Deliverable D6.2.1

Early Prototype

Editor:	Esteban Garcia-Cuesta, iSOCO
Author(s):	Esteban García-Cuesta (iSOCO), Fátima Galán (iSOCO), Andrej Muhic (JSI), Mitja Trampus (JSI), Zhixing Li (THU), Xavier Carreras (UPC)
Deliverable Nature:	Prototype (P)
Dissemination Level: (Confidentiality)[1]	Public (PU)
Contractual Delivery Date:	M12
Actual Delivery Date:	M12
Suggested Readers:	All partners of the XLike project consortium and end-users
Version:	1.0
Keywords:	Demo, prototype, end-users, cross-lingual, dissemination, validation, entity tracking, article tracking.

[1] Please indicate the dissemination level using one of the following codes:
• **PU =** Public • **PP =** Restricted to other programme participants (including the Commission Services) • **RE =** Restricted to a group specified by the consortium (including the Commission Services) • **CO =** Confidential, only for members of the consortium (including the Commission Services) • **Restreint UE =** Classified with the classification level "Restreint UE" according to Commission Decision 2001/844 and amendments • **Confidentiel UE =** Classified with the mention of the classification level "Confidentiel UE" according to Commission Decision 2001/844 and amendments • **Secret UE =** Classified with the mention of the classification level "Secret UE" according to Commission Decision 2001/844 and amendments

Disclaimer

Full Project Title:	Cross-lingual Knowledge Extraction
Short Project Title:	XLike
Number and Title of Work package:	WP6 – Integration and Toolkit
Document Title:	D6.2.1 – Early Prototype
Editor (Name, Affiliation)	Esteban Garcia-Cuesta iSOCO
Work package Leader (Name, affiliation)	Esteban García-Cuesta, iSOCO
Estimation of PM spent on the deliverable:	5 PM

Executive Summary

This document presents the early prototype which implements the needed architecture specifications for the integration of the different modules provided by WP1, WP2, WP3, WP4 and WP5. The work done at each one of these modules is described in the corresponding technical deliverables: D2.1.1, D2.2.1, D2.3.1, D3.1.1, D4.1.1, and D5.2.1 which are also part of the first year work of the XLike project.

The main goal of this document, which is also the main goal of the WP6 work package, is to show the overall integration of all the components in order to obtain a first prototype of the Xlike project and to accomplish with the requirements of the use cases which have been defined in D1.2.1 (STA and Bloomberg).

This document is also partly based in the deliverable D6.1.1 “Early Toolkit architecture specification” which is a preliminary work that defined the overall strategy to be followed in order to develop the first prototype. There are also two other previous documents which are much related with the development of this first prototype which are D1.1.1 and D1.2.1. The first contains the set of available components for the project, some of them have been used towards the completion of this first prototype, and the second defines the requirements needed for accomplishing with the STA and Bloomberg use cases.

This report covers a description of the prototype developed so far including: a description of the prototype and the methodology used for its implementation (Section 1), the overall description of the architecture (Section2), a description of the current architecture specifications (Section 3), the prototype description guided by the uses cases demonstrations including the XLike toolkit (Section 3), the definition of the APIs and the services deployed (Annex A and Annex B), and the data transformation including inputs and outputs at different stages of the pipeline (Annex C).

This document is also the first of three (D6.1.1 Y1, D6.2.1 Y2, and D6.2.3 Y3) which are associated with the prototypes development at the different stages of the project. These different stages of the project will collect incrementally the ongoing work and the improvements obtained after solving the problems detected at each previous prototype.

Table of Contents

List of Figures

List of Tables

Abbreviations

API	**A**pplication **P**rogramming **I**nterface
D	**D**eliverable
NLP	**N**atural **L**anguage **P**rocessing
SOA	**S**ervice **O**riented **A**rchitecture
T	**T**ask
UI	**U**ser **I**nterface
URL	**U**niform **R**esource **L**ocator
REST	**Re**presentational **S**tate **T**ransfer
XML	e**X**tensible **M**arkup **L**anguage
XLike	**Cross**-**li**ngual **K**nowledge **E**xtraction
WP	**W**ork **P**ackage
WS	**W**eb **S**ervice

Definitions

Pipeline Refers to the flux of different processes which are applied to a set of raw data in order to analyze it and interpret it. In XLike project It covers the following phases: gathering data, pre-processing data, application of Natural Language Processing Tools, semantic interpretation, visualization, and finally domain interpretation

Hackathon Is an event in which computer programmers and others in the field of software development, like graphic designers, interface designers, project managers and computational philologists, collaborate intensively on software projects[2].

[2] http://en.wikipedia.org/wiki/Hackathon

1 Introduction

1.1 Integration overview and current status

The XLike early prototype aims to provide a first complete pipeline of the project with the main goal of allowing the easy interaction and upgrades of the different components of the project. At this stage (Y1 prototype) the main focus of the work is not to provide a very robust and final prototype which would be the final goal of the project (although robustness, reliability, and security[3] have been taken into account) but to provide a first global application that includes all the different components (WP1-WP5) that have been developed so far and also to satisfy the requirements of the use cases defined by the end-users (D1.2.1).

To achieve these goals two different pipelines have been implemented in parallel throughout the Y1 of the XLike project. The first one (Figure 1: Technological Demo Pipeline) has been focused on early testing and public accessibility to the technological advances of the project and it is publically allocated at the Sandbox platform which has been deployed at iSOCO[4]. The second one is the prototype itself (Figure 1: Prototyping Pipeline) which contains all the different components developed during the first year to accomplish with the STA and Bloomberg use cases[5]. The technological demo contains all the basic functionality that has been used for the prototype in order to achieve the use case requirements. Therefore the main difference between both of them (technological demo and prototype) is that the technological demo shows the updates of the developed functionalities of the project as soon as they are available with the main purpose of early testing whereas the prototype uses/adapts these functionalities in order to provide the needed data for accomplishing with the defined use cases and the visualization requirements.

These differences can be observed in the Figure 1 where the prototype is focused on the visualization and the use cases (including some analytics and visualization enhancements for improving the usability), but the demo is focused on showing the capabilities of the technology developed showing the raw results of the already deployed functionalities.

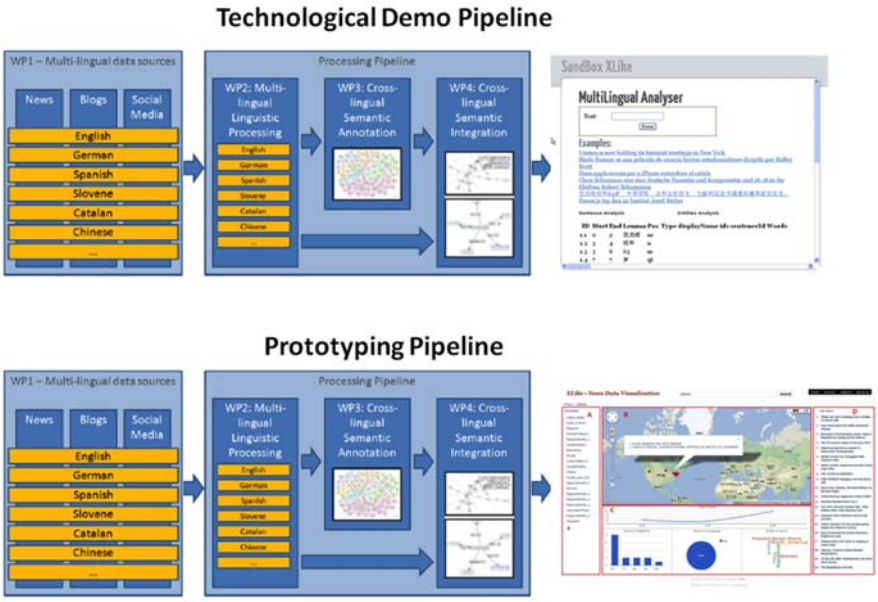

Figure 1 Pipelines implemented in the XLike project

[3] Specially for STA and Bloomberg use cases due to data privacy restrictions
[4] http://sandbox-xlike.isoco.com/demo/
[5] http://sandbox-xlike.isoco.com/portal/

The prototype has been created following the methodology described in the deliverable D6.1.1 allowing the co-design of the different parts of the project between all the partners. This has permitted a dynamic development of the different interfaces and an easier integration of the overall pipeline in order to produce this first prototype.

