EEXCESS

Enhancing Europe's eXchange in Cultural Educational and Scientific reSources

Deliverable D6.2

First Security Proxy Prototype and Reputation Protocols

Identifier:	EEXCESS-D6.1.2-First Security Proxy Prototype and Reputation Protocols_final.doc
Deliverable number:	D6.2
Author(s) and company:	Sonia Ben Mokhtar (INSA), Nadia Bennani (INSA), Lionel Brunie (INSA), Thomas Cerqueus (INSA), Elöd Egyed-Zsigmond (INSA), Omar Hasan (INSA), Albin Petit (INSA), Pierre-Édouard Portier (INSA), Michael Granitzer (Uni Passau), Christin Seifert (Uni Passau), Jörg Schlötterer (Uni Passau)
Internal reviewers:	Know Center
Work package / task:	WP6 / Task 6.2
Document status:	Final
Confidentiality:	Public
Version	30-09-2014

History

Version	Date	Reason of change
1	2014-09-17	Document created.
2	2014-09-19	Inclusion of the content for sections 3 and 4.
3	2014-09-23	Inclusion of the content for sections 1, 2, 5 and 6.
4	2014-09-24	Version for internal review (Know Center).
5	2014-09-24	Final version.

Impressum

Full project title:	Enhancing Europe's eXchange in Cultural Educational and Scientific reSources
Grant Agreement No:	600601
Workpackage Leader:	INSA
Project Co-ordinator:	Silvia Russegger, Jr-DIG
Scientific Project Leader:	Michael Granitzer, Uni-Passau

Acknowledgement: The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 600601.

Disclaimer This document does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of its content.

This document contains material, which is the copyright of certain EEXCESS consortium parties, and may not be reproduced or copied without permission. All EEXCESS consortium parties have agreed to full publication of this document. The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the EEXCESS consortium as a whole, nor a certain party of the EEXCESS consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk, and does not accept any liability for loss or damage suffered by any person using this information.

Table of Contents

1 Executive Summary ... 1

2 Introduction ... 2
2.1 Purpose of this Document ... 2
2.2 Scope of this Document ... 2
2.3 Status of this Document ... 2
2.4 Related Documents ... 2

3 Privacy Proxy Prototype ... 3
3.1 Architecture of the System ... 3
3.2 Features ... 3
3.3 Implementation ... 5

4 Google Chrome Extension ... 6
4.1 User Profile ... 6
4.2 Privacy Settings ... 7

5 Reputation Protocols ... 8
5.1 Introduction ... 8
5.2 Models ... 8
5.3 Integration in the System ... 8

6 Additional Developments ... 11
6.1 Machine Learning Attack ... 11
6.2 Linkability Assessment ... 12
6.2.1 User Profile Similarity ... 12
6.2.2 Group Profile Similarity ... 13

7 Conclusions ... 15
7.1 Summary of the contributions ... 15
7.2 Future work ... 15

8 References ... 16

9 Glossary ... 17

1 Executive Summary

One of the main objectives of the EEXCESS project is to recommend relevant resources to users while they are browsing the web. Among other applications, a Google Chrome extension has been developed to achieve this objective. It allows users to get recommendations based on the page they are viewing, the text they are highlighting, or the query they issue. Ideally the system must provide personalised recommendations to satisfy each user's needs. In order to do that, the system has to collect information about users: user profiles (e.g., name, gender or birth date), browsing history, etc. This situation may lead to privacy violations, and thus restraint users to use the system. Work Package 6 works on the privacy aspects of the project.

As the deliverable is a prototype, this document is predominantly technical. It reports on the work done on the EEXCESS system and on the Google Chrome extension. The main contributions described in this documents are:

- **First implementation of the privacy proxy.** The privacy proxy will be used to filter the information transmitted from the user to the federated recommender according to the user privacy policy. Its role is to avoid privacy breaches. At this stage, the privacy proxy provides with three features: (i) it forwards queries to the federated recommender (no filtering is required, as it is already done on the client side), (ii) it logs the users' activity and (iii) it suggests categories to represent users' topics of interest.
- **Development of graphical user interfaces for the Google Chrome extension**. Two interfaces have been developed. The first one allows users to define and modify their profile (e.g., name, gender, birth date and address). The second one allows users to specify which attributes they want to disclose. For instance, a user may not want to share her birth date, gender, or browsing history with the system. In this case, she can use the user-friendly interface to specify it.
- **Description of the reputation aspects in the project**. We present two possible approaches to integrate reputation in the system. The objective is to consider the reputation of the data provider to help users to choose their privacy setting. This study shows that the integration of a reputation component will raise some issues that need to be discussed with the other partners in the upcoming phase of the project.
- **Additional development.** We focused part of our work on the re-identification of the user. In the literature, an attack shows that it is feasible to link an anonymous query to the public profile of the requester. We implemented an algorithm to evaluate the impact of this attack. Moreover, we developed a protocol to assess the risk of re-identification of the user in the context of this type of attacks. It evaluates if a query can be linked to its requester's public profile. The current implementation gives promising results to integrate this protocol in a future release of the privacy proxy.

