonkey can be used for adding new functions to web pages (for example, embedding price comparison in Amazon.com web pages), fixing rendering bugs, combining data from multiple webpages, and numerous other purposes.
- **jQuery**
jQuery is a lightweight cross-browser JavaScript library that emphasizes interaction between JavaScript and HTML. It was released in January 2006 at BarCamp NYC by John Resig. Used by over 27% of the 10,000 most visited websites, jQuery is the most popular JavaScript library in use today. jQuery is free, open source software, dual-licensed under the MIT License and the GNU General Public License, Version 2.[2] jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for

developers to create plugins on top of the JavaScript library. Providing this option, developers are able to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. This contributes to the creation of powerful and dynamic web pages.

- **IE7Pro**

IE7Pro is an add-on to Internet Explorer 6, 7 and 8 beta version that aims to enhance the feature set provided by the browser. IE7Pro adds features such as tab enhancement, ad blocker, flash blocker, mouse gestures, inline search, privacy enhancements, online bookmark service, Greasemonkey-like user script support, and plug-in support. IE7Pro is also known as IEPro, and is available in several languages – this is made possible by user translations.

3.1.5. Installation

This tool is available for Microsoft Internet Explorer and Mozilla Firefox

3.1.5.1 Mozilla Firefox

Installing an Okkam Empowered Tool for Firefox is pretty simple, just follow the steps explained:

1. Make sure you have the last version of Greasemonkey plugin for Firefox installed. If not just go to <https://addons.mozilla.org/en-US/firefox/search?q=Greasemonkey>, click on the ‘add to firefox button’ of the first result and follow the guided installation procedure.
2. Download the plug-in from the following link <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Once Greasemonkey is up and running, open Firefox and go to a blank tab (just press Ctrl+T).
5. Select the script file and drag it onto the blank tab. Click on the ‘Install’ button and check for the confirmation message appearing at the bottom right corner of Firefox window.
6. In some cases you could see the code of the plug-in in the blank tab. To install the script click on the install button shown in the right side of the yellow strip appearing at the top of the tab or select on the Firefox toolbar Tools > Greasemonkey menu > Install script.

3.1.5.2 Microsoft Internet Explorer

To install an Okkam Empowered Tool for Internet Explorer follow these steps:

1. Make sure you have installed IEPro addon for Internet Explorer. If not you can find it here: <http://www.brothersoft.com/download-ie7pro-54469.html> (or find the right link from the official site <http://www.ie7pro.com>).
2. Download the plug-in from the following link
<http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Copy the script file in the IEPRO ‘userscripts’ folder which probably looks like
c:\program files\IEPRO\userscripts.
5. Just to see if everything is ok, open IE, double click on the IEPRO icon at the right bottom corner of the browser window. In the user script submenu the checkbox labelled “Enable user script” and the one with the name of the script must be checked.

3.2. YAHOO!mail

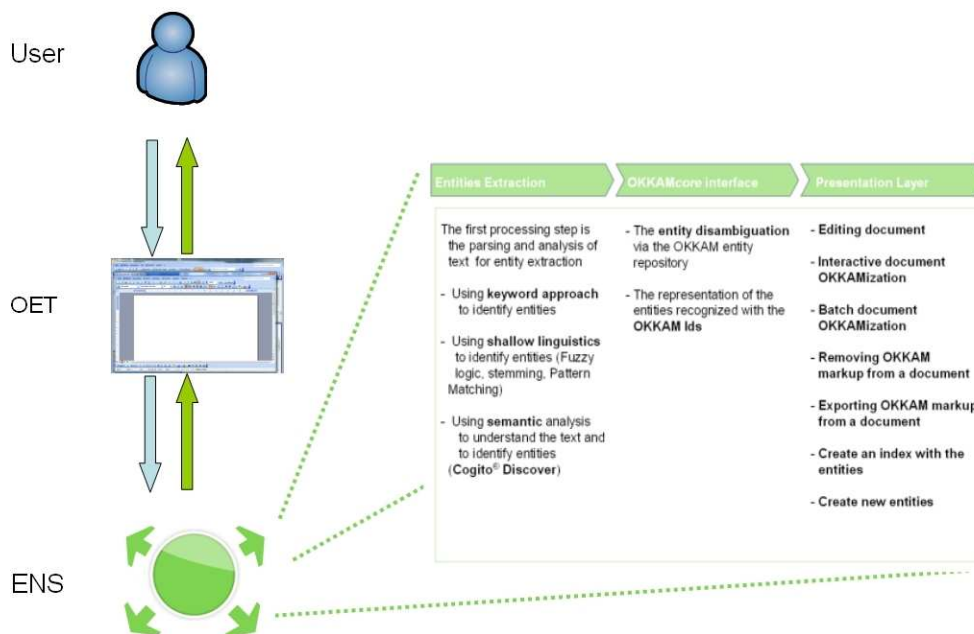
3.2.1. Environment

Yahoo! Mail is a free web mail service provided by Yahoo!. The Yahoo!Mail OKKAM button is a plug-in to OKKAM-enable the Yahoo!mail site.

3.2.2. What it does

This paragraph focuses on the different steps of the **OKKAMization** process. In particular, we will analyze how OKKAM moves and processes data, in order to better understand the OKKAM Empowered tool functionality.

The following picture describes the three main steps of this workflow:



1. The user starts writing an email
2. The YAHOO!mail OKKAM empowered tool takes the email's body and sends it to the OKKAM System. To recognize the named entities inside the email it is fundamental to analyze the email using the OKKAM technology. This technology is very complex, based on a semantic approach and is impossible to use on the client side, so using a server side approach is needed. Before starting, the process requires the user to view and accept the OKKAM privacy statement.