The overall used strategy for the integration has been a service oriented architecture (SOA) providing as many services as functionalities or resources needed. Although, having a central control point as a middleware platform for calling the different services was proposed at deliverable D6.1.1. it turned out that in order to pursue the specific defined use cases requirements it was easier to follow a data-centric approach where the data flows throughout the different steps/services of the pipeline at the same time that it gets richer from an information point of view. Therefore the services which are later in the pipeline make use of the previous ones of the pipeline in an incremental data enrichment approach. This approach is very appealing because there is no need to maintain a central control point which sometimes can be expensive and it allows the rapid prototyping making calls from some components to the others as needed.

We have to highlight that though this approach has many advantages it has been needed some extra-effort in order to provide on time and standardized all the different services/functionalities from previous packages to be used on later ones (see Annex C for a better understanding of the data enrichment process throughout the pipeline).

1.2 Methodology

The followed methodology was described at deliverable D6.1.1 and a co-evolutionary development model has been applied in order to change the implementation iteratively and according to the use cases needs. For software engineering viability a division of functionality has been done by developing services independently and providing a common data structure which is updated by using these different developed services. This early prototype (M12) if the first milestone where some major changes were expected relating with the architecture and functionalities and this document is the result of them. The next major expected changes are on M15 when a revision of the toolkit architecture specification has to be done (D6.1.2).

Regarding the development process each partner has been organized internally and according to the WPs in order to build a work-team to provide the services associated to the functionalities of their assigned tasks. At general project integration level a bi-weekly (until M6) and weekly video-conference meetings (from M6 to M12) have been scheduled in order to provide a feedback of the current ongoing work at the different components of the project and to allow the early standardization of data formats and services as well. These meetings have been used also to help any partner to provide the services needed or to assist them at any integration problem task.

These meetings have been complemented with some sprints (SCRUM methodology) whenever a quick development was needed to pursue some specific requirements (e.g. visualization integration or common data enrichment). Skype calls/chat and mail have been used for communication during the sprints. These sprints have been based mainly on:

- Requirements specification
- Development
- Results
- Quick Review

In order to help an easy final integration for obtaining the Y1 prototype two global hackathons have been also scheduled[6]. The main goal of a hackathon is to develop software collaboratively in order to fill the software gaps and also to provide a general overview of the project to the different people involve at the different component development.

For inter-task work a more informal communication protocol has been used via direct mailing or skype/gotoometing chat/calls.

1.3 Authorship

This document results from the work of the Xlike developers/researchers at the different work packages of the project. All the work done has been supported by iSOCO which has been acting as glue between all the partners and also has helped to define and standardize the different services and the provided data format at the different parts of the project. Relating the creation of the web services each partner (JSI, KIT, UZG, THU, and UPC) has been leading its own development and has followed the established standards in order to be compliant with the overall project strategy and to fulfil the use-cases requirements.

1.4 Relation with Other Work Packages

As introduced above, the Sandbox includes both, services and software components that are developed within the XLike project, as well as selected software relevant to the development of our reference implementation.

Regarding the prototype it includes the visualization for HCI jointly with the different service calls needed to obtain the relevant data according to the use cases.

Taking into account both of the above presented descriptions, the dependence between the different software components of the project is low (as was expected and proposed for making the integration easier at D6.1.1 by using a SOA architecture) and the interaction is mostly by functionality for accessing to the different steps of the pipeline or to the already stored and previously analyzed data.

The provided functionalities by the different partners can be split into the following layers according to the processing pipeline: i) source data, ii) analysis and interpretation, iii) application and interface as shown in the Figure 2.

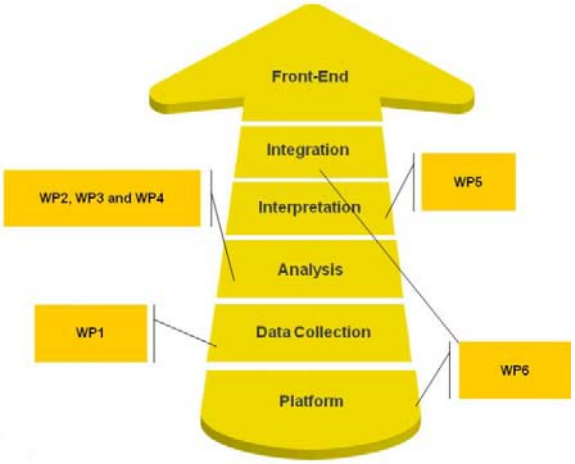

Figure 2 Integration of the different work packages into the XLike pipeline

[6] November 5-6th at Ljubljana, and November 28-29th at Madrid

It can be seen that the dependences at the different WPs are ordered sequentially which provides a loose coupling between them and a one to one dependency from an overall perspective. An example of these pipeline dependencies can be consulted in the previously delivered document D.61.1 at Section 1.2

In the following the overall architecture of the XLike prototype is described at Section 2. Afterwards the current specification and the XLike Toolkit is provided at Section 3 showing also the prototype itself as the combinations of all the components, and finally some conclusions are presented.

2 Overall XLike Architecture

2.1 Introduction

In this section the overall architecture and the interaction between the different components of the prototype is reviewed and explained. Following the four layers interpretation described at D6.1.1 (Section 2) the different developed components are classified in one of these layers.

- **Physical layer**: includes the platform that supports the execution of the different algorithms for the crawling, analysis of multilingual and cross-lingual level, and the posterior visualization. It can be monolithic or distributed as is explained later on the document. This layer has been provided by hosting the developed services by the actual owners (each one of the partners of the XLike project) or by deploying them in the **Sandbox**[7]. The Sandbox is also currently providing the access to the technological demo and also to the prototype via an Apache server.
- **Acquisition**: it contains the **crawling** processes which gather the data from the web, repositories, or any other data provider. This has been provided by using the **JSI NewsFeed** [8]. A deeper description of this component is described in D1.3.1 and what it does is to crawl thousands of RSS feeds in order to provide enough data for testing the use cases.
- **Analysis and interpretation**: it performs the **natural language processing** and any other analysis needed (e.g. semantic enrichment) for the posterior interpretation and application to specific domains (e.g. entity tracking). This layer is being covered by the different services deployed at WP2-WP3 and WP4 (e.g. lemmatization) and partially also with the analytics that are provided with the visualization.
- **Interface or Human-Computer interaction layer**: this allows the interaction between the end-users and XLike functionalities. This is provided mainly via the **visualization** as result of the WP5 work.

It is worth to highlight that the next three characteristics have been pursued as general rules of the architecture: **loose coupling and autonomy** (each one of the components developed have been treated independently), **flexibility and interoperability** (there are different languages which have been used in the project e.g. java and C++ and the use of services has helped for its easier integration), **reusability and format contract** (the different components can be easily deployed in other platforms e.g. Freeling is deployed at iSOCO's servers, and other components as cross-lingual similarity can be reused for different purposes e.g. find similarity between any two documents not only Wikipedia ones).

The early prototype includes the implemented functionalities of the different work packages for accomplishing with the use cases defined at D1.2.1. The main goals for this use cases can be summarized in:

- Entity Tracking (Bloomberg)
- Related Relevant Articles (Bloomberg)
- Article Tracking (STA)
- Topic and Entity Tracking (STA).

[7] http://sandbox-xlike.isoco.com
[8] http://newsfeed.ijs.si/

2.2 Architecture Principles

The overall architecture implementation and the integration of the different functionalities needed have been done by using a SOA (Service Oriented Approach) architecture and more specifically by using a RESTful Web Services approach[9]. This approach has been done by following a decentralized approach (see Figure 3) allowing that any service can call the others. This mode of integrating the different components relies on a very stable data description due to it is necessary to parse the outputs of the different used services in order to obtain the needed information. Addressing this main drawback an incremental data enriching approach has been adopted and, by following the order indicated in the pipeline, the data is incrementally enriched but without having any need of changing or deleting the previous already added information.