These contributions address five of the thirteen requirements that were defined in deliverable D1.1 (section 4.5). It represents a significant progress in the project, especially as four of the achieved requirements were assigned with a high priority.

The last section of the document presents the work that will be made in the next months. It includes the investigation of techniques to increase users' awareness (regarding potential privacy breaches), the study of the link between the accuracy of the recommendation and the privacy preservation, and the development of the second version of the privacy proxy.

2 Introduction

2.1 Purpose of this Document

The deliverable consists of the first proxy prototype for ensuring privacy preservation of user profile, user context and corresponding recommendations.

2.2 Scope of this Document

This document describes the first implementation of the privacy proxy. It covers the integration of the privacy proxy in the EEXCESS architecture, and the Google Chrome extension interfaces to handle user profiles and user privacy settings. This document also describes the reputation aspects that are considered in the project. Additionally, we describe a protocol to prevent the re-identification of the user and assess the risk of re-association of requests.

2.3 Status of this Document

This is the final version of D6.2.

2.4 Related Documents

Before reading this document it is recommended to be familiar with the following documents:

- D1.1 First Conceptual Architecture and Requirements Definition (sections 3 and 4.5),
- D6.1 Policy Model for Privacy and Feasibility Report.

3 Privacy Proxy Prototype

3.1 Architecture of the System

As depicted on Figure 1, the architecture is composed of four major components: the client application, the privacy proxy, the federated recommender, and the set of data providers. In this part of the project, the client application is a Google Chrome extension. It is developed in collaboration with the other partners of EEXCESS.

In the deliverable D6.1, three configurations have been identified. They consider different levels of trust:

L1. All the components of the system are trusted: application (Google Chrome extension), privacy proxy, federated recommender, and data providers.

L2. The application and the privacy proxy are trusted, while the federated recommender and the data providers are not.

L3. Only the application is trusted; all the other components are not.

It has been decided that the first configuration will be considered at this stage of the project. The objective was to integrate the privacy proxy component in the architecture at the early stage of the project in order to get feedback from test users. The current configuration and the current privacy policy are somehow contradictory, as there is no reason to store the user profile on the client side if all the components of the system are trusted. In the future versions, the architecture will change to represent a more realistic case.

3.2 Features

At this stage of the development, the privacy proxy provides three features:

F1. It forwards the queries received from the client application to the federated recommender. In the first version of the privacy proxy, it is not necessary to filter the data provided by the users, as the filtering is done on the client side. As a consequence, the federated recommender uses all the information provided by the user (e.g., her name, her birth date, her gender, her browsing history) to personalise the query and return relevant recommendations.

F2. It logs the activity of users. It considers several types of action including: *query* (when a query is sent from the extension), *activated query* (when the user displays the recommendations sent by the federated recommender), *results* (when a set of results is returned to the user in response to a query), *show/hide extension* (when a user opens or closes the extension panel), *open/close results* (when the user decides to show or hide the sidebar of the extension), *rating* (when a user rates a result by stating if it is relevant for her or not), *facet scape* (when the user interacts with the facet scape visualisation). They are described in more details in deliverable D5.2. For each interaction with the privacy proxy, that is every time a user makes an action, the following fields are logged: a timestamp (date and time at which the action has been initiated), an origin (the version of the plugin installed by the client), and the IP address (address of the machine that initiated the action). In addition, when a set of results is sent back to a user, the following information is logged: the query, and for each result, the identifier of the result and its provenance (i.e., the provider the result comes from).

F3. It invokes a service through a REST API provided by WP5 to suggest categories for the topics of interest. When a user enters a new topic of interests, the service returns a term that corresponds to a category in DBpedia.

In practice, when a query is issued on the client, the query is expanded according to the user's privacy policy and sent to the privacy proxy. The original query is forwarded to the federated recommender (feature F1), and a copy of the information contained in the query and in the results is kept locally (feature F2). Feature F3 is used only when a user is creating or updating her profile.