3. OKKAM starts to analyze the plain text coming from the email. The first step is the Entities extraction. It is clear that every tool analyzes different kinds of information. Depending on the situation, the gathering of data (ACQUISITION) can be carried out in various ways. After acquisition, the first processing step is the parsing and analysis of a text for entities extraction. In particular, the system contains three different approaches: Keyword based, Shallow linguistics, **Semantic Analysis**

The main module is the semantic analysis based on COGITO[®] Discover semantic capabilities, which provides Semantic Analysis for texts in English and Italian. A query is created for each identified named entity (e.g. people, location). It contains the main features useful to better identify the potential entity in a unique way. The query is sent to the OKKAM Engine. The Entity Matching stage generates an OKKAM ID for entities, where this is required. This process is based on the following steps:

1. **Query processing**

The OKKAM Engine analyzes the query, searches the repository and returns, if present, the correct OKKAM ID for the named entity.

2. **The correct OKKAM ID identification**

The OKKAM ID resulting from matching is a value ready to be included as a new field in the document.

The use of this module is the basis for building a Web of Entities, where information about the same entity is consolidated in such a way that aggregation, integration and mash up become easier and faster. The result you have is your email will be tagged with the OKKAMids.

3.2.3. Features

- **RDFa**
This function will write RDF fragment around words representing an entity.
- **Okkamize text**
Selecting this your email will be enhanced with a little report showing, for each entity inside your text, their Okkam identifier.
- **Full Okkam**
This will display three little widgets summarizing all the available information about entities inside your text. The one entitled 'Extraction' shows the name of the recognized entities and their semantic type, the 'Okkamize' one reports the information returned by OKKAM for each entity, and the 'Enrichment' one shows the text enriched with OKKAM semantic information.
- **Search**
Links to Okkam webpage where to search for entities.
- **Add entity**
Links to Okkam webpage where to add an entity. It requires registration.
<http://api.okkam.org/EnsWebToolKit>

- **Homepage**
Keeps links to OKKAM homepage and the pages where to search for and add entities.
- **Help**
If you need additional help, here you will find user's manual.
- **About**
Automatic upgrade detection. It keeps links to OKKAM feedback page to provide feedback.

3.2.4. Technology

This Rich Internet Application is created using:

- **Javascript**
JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.
- **Greasemonkey**
Greasemonkey is a Mozilla Firefox add-on that allows users to install scripts that make on-the-fly changes to HTML web page content on the DOMContentLoaded event, which happens immediately after it is loaded in the browser (also known as augmented browsing). As Greasemonkey scripts are persistent, the changes made to the web pages are executed every time the page is opened, making them effectively permanent for the user running the script. Greasemonkey can be used for adding new functions to web pages (for example, embedding price comparison in Amazon.com web pages), fixing rendering bugs, combining data from multiple webpages, and numerous other purposes.
- **jQuery**
jQuery is a lightweight cross-browser JavaScript library that emphasizes interaction between JavaScript and HTML. It was released in January 2006 at BarCamp NYC by John Resig. Used by over 27% of the 10,000 most visited websites, jQuery is the most popular JavaScript library in use today. jQuery is free, open source software, dual-licensed under the MIT License and the GNU General Public License, Version 2.[2] jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plugins on top of the JavaScript library. Providing this option, developers are able to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. This contributes to the creation of powerful and dynamic web pages.

- **IE7Pro**

IE7Pro is an add-on to Internet Explorer 6, 7 and 8 beta version that aims to enhance the feature set provided by the browser. IE7Pro adds features such as tab enhancement, ad blocker, flash blocker, mouse gestures, inline search, privacy enhancements, online bookmark service, Greasemonkey-like user script support, and plug-in support. IE7Pro is also known as IEPro, and is available in several languages – this is made possible by user translations.

3.2.5. Installation

This tool is available for Microsoft Internet Explorer and Mozilla Firefox

3.2.5.1 Mozilla Firefox

Installing an Okkam Empowered Tool for Firefox is pretty simple, just follow the steps explained:

1. Make sure you have the last version of Greasemonkey plugin for Firefox installed. If not just go to <https://addons.mozilla.org/en-US/firefox/search?q=Greasemonkey>, click on the ‘add to firefox button’ of the first result and follow the guided installation procedure.
2. Download the plug-in from the following link <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Once Greasemonkey is up and running, open Firefox and go to a blank tab (just press Ctrl+T).
5. Select the script file and drag it onto the blank tab. Click on the ‘Install’ button and check for the confirmation message appearing at the bottom right corner of Firefox window.
6. In some cases you could see the code of the plug-in in the blank tab.
To install the script click on the install button shown in the right side of the yellow strip appearing at the top of the tab or select on the Firefox toolbar Tools > Greasemonkey menu > Install script.

3.2.5.2 Microsoft Internet Explorer

To install an Okkam Empowered Tool for Internet Explorer follow these steps:

1. Make sure you have installed IEPro addon for Internet Explorer. If not you can find it here: <http://www.brothersoft.com/download-ie7pro-54469.html> (or find the right link from the official site <http://www.ie7pro.com>).
2. Download the plug-in from the following link
<http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Copy the script file in the IEPRO ‘userscripts’ folder which probably looks like
c:\program files\IEPRO\userscripts.
5. Just to see if everything is ok, open IE, double click on the IEPRO icon at the right bottom corner of the browser window. In the user script submenu the checkbox labelled “Enable user script” and the one with the name of the script must be checked.