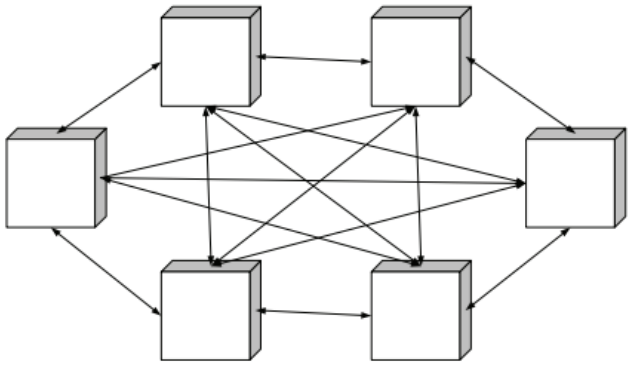

Figure 3 REST, HTTP + XML point to point communications

Currently the pipeline of calls is the following[10]:

1. Service for collecting and indexing pieces of news: it is provided by JSI newsfeed http://newsfeed.ijs.si/xlike/
2. Service for language identification: http://sandbox-xlike.isoco.com/services/language_code/ident
3. Service for shallow language processing for the different language: http://sandbox-xlike.isoco.com/services/analysis_XX/analyze (being XX the language identifier coded following the ISO 639-2 specification)
4. Service for annotating the document with cross-lingual links and wiki links: http://km.aifb.kit.edu/services/annotation-XX (being XX the language identifier coded following the ISO 639-2 specification),

and the data enrichment is the following according to those three calls:

1. It collects the different pieces of news from the different sources (e.g. STA), index and stores them into a local repository. A set of web services APIs are provided for accessing to this data.
2. Provides the language identifier for the given article. This information is used for calling the corresponding NLP services and also to store that information into the article XML file defining the original language of the document analyzed.

[9] The set of developed services can be consulted at Annex A.
[10] Note: although the logical sequence is the presented, actually due to implementation reasons the calls are being executed following this order 1→3→2 allowing to the last service of the pipeline to gaining control over the others

3. These services provide the information related with the shallow processing of the document. This includes the word itself, the lemmatization of a word, the POS tag, position of the word inside the sentence and starting and ending position characters of the word.
4. The service includes the annotations to the document which relates the document with other cross-lingual ones (e.g. other directly related Wikipedia articles or similar ones). These relations can be based on topic similarity or on semantic relatedness between documents. These annotations include a list of descriptions with the URLs associated to the document and an additional parameter for language identification.

The final schema and an example of the current data format obtained at the end of pipeline can be found at Annex C. The result of the integration is shown in the Figure 5 where the different parts of the interface provide the functionalities desired as is described in the D5.2.1.

2.3 Technological Demo

The Sandbox was introduced and specified in the deliverable D6.1.1 Early Toolkit architecture (Section 1 and Section3). This Sandbox has played an important role so far due to it has been a place for early testing of the different components which have been used later for the developed prototype as described in this document.

It has provided a testing environment, a sort of playground, for users and developers so that they can experience and interact with developed Xlike technologies and related software as early as possible during the development process. Moreover, it is the infrastructure that will be used to integrate existing technologies so that the rest of WPs can perform tests, and where the final implementation will be deployed.

Furthermore due to its easy integration properties and therefore rapid development, the Sandbox has been also used to provide the technological demo. This public demo was not one of the main goals of the Sandbox but it turned out to be very useful for early integration and public awareness of the currently ongoing work.

The Figure 4 shows the technological demo visualization which provides the functionality related to the first year of the: WP2 (e.g. lemmatization, tokenization, POS tagging, entity recognition, etc.), WP3 (e.g. cross-lingual annotation), and some WP4 functionalities (e.g. cross-lingual similarity)

Figure 4 Technological Demo

This technological demo provides the following functionalities:

- Text language identification
- Shallow processing (including sentence splitting, tokenization, lemmatization, POS tagging, and entity recognition)
- Annotation of a text based on similarities functions.

2.4 Early Prototype Y1[11]

The sandbox is currently also being used for allocating the early prototype of the XLike project. This prototype is the result of the integration of the work done at WP1, WP2, WP3, WP4 and WP5 during the first year of the project. During the first year it has not been done any work related with improving the performance of the platform (from a point of view of using parallelism or scaling to another platforms although all the services have been independently developed trying to improve its individual performance e.g. by using threads whenever are available) but most of the effort has been put into establishing a unique general framework (including platform and software) in order to allow the validation of the defined use cases and also to provide a good support/baseline for the work to be done during the next years Y2 and Y3 of the project.

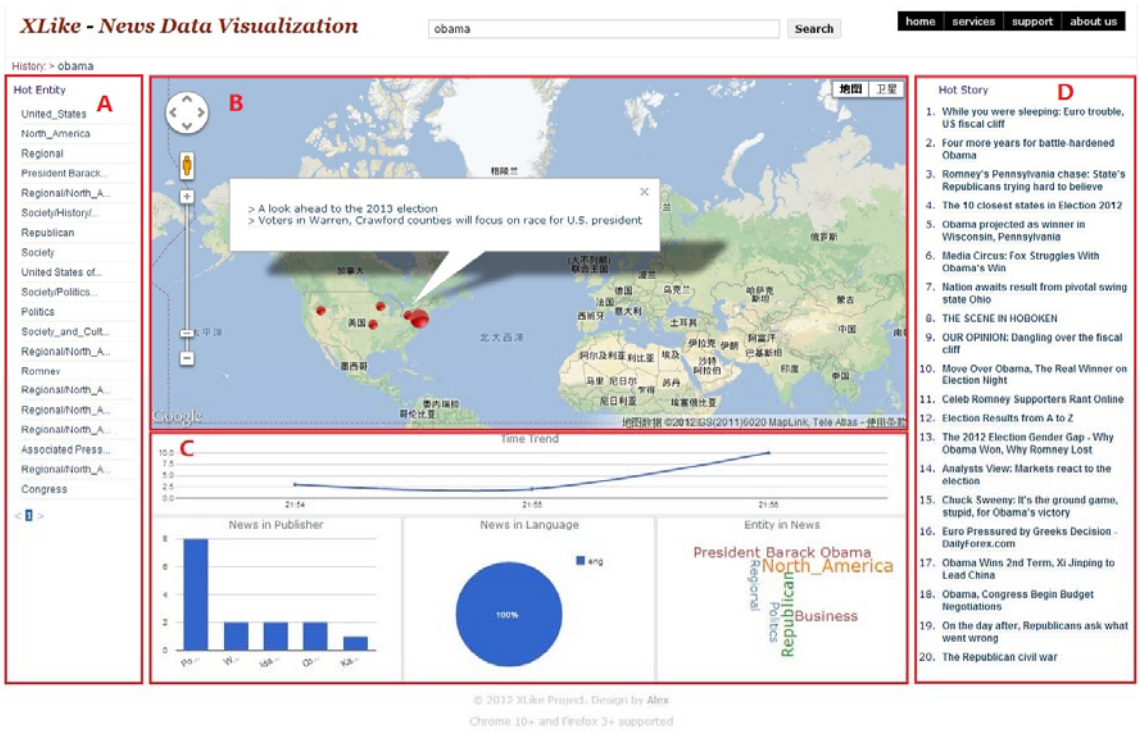

Figure 5 XLike prototype

The Figure 5 shows the visualization (WP5) of the prototype which is using the different services from WP1-WP4 to provide the interface which displays the news articles in a clear way and also the enriched and structured data which includes the article itself, stories, and entities (see D.5.2.1 for a deeper explanation of each one of the parts of the visualization interface and also for a description of the data format).