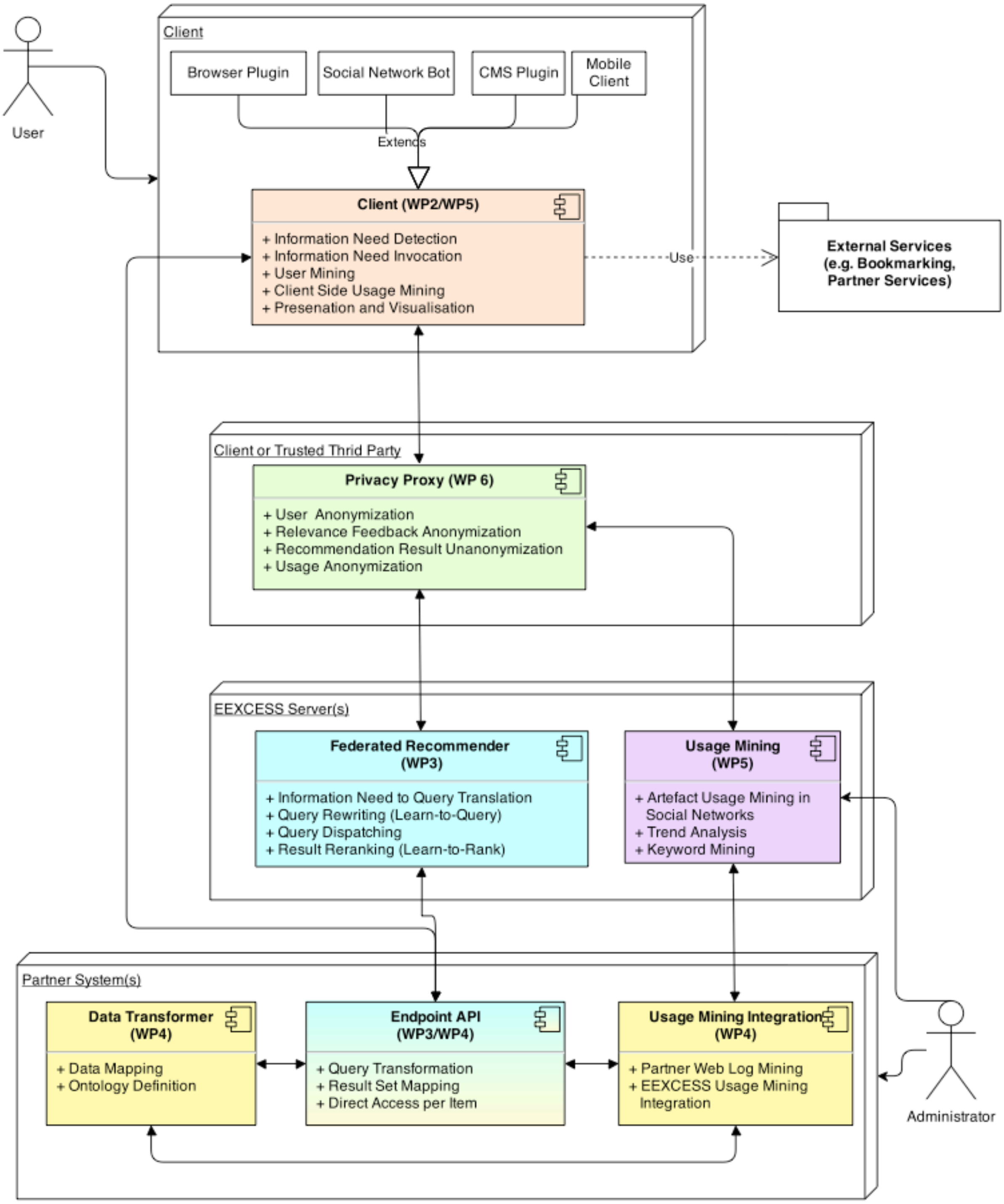

User
Client
Browser Plugin
Social Network Bot
CMS Plugin
Mobile Client
Extends
Client (WP2/WP5)
+ Information Need Detection
+ Information Need Invocation
+ User Mining
+ Client Side Usage Mining
+ Presenation and Visualisation
Use
External Services (e.g. Bookmarking, Partner Services)
Client or Trusted Thrid Party
Privacy Proxy (WP 6)
+ User Anonymization
+ Relevance Feedback Anonymization
+ Recommendation Result Unanonymization
+ Usage Anonymization
EEXCESS Server(s)
Federated Recommender (WP3)
+ Information Need to Query Translation
+ Query Rewriting (Learn-to-Query)
+ Query Dispatching
+ Result Reranking (Learn-to-Rank)
Usage Mining (WP5)
+ Artefact Usage Mining in Social Networks
+ Trend Analysis
+ Keyword Mining
Partner System(s)
Data Transformer (WP4)
+ Data Mapping
+ Ontology Definition
Endpoint API (WP3/WP4)
+ Query Transformation
+ Result Set Mapping
+ Direct Access per Item
Usage Mining Integration (WP4)
+ Partner Web Log Mining
+ EEXCESS Usage Mining Integration
Administrator

Figure 1: System architecture (Figure 3.1 of D1.1).

3.3 Implementation

The implementation of the privacy proxy is composed of five Java classes. They are briefly described in Table 1. The source code of the privacy proxy is available at `http://purl.org/eexcess/components/privacy-proxy`.