3.3. HOTMAIL

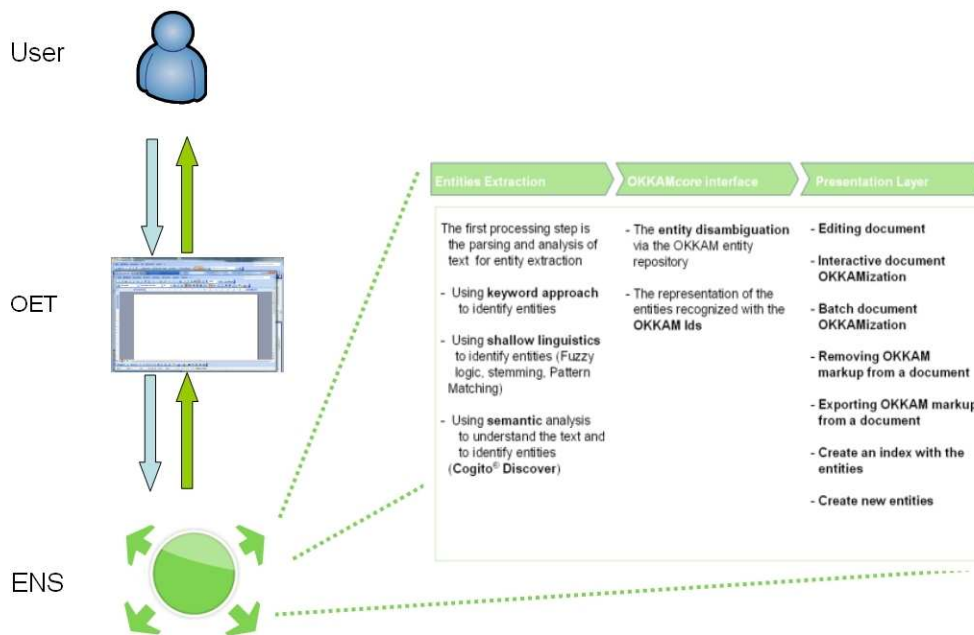
3.3.1. Environment

Windows Live Hotmail, formerly known as MSN Hotmail and commonly referred to simply as Hotmail, is a free web-based email service operated by Microsoft as part of its Windows Live group. The Hotmail OKKAM button is a plug-in to OKKAM-enable the Hotmail site.

3.3.2. What it does

This paragraph focuses on the different steps of the **OKKAMization** process. In particular, we will analyze how OKKAM moves and processes data, in order to better understand the OKKAM Empowered tool functionality.

The following picture describes the three main steps of this workflow:



1. The user starts writing an email
2. The Hotmail OKKAM empowered tool takes the email's body and sends it to the OKKAM System. To recognize the named entities inside the email it is fundamental to analyze the email using the OKKAM technology. This technology is very complex, based on a semantic approach and is impossible to use on the client side, so using a server side approach is needed. Before starting, the process requires the user to view and accept the OKKAM privacy statement.

3. OKKAM starts to analyze the plain text coming from the email. The first step is the Entities extraction. It is clear that every tool analyzes different kinds of information. Depending on the situation, the gathering of data (ACQUISITION) can be carried out in various ways. After acquisition, the first processing step is the parsing and analysis of a text for entities extraction. In particular, the system contains three different approaches: Keyword based, Shallow linguistics, **Semantic Analysis**

The main module is the semantic analysis based on COGITO[®] Discover semantic capabilities, which provides Semantic Analysis for texts in English and Italian. A query is created for each identified named entity (e.g. people, location). It contains the main features useful to better identify the potential entity in a unique way. The query is sent to the OKKAM Engine. The Entity Matching stage generates an OKKAM ID for entities, where this is required. This process is based on the following steps:

1. **Query processing**

The OKKAM Engine analyzes the query, searches the repository and returns, if present, the correct OKKAM ID for the named entity.

2. **The correct OKKAM ID identification**

The OKKAM ID resulting from matching is a value ready to be included as a new field in the document.

The use of this module is the basis for building a Web of Entities, where information about the same entity is consolidated in such a way that aggregation, integration and mash up become easier and faster. The result you have is your email will be tagged with the OKKAMids.

3.3.3. Features

- **RDFa**

This function will write RDF fragment around words representing an entity.

- **Okkamize text**

Selecting this your email will be enhanced with a little report showing, for each entity inside your text, their Okkam identifier.

- **Full Okkam**

This will display three little widgets summarizing all the available information about entities inside your text. The one entitled 'Extraction' shows the name of the recognized entities and their semantic type, the 'Okkamize' one reports the information returned by OKKAM for each entity, and the 'Enrichment' one shows the text enriched with OKKAM semantic information.

- **Search**

Links to Okkam webpage where to search for entities.

- **Add entity**

Links to Okkam webpage where to add an entity. It requires registration.

<http://api.okkam.org/EnsWebToolKit>

- **Homepage**
Keeps links to OKKAM homepage and the pages where to search for and add entities.
- **Help**
If you need additional help, here you will find user's manual.
- **About**
Automatic upgrade detection. It keeps links to OKKAM feedback page to provide feedback.

3.3.4. Technology

This Rich Internet Application is created using:

- **Javascript**
JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.
- **Greasemonkey**
Greasemonkey is a Mozilla Firefox add-on that allows users to install scripts that make on-the-fly changes to HTML web page content on the DOMContentLoaded event, which happens immediately after it is loaded in the browser (also known as augmented browsing). As Greasemonkey scripts are persistent, the changes made to the web pages are executed every time the page is opened, making them effectively permanent for the user running the script. Greasemonkey can be used for adding new functions to web pages (for example, embedding price comparison in Amazon.com web pages), fixing rendering bugs, combining data from multiple webpages, and numerous other purposes.
- **jQuery**
jQuery is a lightweight cross-browser JavaScript library that emphasizes interaction between JavaScript and HTML. It was released in January 2006 at BarCamp NYC by John Resig. Used by over 27% of the 10,000 most visited websites, jQuery is the most popular JavaScript library in use today. jQuery is free, open source software, dual-licensed under the MIT License and the GNU General Public License, Version 2.[2] jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plugins on top of the JavaScript library. Providing this option, developers are able to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. This contributes to the creation of powerful and dynamic web pages.

- **IE7Pro**

IE7Pro is an add-on to Internet Explorer 6, 7 and 8 beta version that aims to enhance the feature set provided by the browser. IE7Pro adds features such as tab enhancement, ad blocker, flash blocker, mouse gestures, inline search, privacy enhancements, online bookmark service, Greasemonkey-like user script support, and plug-in support. IE7Pro is also known as IEPro, and is available in several languages – this is made possible by user translations.