[11] http://sandbox-xlike.isoco.com/demo/

3 Architecture Specifications and Integration: XLike Toolkit

3.1 Introduction

This section describes the specific architecture and the components that have been deployed as part of the architecture explained in the section 2 of this document. This specification can be seen at Figure 6 where the initial and current deployed services by each one of the partners are shown. Notice that the services which currently have been moved (from green to red) indicate that they have already accomplished with some desired overall properties as **loose coupling, autonomy, and reusability.**

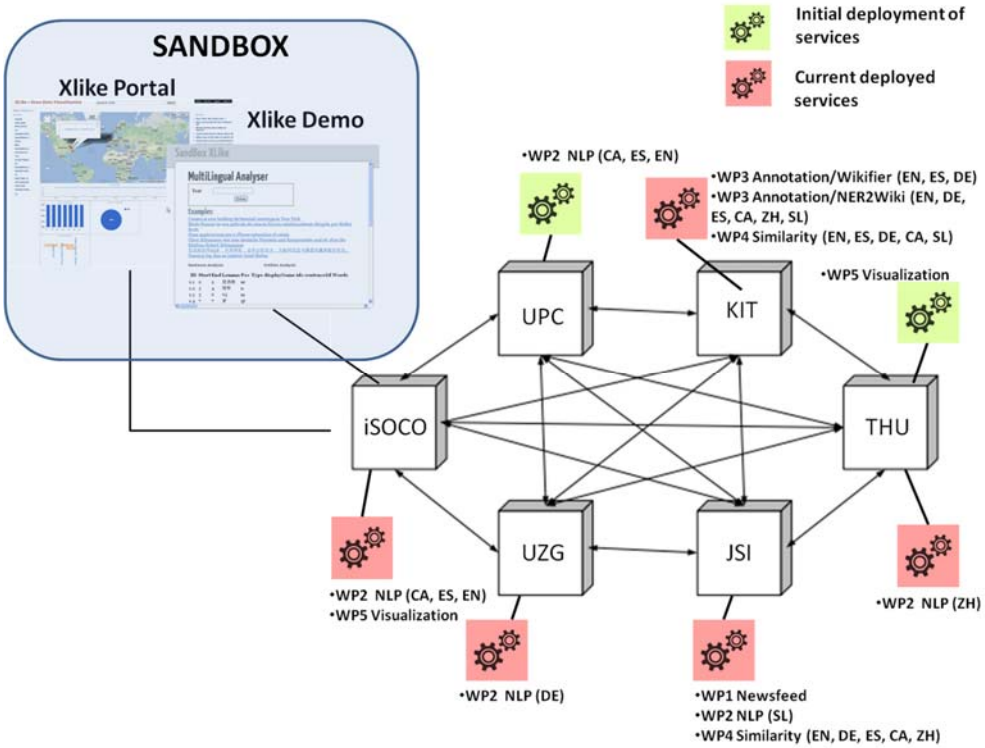

Figure 6 XLike REST, HTTP + XML point to point communications

In the following year this properties would be desirable for all the other services especially if an improvement in performance is needed due to real time requirements and scalability.

3.2 Use case scenario review

This section describes briefly the use case scenarios on which the prototype-Y1 has been focused on and also includes the overview of how they have been achieved successfully by integrating the different deployed services in the XLike project. The four use cases are the following:

- **Entity Tracking** (Bloomberg)
- **Related Relevant Articles** (Bloomberg)
- **Article Tracking** (STA)

- **Topic and Entity Tracking** (STA)

Related with the **entity tracking (Bloomberg)** a multilingual analysis has been adopted in order to provide a way of getting more updated information by tracking media in different languages.

The functionalities needed for these tasks are mainly the natural language processing (shallow level) of the text for the different languages (es, ca, en, de, sl, and zh), the entity extraction functionality, the annotation including the results of the cross-lingual analysis, and the statistical cross-lingual similarity functionality.

Regarding the **related relevant articles (Bloomberg)** use case it mainly makes use of the services provided by the task 4.1 which matches two documents in order to obtain a similarity measurement about its relatedness. It also needs the access to the data infrastructure and to visualize the resulting related documents which are provided by the WP1 services and WP5 visualization.

Regarding **article tracking (STA)** use case it is similar to the just above mentioned due to in order to track an article it is needed to establish a similarity criteria between documents which is provided by the task 4.1.

The **topic and entity tracking (STA)** use case makes use of the multilingual analysis (WP2 services) to track media in the different languages and to provide the collected data for evaluating if it is of interest for STA which is defined as a set of entities by using its Wikipedia[12] page (e.g. http://en.wikipedia.org/wiki/Slovenia) or by a short text description whenever there is not any link to the entity that they are interested on. If the Wikipage is provided then the annotations done over the article will be used to verify if that specific article is of interest and otherwise a similarity measurement between the article and the description provided will be used for that purpose establishing a threshold.

Therefore the complete pipeline consists of multiple consecutive stages: 1) the news articles are crawled by the JSI newsfeed, then 2) these articles are passed through all of them in order to obtain, for each individual article, a description that is richer in both linguistic and metadata annotations and finally 3) these information is structured to be visualize by the front-end. The data flow for these three steps goes from a raw article text to the annotated XML-encoded version of the document. This annotated XML is embedded verbatim into the final output of the newsfeed which is consumed by the visualization front-end in form of queries.

In the next a brief description of the different integrated and used components of the project for this early prototype is introduced.

Toolkit and Components

The next subsections describe the components of the XLike toolkit. The API reference for all of them can be found at Annex A.

3.3 Sandbox[13] (WP6)

The Sandbox belongs to the **physical layer** of the architecture. Its main purpose is to provide a platform for deploying services and also an early testing prototyping. Some of the functionalities developed inside the project have been deployed in the sandbox for public accessing (e.g. natural language processing for ca, es, and en, and the visualization component) although others which are being used as part of the prototype are being hosted by the different partners of the Xlike project (e.g. the annotation services for the different languages is being hosted by KIT).

[12] http://wikipedia.org
[13] http://sandbox-xlike.isoco.com

At this stage the Sandbox is being used for the double purpose of showing the technical functionalities of the project as demos and also to host the prototype which makes use of the different REST services to accomplish with the STA use cases.

A deeper description of the Sandbox, its specifications and its software components can be found in deliverable D6.1.1. "Early Toolkit Architecture Specification".

3.4 JSI NewsFeed[14] (WP1)

The JSI NewsFeed provides the access to the previously collected data from the different tracked sources. These tracked sources include the two use cases: Bloomberg and STA. Due to privacy issues the collected data obtained for these use cases is only available for testing and validation purposes but not accessible to the public domain. It is worth to highlight that any other collected data is available and can be consulted/accessed through the XLike prototype interface[15].

This component belongs to the **acquisition layer** and makes available the data via a web service API which allows obtaining the entities, stories, articles, and searching capabilities (see D5.2.1 for a deeper description).

This component acts as a control structure in the overall pipeline of the project. Of particular interest here is the stage that acts as a client for the linguistic services of WP2, WP3, and WP4. It invokes the services with a raw article text as input and obtains the XML-encoded annotated version of the document. The annotated XML is embedded verbatim into the final output of the newsfeed (see Annex C).

The description of its global architecture can be found at D1.3.1 including how it is deployed to make it available externally via a web service API[16] (see D1.3.1 and D5.2.1). Currently the acquisition process is being done by the partner JSI and hosted in their institution.

3.5 Shallow linguistic processing (WP2)

The language processing capabilities of the project are multilingual and mainly split into shallow and deep processing. The shallow processing capabilities cover the language detection, sentence splitting, tokenization, lemmatisation, POS/MSD-tagging, and named entity recognition. The work related with this component has been developed early in the project (M6) and this has allowed that later until the development of the first prototype (Y1) it has been tested exhaustively and also an important work on standardization of the different languages outputs has been applied. This effort has been grounded in the D2.1.1 where an initial data schema was defined. Afterwards this schema has been also updated (can be consulted at[17]) showing the schema updates including the added annotations provided by the early annotation prototype.

3.6 Early deep linguistic processing (WP2)

Complementary to the Shallow Linguistic Methods, these set of methods perform linguistic analysis of the syntactic structure of sentences. As input, it receives the text itself together with the shallow analysis of the text; hence, we run shallow and deep linguistic processing in a pipeline configuration allowing the independency between the different languages. As output, it produces a syntactic structure, in the form of

[14] http://newsfeed.ijs.si/xlike/
[15] http://sandbox-xlike.isoco.com/portal/
[16] http://newsfeed.ijs.si/xlike/
[17] https://github.com/xlike-project/wp6/blob/master/schemas/document.xsd

a labelled tree that represents the syntactic relations between the words of the sentence. At this point of the project, all six languages of the project have a first prototype that predicts the syntactic dependencies. Over the course of the project (Y2 and Y3), methods that analyze predicate-argument structures, which will facilitate semantic tasks will be implemented. This component still needs to be more robust with respect to receiving input from the shallow methods which will be accomplished also during the next year. So far this component is not incorporated into the early prototype because it is at early stage development and it was not needed by the defined Y1 use cases.