Class	Package	Description
PrivacyProxyServices	eu.eexcess.insa	Forwards the queries that are received to the federated recommender.
ProxyLogProcessor	eu.eexcess.up	Stores locally the queries that are received. It uses the Log4J[1] API.
InteractionType		Lists the possible types of data exchange (e.g., query, result set, user feedback).
Util	eu.eexcess	Contains a set of methods used in the project. Most of the methods are used to manipulate literal variables.
Cst		Contains a list of constant variables.

Table 1: List of the Java classes that implement the privacy proxy.

The *PrivacyProxyServices* class is the main interface of the privacy proxy. It provides the methods that allow the invocation of the three features mentioned in section 3.2. The corresponding methods are:

- *responseJSON*: It logs and forwards the query to the federated recommender. The query does not have to be filtered at this level, as it was already filtered at the client level (as described in section 4.2). This service is accessible at http://eexcess.joanneum.at/eexcess-privacy-proxy/api/v1/recommend.
- *log*: It logs all the actions made by a user. It is used to understand users' behaviour and improve the system accordingly. This information is not used in the recommendation process. This service is accessible at http://eexcess.joanneum.at/eexcess-privacy-proxy/api/v1/log/.
- *disambiguate*: It invokes the disambiguation service provided by WP5. The client application does not directly invoke the service in order to avoid possible privacy violations. This service is accessible at http://eexcess.joanneum.at/eexcess-privacy-proxy/api/v1/disambiguate

The JSON format is used to transmit data in the system: queries and logs are described in this format. The choice of the format has been made at the project level. In order to manipulate JSON messages, we use the Jackson library[2].

In order to implement these methods as web services, we used the JAX-WS API[3]. This API determines how Web Services Description Language (WSDL) operations are mapped to Java methods. In practice Java methods are annotated with JAW-WS annotations, such as *@Path*, *@Consumes*, *@Produces*, *@POST*. They are used to trigger the Java methods by send ning SOAP messages.

[1] http://logging.apache.org

[2] https://github.com/FasterXML/jackson

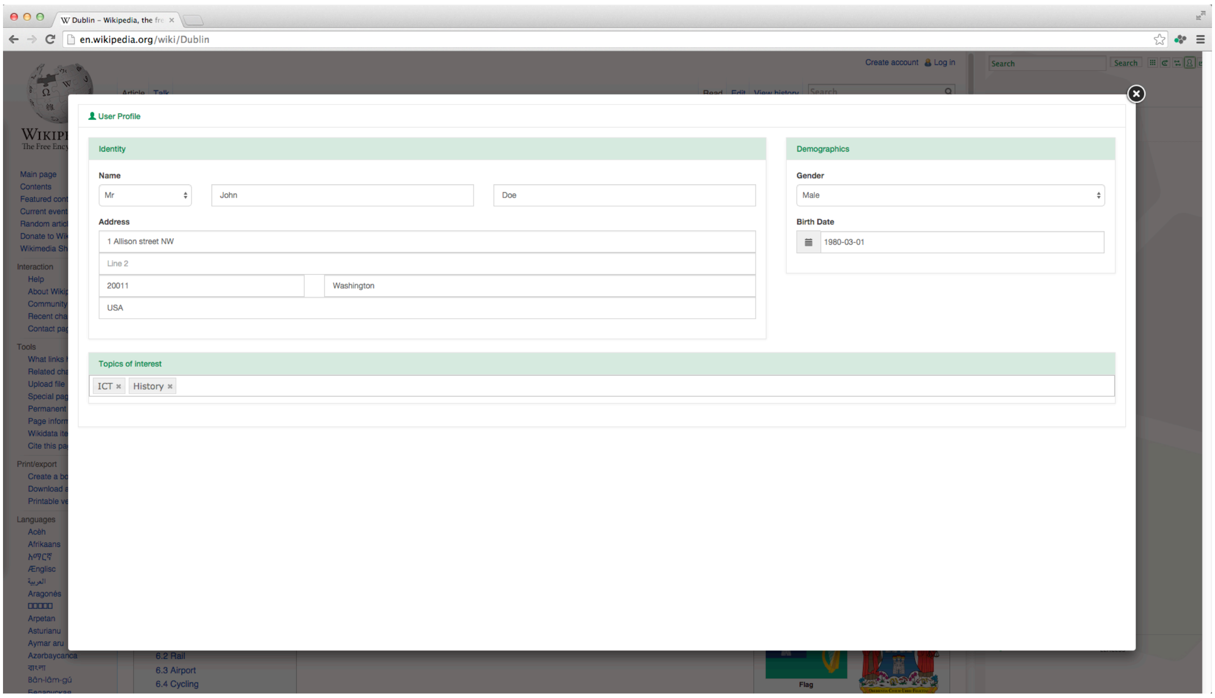
[3] http://docs.oracle.com/javaee/6/api/javax/ws/rs/package-summary.html

4 Google Chrome Extension

This section describes the GUIs of the Google Chrome extension that are related to privacy concerns. The first section describes the user profile interface, while the second one describes the privacy setting interface.