3.3.5. Installation

This tool is available for Microsoft Internet Explorer and Mozilla Firefox

3.3.5.1 Mozilla Firefox

Installing an Okkam Empowered Tool for Firefox is pretty simple, just follow the steps explained:

1. Make sure you have the last version of Greasemonkey plugin for Firefox installed. If not just go to <https://addons.mozilla.org/en-US/firefox/search?q=Greasemonkey>, click on the ‘add to firefox button’ of the first result and follow the guided installation procedure.
2. Download the plug-in from the following link <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Once Greasemonkey is up and running, open Firefox and go to a blank tab (just press Ctrl+T).
5. Select the script file and drag it onto the blank tab. Click on the ‘Install’ button and check for the confirmation message appearing at the bottom right corner of Firefox window.
6. In some cases you could see the code of the plug-in in the blank tab.
To install the script click on the install button shown in the right side of the yellow strip appearing at the top of the tab or select on the Firefox toolbar Tools > Greasemonkey menu > Install script.

3.3.5.2 Microsoft Internet Explorer

To install an Okkam Empowered Tool for Internet Explorer follow these steps:

1. Make sure you have installed IEPro addon for Internet Explorer. If not you can find it here: <http://www.brothersoft.com/download-ie7pro-54469.html> (or find the right link from the official site <http://www.ie7pro.com>).
2. Download the plug-in from the following link
<http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Copy the script file in the IEPRO ‘userscripts’ folder which probably looks like
c:\program files\IEPRO\userscripts.
5. Just to see if everything is ok, open IE, double click on the IEPRO icon at the right bottom corner of the browser window. In the user script submenu the checkbox labelled “Enable user script” and the one with the name of the script must be checked.

3.4. Blogger

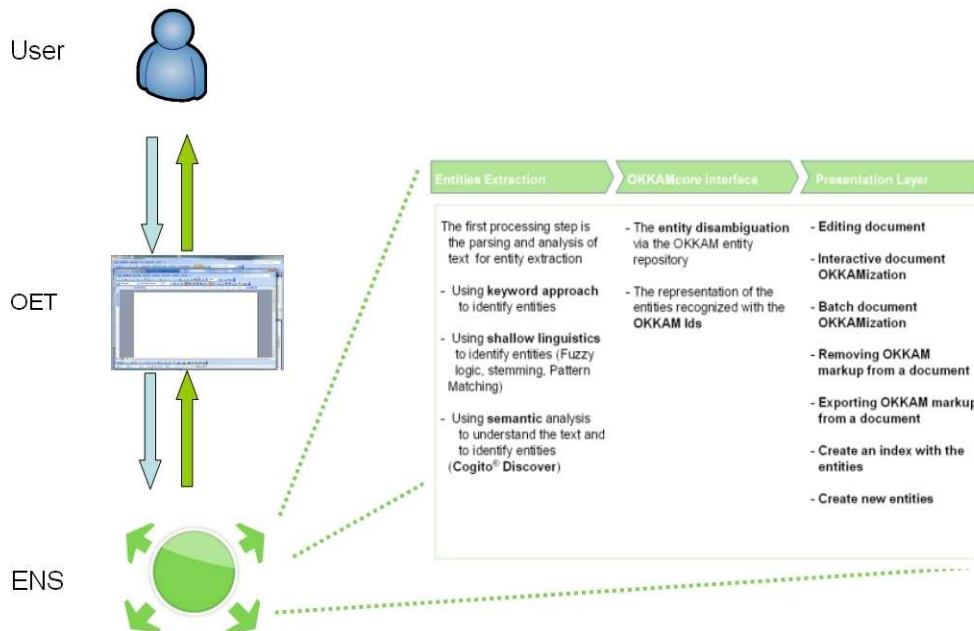
3.4.1. Environment

Blogger is a free blog publishing tool from Google for easily sharing your thoughts with the world.. The Blogger OKKAM button is a plug-in to OKKAM-enable the Blogger site.

3.4.2. What it does

This paragraph focuses on the different steps of the **OKKAMization** process. In particular, we will analyze how OKKAM moves and processes data, in order to better understand the OKKAM Empowered tool functionality.

The following picture describes the three main steps of this workflow:



1. The user starts writing the post
2. The Blogger OKKAM empowered tool takes the blog's post and sends it to the OKKAM System. To recognize the named entities inside the email it is fundamental to analyze the email using the OKKAM technology. This technology is very complex, based on a semantic approach and is impossible to use on the client side, so using a server side approach is needed. Before starting, the process requires the user to view and accept the OKKAM privacy statement.

3. OKKAM starts to analyze the plain text coming from the post. The first step is the Entities extraction. It is clear that every tool analyzes different kinds of information. Depending on the situation, the gathering of data (ACQUISITION) can be carried out in various ways. After acquisition, the first processing step is the parsing and analysis of a text for entities extraction. In particular, the system contains three different approaches: Keyword based, Shallow linguistics, **Semantic Analysis**

The main module is the semantic analysis based on COGITO[®] Discover semantic capabilities, which provides Semantic Analysis for texts in English and Italian. A query is created for each identified named entity (e.g. people, location). It contains the main features useful to better identify the potential entity in a unique way. The query is sent to the OKKAM Engine. The Entity Matching stage generates an OKKAM ID for entities, where this is required. This process is based on the following steps:

1. **Query processing**

The OKKAM Engine analyzes the query, searches the repository and returns, if present, the correct OKKAM ID for the named entity.

2. **The correct OKKAM ID identification**

The OKKAM ID resulting from matching is a value ready to be included as a new field in the document.

The use of this module is the basis for building a Web of Entities, where information about the same entity is consolidated in such a way that aggregation, integration and mash up become easier and faster. The result you have is your email will be tagged with the OKKAMids.