3.7 Informal language analysis report (WP2)

A set of tools that evaluate the linguistic processing methods on texts that are informal have been developed. This component is based on the Google Web Treebank, an annotated corpus that contains a standard newswire texts, together with web blogs and emails, all of them annotated with linguistic structure. A first set of evaluations have allowed the quantification of how the error increases when ranging from standard texts to informal texts. This also provides information about what type of errors are introduced allowing to improve the component during the Y2 of the project. So far this component is not incorporated into the early prototype because it is at early stage development and it was not needed by the defined Y1 use cases.

3.8 Early Annotation prototype (WP3)

The early annotation prototype mainly provides a set of services which allow to semantically enriching the article with information regarding its entities and their description by linking them to Wikipedia articles (this provides a deeper description of the entities which occur in the raw text). Furthermore some of the implemented services also provide a set of functionalities which allows the measurement of the similarity between text fragments to wikipedia articles. These cross-lingual services are also used for the entity tracking use cases specially whenever there is only a description of entities to track. Alternatively a link to the entity Wikipedia pages can also be extracted by comparison and it can be used for translating the descriptions into Wikipedia links.

The information that this components add to the previously collected data from the different sources is a set of annotations which are related to specific entities, and also contains some descriptions of them by indicating the Wikipedia URLs. Therefore the article is at some point described by the list of annotated multilingual URLs.

3.9 Statistical cross-lingual document linking (WP4)

The statistical cross-lingual document linking researches how to compute similarities between documents written in different languages based on the Wikipedia multi-lingual comparable corpus. The cross-lingual similarities algorithms are based on an aligned set of basis vectors obtained by one of following statistical methods: 1) latent semantic indexing (LSI), 2) a generalized version of canonical correlation analysis (CCA) by using an aligned multi-lingual corpus, or by 3) explicit semantic analysis (ESA). Basically all of these methods allow the mapping of multi-lingual documents into a common low dimensional (“semantic”) space where the similarity can be computed fast and efficiently.

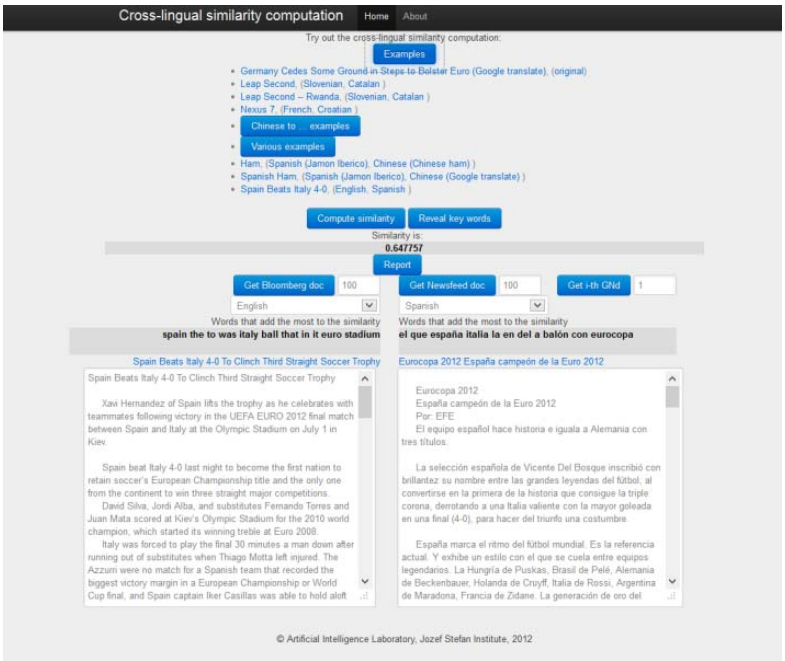

Figure 7 Cross-lingual document linking standalone Web application

These functionalities have been implemented for the following languages: 1) and 2) in English, German, Spanish, Chinese, Slovenian and Catalan, and 3) in English, German, Spanish, Slovenian and Catalan. The Figure 7 shows the developed web prototype for 1) and 2) which is available at[18]. The prototype allows comparing between documents written in different languages and reveals the words that added the most to the similarity score. There is also a web service deployed at[19]. Currently this functionality is being used for Bloomberg use case as an internal service that computes similarity between English, German, Spanish and Chinese news articles. Given a newsfeed article it returns 10 most similar newsfeed article IDs and their corresponding similarity scores for each language for the current day. A demo of this functionality applied to the Bloomberg use case can be accessed at[20]. It illustrates (see Figure 8) the internal service used for retrieving the most similar German articles when given a Bloomberg article. Also the bloombergness score that measures the relevance of the article for the Bloomberg is displayed. Similarity and bloombergness score computation services will be extended to deal with all the languages during Y2.

There is also a service for 3) which allows the use of the ESA approach to obtain a specified number of documents from wikipedia similar to a given one[21]. This service has been use for STA use case in order to provide a set of annotations/URLs for the different languages (see Annex C) associated to an article allowing the tracking of specified STA entities.

[18] http://xling.ijs.si:1111/wikipedia.html
[19] http://xling.ijs.si:1111/clsi
[20] http://xling.ijs.si:1111/bloomberg.html
[21] http://km.aifb.kit.edu/services/clesa/analyzer

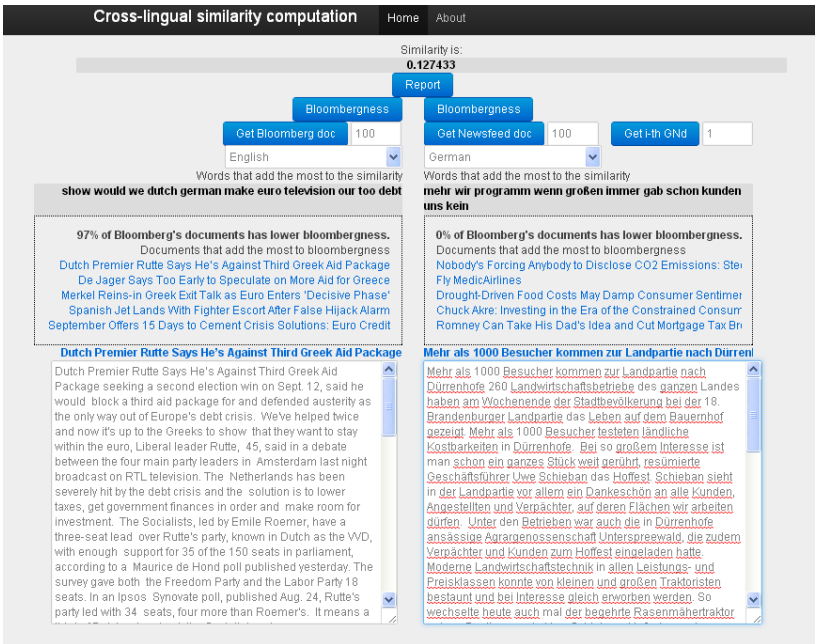

Figure 8 Bloomberg cross-linguality demo

3.10 Early Information Visualization (WP5)

The early information visualization component aims to be the main HCI for the XLike project. It is use-case oriented due to the information that it provides is related with the defined requirements at D1.2.1 *"Requirements for early prototypes".* It is language independent and it relies on the different pipelines in order to obtain the needed information to be displayed. This needed information is provided by the services deployed at JSI (see Annex A WP1 Definition and Data Provision) which also make use of the services provided at WPs 2-4 to collect the needed data and structure it accordingly to the use cases.