4.1 User Profile

Figure 2 presents the GUI from which a user can specify her profile. Basic attributes are considered: first name, last name, address, gender, birth date and topics of interest. These attributes may be modified at any time.
The GUI is developed in HTML and uses the Bootstrap[4] library to include user-friendly elements (e.g., buttons, menus). The interaction with the back end is made with JavaScript. This language is well suited to this situation, as it does not require interacting the privacy proxy. In the current version, the user profile is stored locally using the IndexedDB[5] API. The API allows the storage of key/value pairs. On one hand, it is convenient to store the data locally, as it provides good performance. On the other hand, it obliges the users to create and update a profile on each device she is using (e.g., laptops, smartphone). In the next release, the users profile will be stored on the privacy proxy. This evolution corresponds to the requirement I.8 defined in D1.1 (section 4.5). It will require considering security and performance aspects.



User Profile

Identity

Name: Mr | John | Doe

Address:
1 Allison street NW
Line 2
20011 | Washington
USA

Demographics

Gender: Male

Birth Date: 1980-03-01

Topics of interest: ICT × History ×

Figure 2: Screenshot of the user profile GUI.

[4] http://getbootstrap.com/css/

[5] https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API

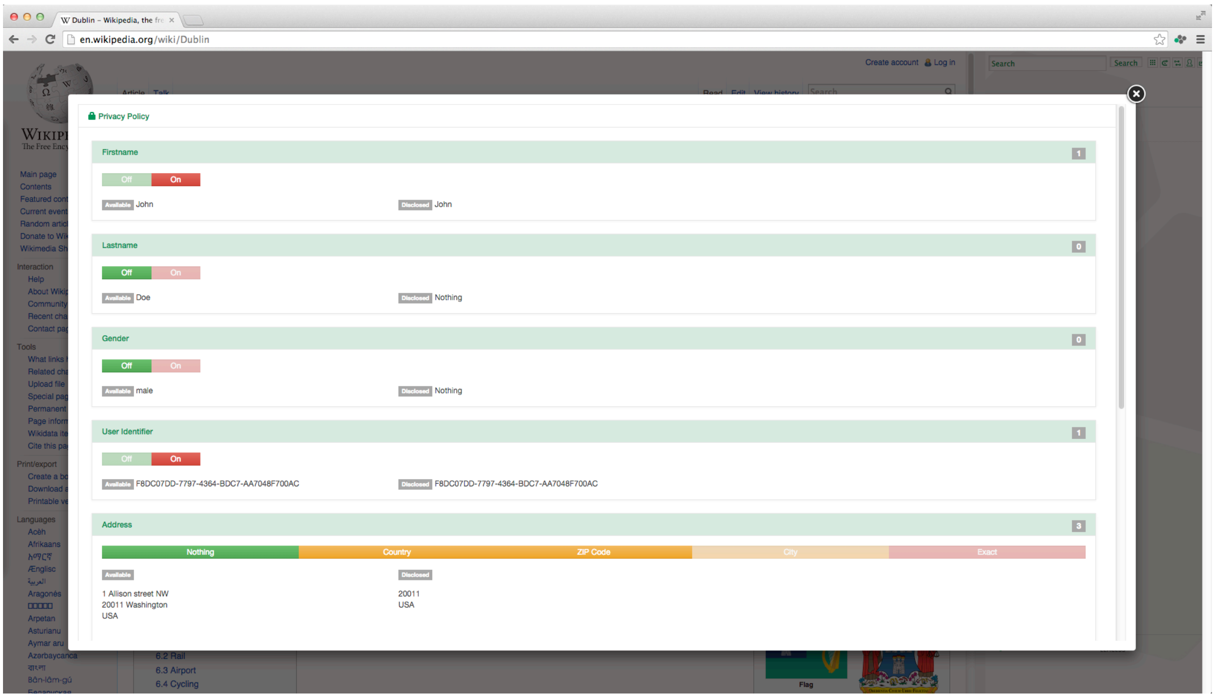
4.2 Privacy Settings

Figure 3 presents the GUI from which a user can explicitly specify what attributes she agrees to pass to the system. The interface allows to hide some attributes (e.g., first name, last name and gender) and to restrict some others (e.g., browsing history, address). For instance, for the address, the user can choose to share:

- Street, city, zip code and country (no restriction: the full address is shared),
- City, zip code and country,
- Zip code and county,
- Country only,
- Nothing (full restriction).

For the browsing history, the user can decide to share it completely, or to restrict it to her last week, month or year of activity.

This fine-grained GUI allows the users to have a strong control on their data. Default values are set to avoid confusion for non-expert users. In the next version of the interface, several easy-to-understand policies will be presented to the users so it is easier to select a suitable policy.