3.4.3. Features

- **RDFa**
This function will write RDF fragment around words representing an entity.
- **Okkamize text**
Selecting this your email will be enhanced with a little report showing, for each entity inside your text, their Okkam identifier.
- **Full Okkam**
This will display three little widgets summarizing all the available information about entities inside your text. The one entitled 'Extraction' shows the name of the recognized entities and their semantic type, the 'Okkamize' one reports the information returned by OKKAM for each entity, and the 'Enrichment' one shows the text enriched with OKKAM semantic information.
- **Search**
Links to Okkam webpage where to search for entities.
- **Add entity**
Links to Okkam webpage where to add an entity. It requires registration.
<http://api.okkam.org/EnsWebToolKit>

- **Homepage**
Keeps links to OKKAM homepage and the pages where to search for and add entities.
- **Help**
If you need additional help, here you will find user's manual.
- **About**
Automatic upgrade detection. It keeps links to OKKAM feedback page to provide feedback.

3.4.4. Technology

This Rich Internet Application is created using:

- **Javascript**
JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.
- **Greasemonkey**
Greasemonkey is a Mozilla Firefox add-on that allows users to install scripts that make on-the-fly changes to HTML web page content on the DOMContentLoaded event, which happens immediately after it is loaded in the browser (also known as augmented browsing). As Greasemonkey scripts are persistent, the changes made to the web pages are executed every time the page is opened, making them effectively permanent for the user running the script. Greasemonkey can be used for adding new functions to web pages (for example, embedding price comparison in Amazon.com web pages), fixing rendering bugs, combining data from multiple webpages, and numerous other purposes.
- **jQuery**
jQuery is a lightweight cross-browser JavaScript library that emphasizes interaction between JavaScript and HTML. It was released in January 2006 at BarCamp NYC by John Resig. Used by over 27% of the 10,000 most visited websites, jQuery is the most popular JavaScript library in use today. jQuery is free, open source software, dual-licensed under the MIT License and the GNU General Public License, Version 2.[2] jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plugins on top of the JavaScript library. Providing this option, developers are able to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. This contributes to the creation of powerful and dynamic web pages.

- **IE7Pro**

IE7Pro is an add-on to Internet Explorer 6, 7 and 8 beta version that aims to enhance the feature set provided by the browser. IE7Pro adds features such as tab enhancement, ad blocker, flash blocker, mouse gestures, inline search, privacy enhancements, online bookmark service, Greasemonkey-like user script support, and plug-in support. IE7Pro is also known as IEPro, and is available in several languages – this is made possible by user translations.

3.4.5. Installation

This tool is available for Microsoft Internet Explorer and Mozilla Firefox

3.4.5.1 Mozilla Firefox

Installing an Okkam Empowered Tool for Firefox is pretty simple, just follow the steps explained:

1. Make sure you have the last version of Greasemonkey plugin for Firefox installed. If not just go to <https://addons.mozilla.org/en-US/firefox/search?q=Greasemonkey>, click on the ‘add to firefox button’ of the first result and follow the guided installation procedure.
2. Download the plug-in from the following link <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Once Greasemonkey is up and running, open Firefox and go to a blank tab (just press Ctrl+T).
5. Select the script file and drag it onto the blank tab. Click on the ‘Install’ button and check for the confirmation message appearing at the bottom right corner of Firefox window.
6. In some cases you could see the code of the plug-in in the blank tab.
To install the script click on the install button shown in the right side of the yellow strip appearing at the top of the tab or select on the Firefox toolbar Tools > Greasemonkey menu > Install script.

3.4.5.2 Microsoft Internet Explorer

To install an Okkam Empowered Tool for Internet Explorer follow these steps:

1. Make sure you have installed IEPro addon for Internet Explorer. If not you can find it here: <http://www.brothersoft.com/download-ie7pro-54469.html> (or find the right link from the official site <http://www.ie7pro.com>).
2. Download the plug-in from the following link
<http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
3. Unzip the zip file, keeping the javascript and css file in the same folder.
4. Copy the script file in the IEPRO ‘userscripts’ folder which probably looks like
c:\program files\IEPRO\userscripts.
5. Just to see if everything is ok, open IE, double click on the IEPRO icon at the right bottom corner of the browser window. In the user script submenu the checkbox labelled “Enable user script” and the one with the name of the script must be checked.

3.5. TinyMCE

3.5.1. Environment

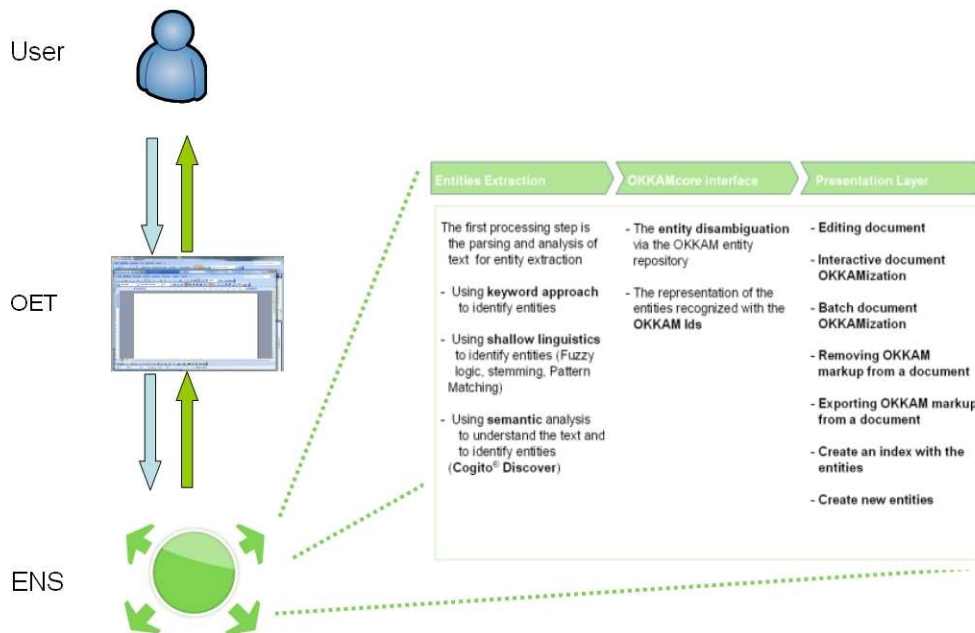
TinyMCE, also known as the Tiny Moxiecode Content Editor, is a platform-independent web-based JavaScript/HTML WYSIWYG editor control, released as open source software under the LGPL by Moxiecode Systems AB. It has the ability to convert HTML textarea fields or other HTML elements to editor instances. TinyMCE is designed to easily integrate with content management systems.