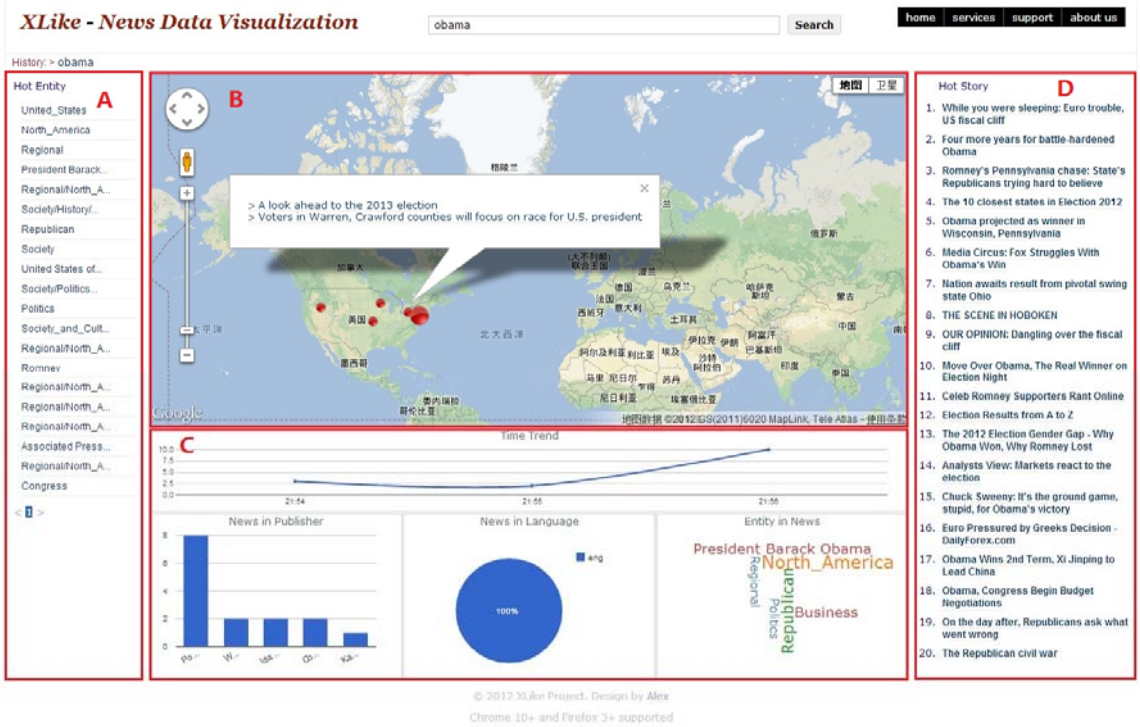

Figure 9 Early prototype visualization[22]

The early prototype visualization is shown in the Figure 9 and its description is available at D5.2.1. The visualization component is developed by using JavaScript technologies included into HTML and making calls to the Google API for showing the geographical map where the articles of interest are pinned. This implementation specification make easy to move the visualization component from one platform to another as has been done during this first year switching it from THU to iSOCO accomplishing also with the architecture desired characteristics of **loose coupling, autonomy, and reusability.**

[22] http://sandbox-xlike.isoco.com/portal/

4 Conclusions

This document has presented the description and first implementation of the XLike prototype as result of the work done during the first year by all the partners at the different work packages (WP1-WP5) and specially the work done at WP6 pursuing a real integration for early assessing of the functionality and early project development.

This first prototype has been mostly focused in the accomplishment of the two use cases defined in D1.2.1 which are related with the topic and entity tracking for STA, and entity tracking and relevant articles discovery for Bloomberg. The use cases have been validated independently by using the developed prototype and these results can be consulted at D7.1.1.

Furthermore, a technical demo has also been developed in order to allow the rapid development environment for early testing of the ongoing work at the different WPs. This demo is also used for showing the technical functionalities developed so far and to allow the integration of the newer functionalities without having to modify the overall prototype. This has turned out to be a very useful idea in order to merge the work at different WPs and the presented prototype.

Regarding the proposed methodology at D6.1.1 it has been shown to be very successful due mainly to the independency between the different offered functionalities, which has been accomplished by using a service oriented approach, and also due to the well defined pipeline which allows a small number of calls (compared with an all to all functionalities relation) to achieve the pursued goals of the project. We have also to point out that although this approach has many advantages it has been needed some extra-effort in order to provide on time and standardized all the different services/functionalities from previous packages to be used on later ones (see Annex C for a better understanding of the data enrichment process throughout the pipeline).

The communication plan composed mainly into internal team work by each partner, bi-weekly/weekly videoconference meetings, inter-partners casual mailing/chatting, hackhaton meetings for face to face global integration, and the use of rapid development as needed based on SCRUM methodology has worked successfully and we plan to keep working in this way for the next year (Y2).

This document is also the first of three (D6.1.1 Y1, D6.2.1 Y2, and D6.2.3 Y3) which will update the prototype and the different already developed functionalities (or create the new ones) in order to achieve the use cases defined by Bloomberg, STA and also the industrial showcase (M24) which can be consulted at D8.2.1. This deliverable will also be used to provide some feedback for the D6.1.2 *"Final Toolkit Architecture Specification"* which will provide the Toolkit specifications.

References

[1] XLike deliverable[23] "*D1.2.1 – Requirements for early prototype*"

[2] XLike deliverable *"D1.3.1 – Early prototype data infrastructure"*

[3] XLike deliverable *"D2.1.1 – Shallow linguistic processing prototype"*

[4] XLike deliverable "*D4.1.1 – Cross-lingual document linking prototype"*

[5] XLike deliverable *"D5.2.1 – Early information visualization prototype"*

[6] XLike deliverable *"D6.1.1 – Early toolkit architecture specification"*

[7] XLike deliverable *"D7.1.1 – Early prototype and validation report"*

[8] XLike deliverable *"D8.2.1 – XLike Showcase specification"*

[9] ŠTAJNER, Tadej, RUSU, Delia, DALI, Lorand, FORTUNA, Blaž, MLADENIĆ, Dunja, GROBELNIK, Marko. A service oriented framework for natural language text enrichment. Informatica (Ljublj.), 2010, vol. 34, no. 3, 307-313.

[23] All the deliverables of the XLike project are accessible at: http://www.xlike.org/deliverables/

Annex A API Definition

This Annex collects the APIs for the web services of all the different components used for the development of the XLike Y1-Prototype

WP1 Definition and Data Provision

Table 1 WP1 Definition and Data provision API

Language	Description	URL SandBox	Parameters	Example of use
INDEPENDENT	Stories	http://newsfeed.ijs.si/xlike/stories	Id: identification of the story to be searched which contains a set of articles	http://newsfeed.ijs.si/xlike/stories?id=1
INDEPENDENT	Entities	http://newsfeed.ijs.si/xlike/entities	Id: identification of the entity to be searched which is contained in a set of article	http://newsfeed.ijs.si/xlike/entities?id=1
INDEPENDENT	Articles	http://newsfeed.ijs.si/xlike/article	Id: identification of the article to be searched	http://newsfeed.ijs.si/xlike/articles?id=1
INDEPENDENT	Search functionalities	http://newsfeed.ijs.si/xlike/search	q: the keyword to be searched which is contained in a set of articles	http://newsfeed.ijs.si/xlike/search?q=obama