Privacy Policy

Firstname 1
Off On
Available John
Disclosed John

Lastname 0
Off On
Available Doe
Disclosed Nothing

Gender 0
Off On
Available male
Disclosed Nothing

User Identifier 1
Off On
Available F8DC07DD-7797-4364-BDC7-AA7048F700AC
Disclosed F8DC07DD-7797-4364-BDC7-AA7048F700AC

Address 3
Nothing Country ZIP Code City Exact
Available
1 Allison street NW
20011 Washington
USA
Disclosed
20011
USA

Figure 3: Screenshot of the privacy policy GUI.

When a user issues a query (basically composed of keywords), the query is expanded with the context of the user: profile (e.g., name, gender, birth date), browsing history, etc. Obviously, the query only contains the attributes that the user chose to disclose. In other words, the expansion of the query with the user profile is made on the client. The reason for that is that the information about users is stored locally, and not on the privacy proxy.

In the next release, all this information will be stored on the privacy proxy. As mentioned in section 3.1, the evolution corresponds to the requirement I.8 (section 4.5 of D1.1). Consequently, the expansion process will have to be implemented in the privacy proxy. It will consist in four steps:

1. Reception of the query,
2. Identification of the issuer (using its identifier),
3. Expansion of the query with the attributes that the user allowed to disclose,
4. Transmission of the query to the federated recommender.

5 Reputation Protocols

5.1 Introduction

Three levels of trust have been discussed in section 3.1 (L1, L2 and L3). When some of the components are not trusted (levels L2 and L3), it is possible to consider the notion of reputation to estimate their respective level of trust. In particular, if we consider that data providers are not trusted, we can assess their reputation (from a user perspective) to adapt the system accordingly. In this context the reputation relates to the trust level. Thanks to reputation assessment, users can decide what level of privacy they want to set for each data provider. For instance, given three providers A, B and C, a user may want to hide all her personal data to A (because she does not trust it), disclose all her data to B (because she trust it), and share only some of her data with C (because she thinks that it is not safe to disclose all attributes to it). The decision could be made by the user if she has a precise knowledge of the data provider and their internal functioning. If not, the reputation of the different data providers (computed from all users' opinion) would be helpful. A way to compute the reputation is to collect users' feedback (positive or negative) and consider a suitable reputation model.

5.2 Models

As many models and techniques have been proposed in the literature, we do not aim at designing a new model. According to the potential needs of the EEXCESS project, one of the following models could be used:

- Sum and Mean model: Users assign a quantitative feedback value (e.g., -1, 0 or +1) to the data sources according to their perceived trustworthiness. A mathematical operation (sum or mean) is used to get an aggregated value, which is considered as the reputation of the data source. This model is the one preferred by eBay [Resnick, 2002].
- Flow model: In this model, the trust or the reputation is computed iteratively by searching the graph formed by the participants of the system. It is specifically suited for distributed systems. The PageRank algorithm relies on this model [Page, 1999].
- Bayesian model: As in the Sum and Mean model, users assign a quantitative feedback value to the data sources. The global reputation of a source is based on a beta probability density function [Jøsang, 2002].

More detailed descriptions of reputation models are presented in [Jøsang, 2007] and [Hasan, 2010].

5.3 Integration in the System

The integration of the concept of reputation in the project requires considering the potential approaches. In the remaining of this section, let us consider that:

- Users issue queries from the Google Chrome extension (it is the case in the current version, and it will remain true in the future versions),
- User profiles are stored in the privacy proxy (it will be the case in the next version),
- Users can specify privacy settings for each data provider (in the current version the privacy setting specified by the users are applied for all the providers).

In the next version of the system, when a query Q is issued, it is transmitted to the privacy proxy that can expand it with the personal data (PD) of the issuer according to her privacy setting (PS)[6]. The expanded query Q' is then sent to the federated recommender that forwards it to all the data providers. Figure 4 illustrates this process. In this example, all the data providers (i.e., A, B and C) receive the same query Q'. The

[6] Even if the privacy setting is specified by the user, PS is stored in the privaxy proxy in order to support multi-client usage.

drawback of this approach is that all the providers are considered with the same level of trust (which may not be realistic).

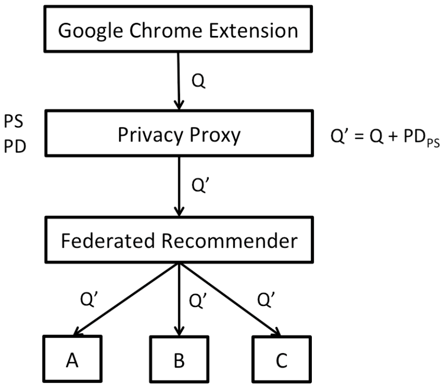


Figure 4: System architecture and query transmission process.

(PS: Policy Setting, PD: Personal Data, Q: Query, Q': Personalised Query)

Figure 5 presents two potential approaches to consider the concept of reputation in the system. In both approaches, users need to specify their privacy setting for all the data providers (PS-A, PS-B and PS-C).