TinyMCE integrates with many different open source systems, such as Mambo, Joomla!, Drupal, Plone, WordPress, b2evolution, e107, phpWebSite and Freedomeditor.

3.5.2. What it does

This paragraph focuses on the different steps of the **OKKAMization** process. In particular, we will analyze how OKKAM moves and processes data, in order to better understand the OKKAM Empowered tool functionality.

The following picture describes the three main steps of this workflow:



1. The user starts writing the post
2. The TinyMCE OKKAM empowered tool takes the post and sends it to the OKKAM System. To recognize the named entities inside the email it is fundamental to analyze the email using the OKKAM technology. This technology is very complex, based on a semantic

approach and is impossible to use on the client side, so using a server side approach is needed. Before starting, the process requires the user to view and accept the OKKAM privacy statement.

3. OKKAM starts to analyze the plain text coming from the tool. The first step is the Entities extraction. It is clear that every tool analyzes different kinds of information. Depending on the situation, the gathering of data (ACQUISITION) can be carried out in various ways. After acquisition, the first processing step is the parsing and analysis of a text for entities extraction. In particular, the system contains three different approaches: Keyword based, Shallow linguistics, **Semantic Analysis**

The main module is the semantic analysis based on COGITO[®] Discover semantic capabilities, which provides Semantic Analysis for texts in English and Italian. A query is created for each identified named entity (e.g. people, location). It contains the main features useful to better identify the potential entity in a unique way. The query is sent to the OKKAM Engine. The Entity Matching stage generates an OKKAM ID for entities, where this is required. This process is based on the following steps:

1. **Query processing**

The OKKAM Engine analyzes the query, searches the repository and returns, if present, the correct OKKAM ID for the named entity.

2. **The correct OKKAM ID identification**

The OKKAM ID resulting from matching is a value ready to be included as a new field in the document.

The use of this module is the basis for building a Web of Entities, where information about the same entity is consolidated in such a way that aggregation, integration and mash up become easier and faster. The result you have is your email will be tagged with the OKKAMids.

3.5.3. Features

- **RDFa**
This function will write RDF fragment around words representing an entity.
- **Okkamize text**
Selecting this your email will be enhanced with a little report showing, for each entity inside your text, their Okkam identifier.
- **Search**
Links to Okkam webpage where to search for entities.
- **Add entity**
Links to Okkam webpage where to add an entity. It requires registration.
<http://api.okkam.org/EnsWebToolKit>
- **Homepage**
Keeps links to OKKAM homepage and the pages where to search for and add entities.

- **Help**
If you need additional help, here you will find user's manual.

3.5.4. Technology

This Rich Internet Application is created using:

- **Javascript**
JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.
- **jQuery**
jQuery is a lightweight cross-browser JavaScript library that emphasizes interaction between JavaScript and HTML. It was released in January 2006 at BarCamp NYC by John Resig. Used by over 27% of the 10,000 most visited websites, jQuery is the most popular JavaScript library in use today. jQuery is free, open source software, dual-licensed under the MIT License and the GNU General Public License, Version 2.[2] jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plugins on top of the JavaScript library. Providing this option, developers are able to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. This contributes to the creation of powerful and dynamic web pages.

3.5.5. Installation

This tool is available for Tiny MCE editor.

This distribution provides you just the plugin folder.

All you have to do is to copy this folder in the tinyMCE plugin directory on your server and properly configure the tinyMCE instance as explained below.

1. Make sure you have installed Tiny MCE on your server. If not you can find it here: <http://tinymce.moxiecode.com/download.php> along with all the explanations about how to install and configure the editor.
2. Download the plug-in from the following link <http://community.okkam.org/index.php/Downloads/>
3. Copy the entire folder in the 'plugins' folder which probably looks like "tinymce/jscripts/tiny_mce/plugins".

4. Then all you have to do is to integrate the editor in your web page. There just a few lines of code to write: here is a basic example which shows a blank page with just one editor loading only the okkam plugin.
 1. In the head section load the TinyMCE main script and initialize it. The “advanced” theme is the only one that allows custom plugin, the textareas mode converts all the textarea HTML tags into editor instances, then are specified the plugin name and the button name:

```
<script type="text/javascript" src="{path to tiny_mce.js}"></script>
<script type="text/javascript">tinyMCE.init({ theme : "advanced", mode :
"textareas", plugins : "okkam", theme_advanced_buttons3 :
"okkambutton" });</script>
```
 2. Insert an editor instance in your page writing something like:

```
<textarea rows="15" cols="50" name="content">This is some content that will be
editable with TinyMCE.</textarea>
```

3.6. NEON

3.6.1. Environment

The NeOn toolkit is a state-of-the-art, open source multi-platform ontology engineering environment,. Okkam plugin for NeOn (Okkam4N) can reuse the global unique identifier (Okkam ID) for individuals in OWL/RDF knowledge bases.

3.6.2. Features

- **directly integrated in the NeOn Toolkit interface**
- **easily add Okkam IDs to instances**
- **detailed adaptability of search properties**

3.6.3. Technology

- This Rich Internet Application is created using the NeOn plugin capability.

3.7. OPENOFFICE

3.7.1. Environment

OpenOffice.org is the leading open-source office software suite for word processing, spreadsheets, presentations, graphics, databases and more. It is available in many languages and works on all common computers.

The plug-in can be accessed by means of a OpenOffice Word-integrated toolbar.