WP2 Early shallow linguistic processing

Table 2 WP2 Shallow Linguistic Processing API

Language	Description	URL SandBox	Parameters	Example of use

Language Identification Service	base_url/language_code/ident	http://sandbox-xlike.isoco.com/services/language_code/ident	<analyze><text>Article</text></analyze>	<analyze><text>Bruce Springsteen is an American singer-songwriter and multi-instrumentalist.</text></analyze>
English Service	base_url/ analysis_en /analyze	http://sandbox-xlike.isoco.com/services/analysis_en/analyze	<analyze><text>Article</text><target>relations</target><conll>true</conll></analyze>	<analyze><text>Bruce Springsteen is an American singer-songwriter and multi-instrumentalist.</text><target>relations</target><conll>true</conll></analyze>
Spanish Service	base_url/ analysis_es /analyze	http://sandbox-xlike.isoco.com/services/analysis_es/analyze	<analyze><text>Article</text><target>relations</target><conll>true</conll></analyze>	<analyze><text>Blade Runner es una película de ciencia ficción estadounidense dirigida por Ridley Scott.</text><target>relations</target><conll>true</conll></analyze>
Catalan Service	base_url/ analysis_ca /analyze	http://sandbox-xlike.isoco.com/services/analysis_ca/analyze	<analyze><text>Article</text><target>relations</target><conll>true</conll></analyze>	<analyze><text>L'iPhone és un dispositiu electrònic multimèdia presentat per Apple Computer el 9 de gener de 2007.</text><target>relations</target><conll>true</conll></analyze>
German Service	base_url/ analysis_de /analyze	http://lt.ffzg.hr:9090/xlike/analysis_de/analyze	<analyze><text>Article</text><conll>true</conll></analyze>	<analyze><text>Clara Schumann war eine deutsche Pianistin und Komponistin und ab 1840 die Ehefrau Robert Schumanns.</text><conll>true</conll></analyze>
Chinese Service	base_url/ analysis_zh /analyze	http://keg.cs.tsinghua.edu.cn:8080/analysis_zh/analyze	<analyze><text>Article</text><conll>true</conll></analyze>	<analyze><text>正面的观点认为,由于元朝从忽必烈即位后就开始“行汉法”,</text><conll>true</conll></analyze>
Slovenian Service	base_url/ analysis_sl /analyze?text=text to identify	http://aidemo.ijs.si/xlike/analysis_sl/analyze	<analyze><text>Article</text><conll>true</conll></analyze>	<analyze><text>Clara je bila žena skladatelja Roberta Schumanna in ena vodilnih pianistov in skladateljev romantike.</text><conll>true</conll></analyze>

WP3 Early annotation text prototype

Table 3 WP3 Early Annotation Text Prototype

Language	Description	URL SandBox	Parameters	Example of use
English Service	Wifier: base_url/ annotation-en/ NER based: base_url/ner-annotation-en	http://km.aifb.kit.edu/services/annotation-en/ http://km.aifb.kit.edu/services/ner-annotation-en/	<item> <sentences> <sentence id="">> <text> </text> <tokens> <token pos=" " end="" lemma=" " id="" start="">> </token> </tokens> </sentence> </sentences> <entities> <entity type=" " displayName=" " id="">> <mentions> <mention sentenceId="" id="" words=" "></mention> </mentions> </entity> </entities> </item>	See Table 4 for description
Spanish Service	Wikifier: base_url/ annotation-es/ NER based: base_url/ner-annotation-es	http://km.aifb.kit.edu/services/annotation-es/ http://km.aifb.kit.edu/services/ner-annotation-es/	Same as English Service	See Table 4 for description (the schema is the same than for Spanish)

Catalan Service	NERbased: base_url/ner-annotation-ca	http://km.aifb.kit.edu/services/ner-annotation-ca/	Same as English Service	See Table 4 for description (the schema is the same than for Spanish)
German Service	Wikifier: base_url/ annotation-de/ NER based: base_url/ner-annotation-de	http://km.aifb.kit.edu/services/annotation-de/ http://km.aifb.kit.edu/services/ner-annotation-de/	Same as English Service	See Table 4 for description (the schema is the same than for Spanish)
Chinese Service	NER based: base_url/ner-annotation-zh	http://km.aifb.kit.edu/services/ner-annotation-zh/	Same as English Service	See Table 4 for description (the schema is the same than for Spanish)
Slovenian Service	NER based: base_url/ner-annotation-sl	http://km.aifb.kit.edu/services/ner-annotation-sl/	Same as English Service	See Table 4 for description (the schema is the same than for Spanish)

Table 4 Example of use of the early text annotation prototype

- <item>
- <sentences>
- <sentence id="1"><text>Unesco is now holding its biennial meetings in New York.</text>
- <tokens>
- <token pos="NP00SP0" end="6" lemma="unesco" id="1.1" start="0">Unesco</token>
- <token pos="VBZ" end="9" lemma="be" id="1.2" start="7">is</token>
- <token pos="RB" end="13" lemma="now" id="1.3" start="10">now</token><token pos="VBG" end="21" lemma="hold" id="1.4" start="14">holding</token>
- <token pos="PRP$" end="25" lemma="its" id="1.5" start="22">its</token>
- <token pos="JJ" end="34" lemma="biennial" id="1.6" start="26">biennial</token>
- <token pos="NNS" end="43" lemma="meeting" id="1.7" start="35">meetings</token>
- <token pos="IN" end="46" lemma="in" id="1.8" start="44">in</token>
- <token pos="NP00G00" end="55" lemma="new_york" id="1.9" start="47">New_York</token>

- <?xml version="1.0" encoding="UTF-8" standalone="no"?>
- <item>
- <sentences>
- <sentence id="1">
- <text>Unesco is now holding its biennial meetings in New York.
- </text>
- <tokens>
- <token end="6" id="1.1" lemma="unesco" pos="NP00SP0" start="0">Unesco</token>
- <token end="9" id="1.2" lemma="be" pos="VBZ" start="7">is</token>
- <token end="13" id="1.3" lemma="now" pos="RB" start="10">now</token>
- <token end="21" id="1.4" lemma="hold" pos="VBG" start="14">holding</token>
- <token end="25" id="1.5" lemma="its" pos="PRP$" start="22">its</token>
- <token end="34" id="1.6" lemma="biennial" pos="JJ" start="26">biennial</token>
- <token end="43" id="1.7" lemma="meeting" pos="NNS" start="35">meetings</token>
- <token end="46" id="1.8" lemma="in" pos="IN" start="44">in</token>
- <token end="55" id="1.9" lemma="new_york" pos="NP00G00" start="47">New_York</token>
- <token end="56" id="1.10" lemma="." pos="Fp" start="55">.</token>
- </tokens>
- </sentence>
- </sentences>
- <entities>
- <entity displayName="new_york" id="2" type="location">

- <token pos="Fp" end="56" lemma="." id="1.10" start="55">.</token>
- </tokens>
- </sentence>
- </sentences>
- <entities>
- <entity type="location" displayName="new_york" id="2">
- <mentions>
- <mention sentenceId="1" id="1.9" words="New York"></mention>
- </mentions>
- </entity>
- <entity type="person" displayName="unesco" id="1">
- <mentions>
- <mention sentenceId="1" id="1.1" words="Unesco"></mention>
- </mentions>
- </entity>
- </entities>
- </item>
- <mentions>
- <mention id="1.9" sentenceId="1" words="New York"/>
- </mentions>
- </entity>
- <entity displayName="unesco" id="1" type="person">
- <mentions>
- <mention id="1.1" sentenceId="1" words="Unesco"/>
- </mentions>
- </entity>
- </entities>
- <annotations>
- <annotation displayName="UNESCO" entityId="1" weight="0.926">
- <descriptions>
- <description URL="http://en.wikipedia.org/wiki/UNESCO" lang="en"/>
- <description URL="http://de.wikipedia.org/wiki/UNESCO" lang="de"/>
- <description URL="http://es.wikipedia.org/wiki/Unesco" lang="es"/>
- <description URL="http://sl.wikipedia.org/wiki/Organizacija_Združenih_narodov_za_izobraževanje,_znanost_in_kulturo" lang="sl"/>
- <description URL="http://zh.wikipedia.org/wiki/联合国教育、科学及文化组织" lang="zh"/>
- <description URL="http://ca.wikipedia.org/wiki/UNESCO" lang="ca"/>
- </descriptions>
- <mentions>
- <mention sentenceId="1" words="Unesco"/>
- </mentions>
- </annotation>
- <annotation displayName="New York" entityId="2" weight="0.03">
- <descriptions>
- <description URL="http://en.wikipedia.org/wiki/New_York" lang="en"/>
- <description URL="http://de.wikipedia.org/wiki/New_York_(Bundesstaat)" lang="de"/>
- <description URL="http://es.wikipedia.org/wiki/Nueva_York_(estado)" lang="es"/>
- <description URL="http://sl.wikipedia.org/wiki/New_York_(zvezna_država)" lang="sl"/>
- <description URL="http://zh.wikipedia.org/wiki/纽约州" lang="zh"/>
- <description URL="http://ca.wikipedia.org/wiki/Nova_York_(estat)" lang="ca"/>
- </descriptions>
- <mentions>
- <mention sentenceId="1" words="New York"/>
- </mentions>
- </annotation>
- <annotation displayName="Meeting" entityId="3" weight="0.088">
- <descriptions>
- <description URL="http://en.wikipedia.org/wiki/Meeting" lang="en"/>
- <description URL="http://de.wikipedia.org/wiki/Besprechung" lang="de"/>