In the first approach (Figure 5-a), when a query is received on the privacy proxy, it is expanded with the personal data of the user. The expansion takes into account the privacy setting of each provider and potentially uses different subsets of the personal data of the user (PD_{PS-A}, PD_{PS-B} and PD_{PS-C}). The resulting queries (Q'_A, Q'_B and Q'_C) are then forwarded to the federated recommender. In turn, the recommender distributes the queries to the appropriate providers: Q'_A to A, Q'_B to B and Q'_C to C. The drawbacks of this approach are:

- The federated recommender receives all the queries (which contains different subsets of the users' personal data). If the federated recommender is malicious (assumptions made in section 3.1 – trust levels L2 and L3), then it could aggregate the subsets of personal data and forward this information to the data providers. As a result, a data provider may be able to access to information that the user intentionally wanted to hide from it. This problem can be addressed by encrypting the messages sent from the privacy proxy (that stored the privacy settings and the personal data) to the data providers. Nevertheless, this solution requires changing the interfaces of the data providers' services.
- The implementation of query forwarding mechanism has to be modified to allow the routing of queries (e.g., to send Q'_A to A but not to B and C).
- The number of queries sent from the privacy proxy to the federated recommender is increased: X queries will be sent, instead of only one in the version presented in Figure 4 (X being the number of data providers).
- Users need to specify privacy settings for all the providers. Even if a user-friendly GUI can be designed, the complexity of this task will increase.

In the second approach (Figure 5-b), when a query is received on the privacy proxy, it is directly forwarded to the federated recommender without any modification. When the federated recommender receives the query, it generates one query for each data provider. As in the first approach, the generation takes into account the privacy settings (PS-*) specified by the user, and her personal data (PD). In this approach, the federated recommender needs to access PS-* and PD to generate the queries (this access is materialised by the dotted arrow in Figure 5-b). Finally, the queries are sent to the appropriate providers. The drawbacks of this approach are:

- As in the first approach, the federated recommender has access to all the data contained in the queries.
- As in the first approach, users have to specify the privacy settings for all the data providers.

- The federated recommender has an access to the privacy settings and personal data (PS-* and PD). This is not permissible if the recommender is not trusted (see trust levels L2 and L3 in section 3.1). This situation strongly questions the role of the privacy proxy.
- As the federated recommender is in charge of the query expansion, its internal implementation needs to be modified. It

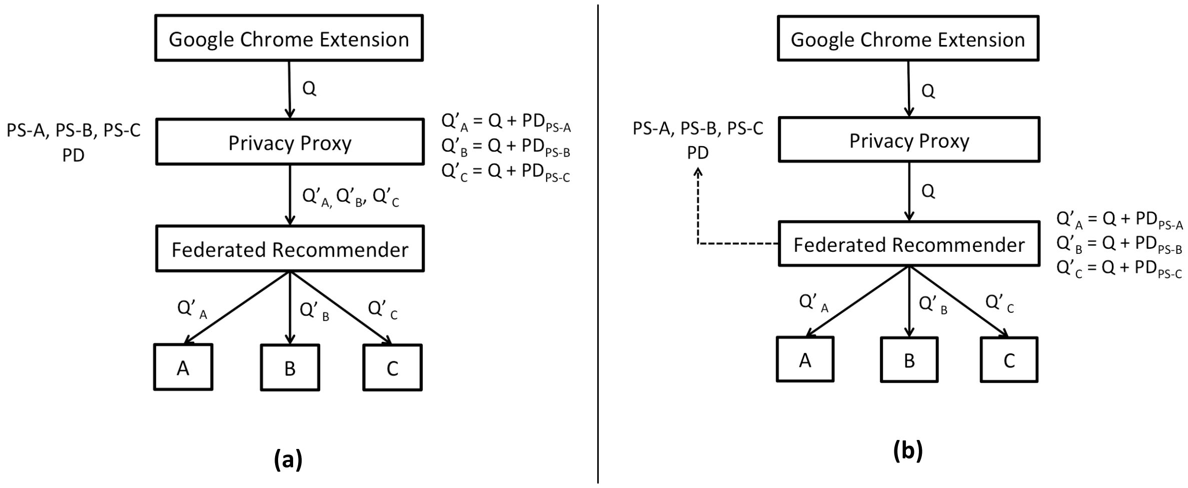


Figure 5: Potential system architectures and query transmission processes to introduce reputation. (PS-x: Policy Setting for provider x, PD: Personal Data, Q: Query, Q'$_x$: Personalised Query for provider x)

To sum up, the integration of reputation in the system raises some issues, as it questions the current implementation of the federated recommender, and the role of the privacy proxy. At this stage of the project the integration of a reputation component is not a priority, as it is assumed that all the data providers have the same level of trust. These aspects need to be discussed with other partners in the upcoming phase of the project.