3.7.2. Features

- **Interactive document OKKAMization**
Users can analyze part of a text just selecting it in advance
- **Batch document OKKAMization**
Users can analyze a text in background mode
- **Interaction with OKKAM**
Users can select the correct OKKAM entity

3.7.3. Technology

- Javascript integrates OKKAM functionalities directly in your document

4. Download

All the OKKAM Empowered tool suite is available at this web site: <http://community.okkam.org/>

Documentation, user manuals, OKKAM Empowered tool are available for each tools:

- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-gmail.html>
- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-yahoomail.html>
- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-hotmail.html>
- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-blogger.html>
- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-tiny-mce.html>
- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam4n.html>
- <http://community.okkam.org/index.php/Downloads/ENS-enabled-Tools/okkam-for-openoffice.html>

5. Conclusions and next step

This deliverable contains a guide to the public OKKAM Empowered tool developed for this second suite. The guide reflects the functionality of the APIs as of April 2010 (the contractual date of delivery of this document). As the software evolves, we will consider revising the individual chapters at a later stage of the project.

6. References

- [1] <http://www.icefaces.org/main/home/>
- [2] <http://java.sun.com/javaee/javaserverfaces/>
- [3] <https://facelets.dev.java.net/>
- [4] <http://jena.sourceforge.net/>
- [5] <http://www.ie7pro.com/>
- [6] <http://www.greasespot.net/>

7. Appendix – Information on the COGITO Technology

7.1. Semantic Analysis

COGITO® is a software system that “understands” the language in way that is very similar to the way people do. It collects all the structural and lexical text aspects in order to understand the meaning.

COGITO® is the result of hundreds of man years of research and development., from the research and extraction, to the analysis, classification and transformation of unstructured information,

COGITO® is the most advanced technology available on the market because it overcomes the more traditional approaches to the automatic processing of natural language by leveraging semantic comprehension.

COGITO® is powered by Sensigrafo®, a comprehensive semantic network available in different languages enabling disambiguation of terms, the secret behind Semantic Intelligence.

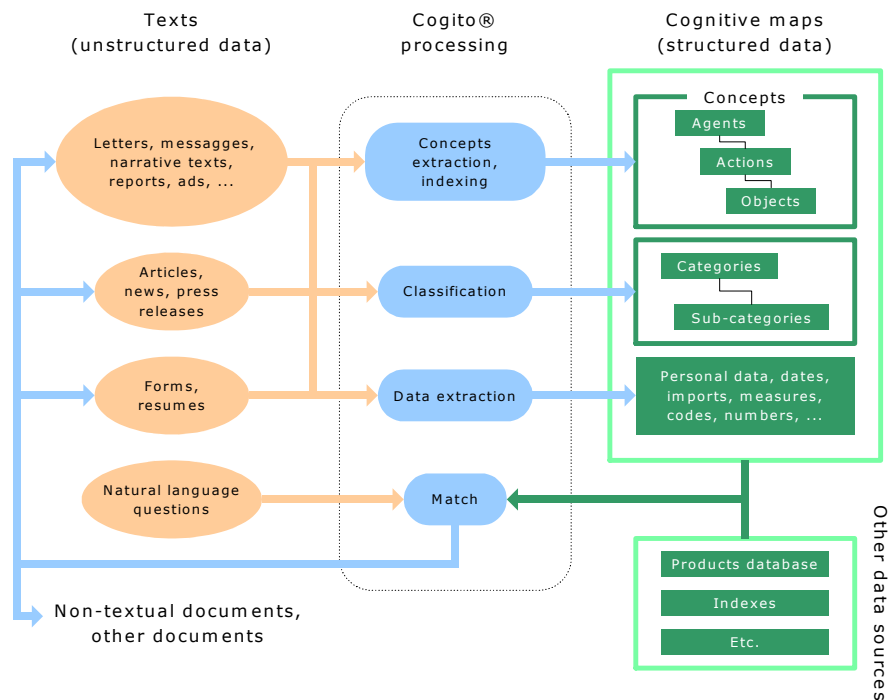
- more than 3,500,000 connections (regarding contests, constructions, subjects and domains, idioms, phraseologies...)
- more than 600,000 semantic concepts
- more than 400,000 hyponymy and hyperonymy connections
- more than 55,000 hyperonymy and troponymy connections
- more than 370,000 connections for the corpus and tens of thousands of links among subjects, objects, meronyms, etc.

7.2. COGITO® Semantic Technology

COGITO® represents our solution. It is the result achieved after 10 years of work in the development of state-of-the-art linguistic technologies.

COGITO® is a software system that “understands” the language in way that is very similar to the way people do. It collects all the structural and lexical text aspects in order to understand the meaning.

COGITO® processing result is a cognitive and conceptual map, i.e. a structured representation of qualifying aspects of incoming unstructured data. The output structuring allows the automatic processing of the most relevant elements of the text.



Rival systems have a superficial processing level. For example, they completely ignore the so-called stop words, i.e. words that do not have a proper meaning (like articles and prepositions), when indeed they are fundamental in forming the meaning of sentences.

On the other hand, COGITO®, with its linguistic analysis capabilities and semantic knowledge, takes into account every word; in the same way people do when they read, listen to and write.

In order to better understand this feature, consider these sentences:

- (a) I've been flying my bird of prey.
- (b) I've been flying my bird as prey.

The meaning of these two sentences is completely different, because of the prepositions. In the sentence (a) the words “bird” and “prey” refer to a raptorial bird - a type of bird, such as an eagle, falcon or hawk, which hunts other birds and small animals - while in the sentence (b) “bird” and “prey” refer to the use of a bird as a decoy, or lure, to facilitate the capture of another animal.

This difference is considered to be clear for a person, but this doesn't happen with current text management programs: from their point of view the two sentences are the same.

COGITO® can detect the difference and distinguish between the two concepts, allowing a conceptual search.

COGITO® has an open and scalable infrastructure, which can be easily customized in order to manage any type of contents: technical documentation, encyclopedic material, financial information, newspaper articles and so on.