	• <description URL="http://es.wikipedia.org/wiki/Reunión_(organización)" lang="es"/> • <description URL="http://zh.wikipedia.org/wiki/會議" lang="zh"/> • </descriptions> • <mentions> • <mention sentenceId="1" words="meetings"/> • </mentions> • </annotation> • <annotation displayName="Biennial plant" entityId="4" weight="0.16"> • <descriptions> • <description URL="http://en.wikipedia.org/wiki/Biennial_plant" lang="en"/> • <description URL="http://de.wikipedia.org/wiki/Zweijährige_Pflanze" lang="de"/> • <description URL="http://es.wikipedia.org/wiki/Planta_bienal" lang="es"/> • <description URL="http://sl.wikipedia.org/wiki/Dvoletnica" lang="sl"/> • <description URL="http://zh.wikipedia.org/wiki/二年生植物" lang="zh"/> • <description URL="http://ca.wikipedia.org/wiki/Planta_biennal" lang="ca"/> • </descriptions> • <mentions> • <mention sentenceId="1" words="biennial"/> • </mentions> • </annotation> • </annotations> • </item>

WP4 Cross-lingual document linking prototype

Table 5 WP4 Cross-lingual Document Linking Prototype

Language	Description	URL SandBox	Parameters	Example of use
EN, ES, DE, SL, CA	Document similarity (to be use for comparison between definition of a STA use case and an article)	http://km.aifb.kit.edu/services/clesa/similarity	doc1 lang1 doc2 lang2	http://km.aifb.kit.edu/services/clesa/similarity?doc1=Bruce%20Springsteen%20is%20an%20American%20singer-songwriter%20and%20multi-instrumentalist.&lang1=en&doc2=Bruce%20Springsteen%20es%20un%20cantante%20y%20m%C3%BAsico%20americano&lang2=es
EN, ES, DE,	Wikipedia analysis based on	http://km.aifb.kit.edu/services/clesa/a	Doc1	http://km.aifb.kit.edu/services/clesa/analyzer?doc=Bruc

SL, CA	ESA	nalyzer	lang1 lang2 retrieve	e%20Springsteen%20is%20an%20American%20singer-songwriter%20and%20multi-instrumentalist.&lang1=en&lang2=es&retrieve=2
EN, DE,ES, SL, CA, ZH	Document similarity between two given documents/articles	http://xling.ijs.si:1111/clsi	Doc1 Lang1 Doc2 Lang2	http://xling.ijs.si:1111/clsi?doc1=car&lang1=en&doc2=coche&lang2=es

WP5 Cross-lingual document linking prototype

Table 6 Cross-lingual Document Linking Prototype

Language	Description	URL SandBox	Parameters	Example of use
ALL	Xlike - News Data Visualization	http://sandbox-isoco.com/portal/	None	Visual human-computer interaction

Annex B Demos and prototypes

This Annex contains the references to the demos and prototypes implemented during the Y1 of the XLike project.

Table 7 Demos and Prototypes

Language	Description	URL SandBox
INDEPENDENT	Technological demo which provides the WP2 functionalities of the Project	http://sandbox-xlike.isoco.com/demo/
INDEPENDENT	Early prototype of the Project (Public)	http://sandbox-xlike.isoco.com/portal/
INDEPENDENT	Early prototype of the XLike Project to be used for validation purposes of the STA use case (Private[24])	http://sandbox-xlike.isoco.com/portal/STA/
INDEPENDENT	Cross-lingual similarity demo	http://xling.ijs.si:1111/wikipedia.html
German (all other languages expected for Y2)	Cross-lingual similarity demo applied to Bloomberg use case	http://xling.ijs.si:1111/bloomberg.html

[24] This prototype makes use of private STA data and therefore it has been agreed to forbid its access to the public.

Annex C Data Format

This annex contains an example of the data and its format collected after going through the pipeline from WP1, WP2, WP3, and WP4 (see Table 8) and shows how this data enriches the raw article by adding it verbatim to the previous information obtained by Newsfeed (see Table 9).

Table 8 Example of the XML format obtained by WP2 + annotations (WP3 and WP4)


```
<item>
        <sentences>
        <sentence id="1">
        <text>Unesco is now holding its biennial meetings in New York.</text>
        <tokens>
                <token pos="NP00SP0" end="6" lemma="unesco" id="1.1"
                start="0">Unesco</token>
                <token pos="VBZ" end="9" lemma="be" id="1.2" start="7">is</token>
                <token pos="RB" end="13" lemma="now" id="1.3" start="10">now</token>
                <token pos="VBG" end="21" lemma="hold" id="1.4"
                start="14">holding</token>
                <token pos="PRP$" end="25" lemma="its" id="1.5" start="22">its</token>
                <token pos="JJ" end="34" lemma="biennial" id="1.6"
                start="26">biennial</token>
                <token pos="NNS" end="43" lemma="meeting" id="1.7"
                start="35">meetings</token>
                <token pos="IN" end="46" lemma="in" id="1.8" start="44">in</token>
                <token pos="NP00G00" end="55" lemma="new_york" id="1.9"
                start="47">New_York</token>
                <token pos="Fp" end="56" lemma="." id="1.10" start="55">.</token>
        </tokens>
        </sentence>
        </sentences>
        <entities>
                <entity type="location" displayName="new_york" id="2">
                        <mentions>
                        <mention sentenceId="1" id="1.9" words="New York"/>
                        </mentions>
                </entity>
        <entity type="person" displayName="organization" id="1">
                <mentions>
                        <mention sentenceId="1" id="1.1" words="Unesco"/>
                </mentions>
        </entity>
        </entities>
        <annotations>
        <annotation URI="" displayName="new_york" lang="es" weight="0.6" entityid="2">
         <mentions><mention sentenceId="1" id="1.9" words="New York"/></mentions>
        </annotation>
        </annotations>
</item>
```


The schema for this example is the following and can be found at[25], and the global structure of the xml file including the information of the article provided by the WP1 is the following[26],[27] [9]:

[25] https://github.com/xlike-project/wp6/blob/master/schemas/document.xsd
[26] http://newsfeed.ijs.si/
[27] The enrycher provided information can be consulted at http://enrycher.ijs.si/

Table 9 Complete XML data format of the XLike prototype

```
<article id="internal article ID; consistent across streams">
   <source>
      <hostname> Publisher hostname </hostname>
      <title> Name of the publisher; failing that, title of the RSS feed </title>
      <location?>
         <longitude?> publisher longitude in degrees </longitude>
         <latitude?> publisher latitude in degrees </latitude>
         <city?> publisher city </city>
         <country?> publisher country </country>
      </location>
      <tags?>
         <tag+> a tag for the publisher; the vocabulary is not controlled </tag>
      </tags>
   </source>
   <feed+>
      <uri> URL from which the article was discovered; typically the RSS feed </uri>
   </feed>
   <uri> URL from which the article was downloaded </uri>
   <publish-date?> The publication time and date. </publish-date>
   <retrieve-date> The retrieval time and date. </retrieve-date>
   <lang> 3-letter ISO 639-2 language code </lang>
   <location? +>
      <longitude?> story content longitude in degrees </longitude>
      <latitude?> story content latitude in degrees </latitude>
      <city?> story city </city>
      <country?> story country </country>
   </location>
   <tags?>
      <tag+> a tag for the article; the vocabulary is not controlled </tag>
   </tags>
   <img?> The URL of a related image, usually a thumbnail. </img>
   <title> Title. Can be empty if we fail to identify it. </title>
   <body-cleartext>
       Clear text body of the article, formatted only with <p> tags
   </body-cleartext>
   <body-rych?; only English, Slovene>
       Enriched article body; an XML subtree as returned by Enrycher.
   </body-rych>
   <body-xlike?; only English, Spanish, Catalan>
       Enriched article body; an XML subtree as returned by iSOCO; experimental.
   </body-xlike>
</article>
```

Article information

Enrycher information

XLike information