The power and the level of integration between its components makes COGITO® a revolutionary technology and very unique in the marketplace. Its main components are:

- parser: it carries out morphological, grammatical and syntactical analysis of the sentence
- lexicon: it recognizes words and all their meanings
- memory: it keeps trace of previous analysis outcomes
- knowledge: a representation of real world knowledge
- content representation: text contents in the form of a conceptual map

7.2.1. Parser

The first step to take in order to understand the meaning of a sentence is to determine the grammar role of each word. For example, in these sentences:

- (a) He aged 40 years.
- (b) The wine is aged.

The word “aged” appears with two different grammar types: in sentence (a) the word is an adjective, while in sentence (b) it is a verb.

According to traditional systems the two words are the same, while the linguistic technologies assign different meaning to them.

Recognizing a word independently of its written form is equally important; nouns and verbs have several forms:

- (a) Marcello Mastroianni was the most popular Italian actor abroad.
- (b) Today it is difficult that no more young actresses play a protagonist role.

In the sentences above, two forms (“actor” and “actresses”) expressing the same concept are used. COGITO® individuates gender - masculine/feminine - and number - singular/plural – to recognize both words as forms of “acting person” associating all of them to their common meanings correctly, instead of individuating different words as other systems do.

COGITO® parser manages the grammar characteristics of the sentence completely and optimally. It carries out the logical analysis, providing a solid base for contents processing.

7.2.2. Lexicon

The information of all the possible meanings of words is fundamental in order to process text contents with high precision.

Being unable to detect different meaning brings a misleading understanding of the phrase. Consider this example:

- (a) The car driver was injured.
- (b) I used the long driver.
- (c) The driver was installed in the computer.

The word “driver” is ambiguous, because its meaning depends on context. In order to carry out word meaning disambiguation, COGITO® looks up its lexicon to find all possible meanings. These lexicons are semantic networks.

Semantic networks are not plain dictionaries, but resources that have been optimized for programming use, where word forms are knots linked to each other by multiple links denoting semantic or lexical relations. For example, the knots “secret agent” and “spy” are linked by a semantic relation named “synonymy” (they have similar meaning), while “angel” and “devil” are linked by “antonymy” (they indicate opposite concepts).

Something impossible for all those systems considering words as simple character strings, not concepts.

Sensigrafo® is an Expert System innovative semantic network. At a high level it has similarities with other existing semantic networks (like WordNet for English language from Princeton University), in that it captures the structure of a language through the definition of concepts and relationships between concepts. However, Sensigrafo® is uniquely designed from the start, implemented and optimized for automatic language processing.

Concepts (like nouns, verbs, adjectives and adverbs) are defined by “synsets”, which are lists of cognitive synonyms, interlinked through semantic and lexical relations.

Some definitions may be useful:

Lemma – A pairing of a particular orthographic form with some form of symbolic meaning representation.

Semantic relations – The relation between meanings of lemmas, such as homonymy, homophones, homographs, synonymy, hyponymy, hypernymy and polysemy.

Syncon – A syncon is a list of lemmas with the same meaning. It represents the semantic relation of synonymy between lemmas.

Sensigrafo® embodies some unique features that allow a superior semantic analysis and disambiguation of text. No identified deep text analysis solution today has these features and is able to achieve similar levels of performance. Among the unique features of Sensigrafo® are:

- Expanded definition of lemmas and semantic relations
- Categories of attributes
- Domains of usage
- Differentiated sets of attributes
- Small footprint of 50MB of memory

7.3. COGITO® Discover

The problem of information management has reached relevant proportions. The intellectual capital (know how, brands, copyrights, patents, projects) and web resources cause a constant increase in the amount of corporate data everyday. Instead of becoming a great opportunity for growth and development, this ironically becomes the main obstacle to the activities of strategy definition, analysis, control, and project management. This happens because quantity and quality often do not necessarily go hand in hand, but above all because a large part of the data is unstructured: more than 80% of the information on which the companies base their business is in fact generally in text form. The management of this vast “sea” of information (reports, in-depth sector analyses, research, meeting proceedings, analyses and balance sheets, articles, press releases, etc.), filed in different places and formats is really complicated. Selecting useful material can be difficult. Often nothing is found, documents already present are duplicated to the detriment of the quality of the results. In addition, the risk of selecting wrong information while ignoring important details or, even worse, not knowing all the data required to make the best possible decision, can result in missing goals and wasting time and money.

COGITO® Discover is capable of extracting relevant information and knowledge from the mass of data, thus making the entire corporate knowledge wealth always available: expertise, best practices, internal and external information, regardless of it being available in structured or unstructured form. The system enables the user to select the data, individuating it in an autonomous way and extracting it automatically from the available documents generally stored in different formats, or available in more than one database, already structured or semi-structured.

Unlike other tools for the automated extraction of data, COGITO® Discover relies on linguistic and semantic technologies to analyze and comprehend the meaning of texts. The ETL process (Extraction, Transformation, Loading) carried out with the linguistic intelligence of COGITO® Discover is, in fact, focused on the normalization of contents. The use of semantics in the management of content permits the tracing back to a univocal form (or “normalizing”, actually) the most simple data such as prices, dates, quantities, measures, etc.,. It also extracts and standardizes non-numerical information such as acronyms, abbreviations, names, and alternative forms to express the same concept.

The combination of the module for the extraction of concepts and of that for the normalization, guarantees a correct disambiguation of the terms (that is, the attribution of the right meaning to the words of the sentence as related to the context) and at the same time an automated selection of the data which is already normalized.

COGITO® Discover can therefore find, in any unstructured text, elements marked as interesting: names, companies or agencies, products and places, in addition to dates, prices, as well as numbers in general. Furthermore, always relying on the evolved linguistic technology of Expert System, COGITO® Discover is able to “understand” the subjects of texts, tracing and isolating automatically the relevant concepts contained in the sentences.