6 Additional Developments

In addition to the development of the privacy proxy, we implemented two distinct mechanisms:

- Machine learning algorithm: The goal of this algorithm is to discover users' profile from the queries they issue. The ultimate goal is to be able to link a query to its corresponding user profile.
- Linkability assessment: The goal of the module is to assess the probability to link a query to the user profile.

6.1 Machine Learning Attack

It has been shown [Peddinti, 2011] that hiding the identity of the user is not enough to protect her privacy. Indeed, a statistical attack using machine learning techniques showed that it is feasible to identify the requester of an anonymous query. We implemented this attack with WEKA [Hall, 2009]. Our main objective is to evaluate the impact of this attack and categorise queries that seem linkable to user profiles. We used Support Vector Machine (SVM) as machine learning algorithm. We worked on query logs released by AOL in 2006. This dataset contains 20,000,000 queries issued by 65,000 users over 3 months. We consider the first two months as a training dataset and the other one as a test dataset. To evaluate the impact of the statistical attacks, we created 4 subsets with 60 users each:

- Bottom60: approximately 15 queries per user in the training dataset,
- Middle60: approximately 60 queries per user in the training dataset,
- Top60: approximately 2000 queries per user in the training dataset,
- Diversity60: from 15 to 3163 queries per user in the training dataset.

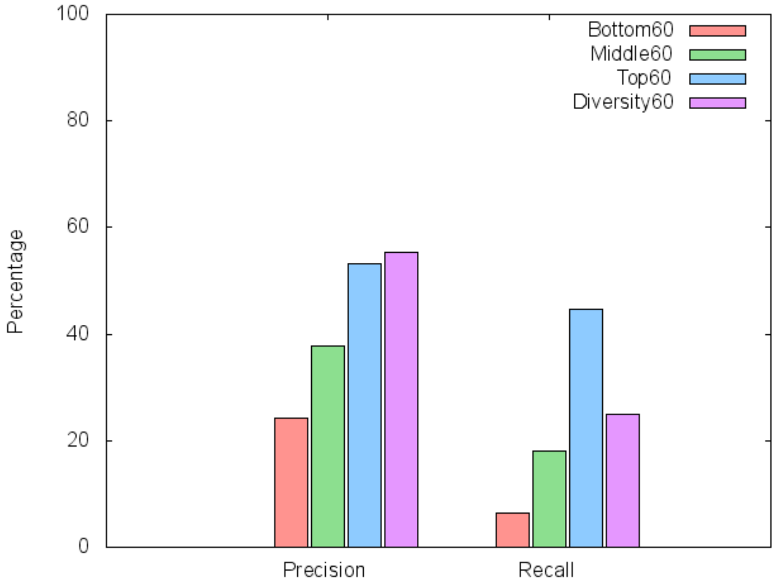


Figure 6: Machine leaning attack results

Figure 6 shows the results of the machine learning attack for the 4 datasets. The recall and the precision are defined as follow:

- Precision = True_Positive / (True_Positive + False_Positive),
- Recall = True_Positive / (True_Positive + False_Negative).

Given a user *u*, True_Positive is the number of queries issued by *u* that have been correctly linked to *u*, False_Negative is the number of queries issued by *u* that have been linked to another user, and False_Positive is the of queries issued by other users that have been linked to *u*.

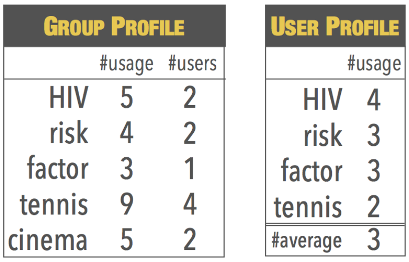
The recall and the precision vary according to the size of the user profile (number of queries in the training dataset). We deduce that the more details are contained in the user profile, the easier it is for an adversary to link a query to a public profile. We have three main explanations:

- The same query has already been issued by the user,
- A subset of the query has already been issued by the user,
- No other user issued a query on this topic.

These explanations are useful to protect the user against machine learning attacks. In the next section, we use this information to design metrics to assess the risk to link a query to its requester's profile (i.e., to assess the linkability).

6.2 Linkability Assessment

The linkability assessment module evaluates if the query is linkable to the requester's public profile. This could be interesting if we want the user to be aware and control her privacy. This module detects if an anonymous query is resistant to statistical attacks (i.e., machine learning algorithms). To reach this objective, we defined a linkability metric that quantifies if the query is likely to be linked to her requester public profile. This evaluation is based on two types of information: the popularity of a query (using the group profile) and the proximity of a query to the user profile. Figure 7 gives an example of a group profile and a user profile. This user profile does not refer to the one described in section 4.1; in this part, the user profile is built automatically from the user request history.



Group Profile		
	#usage	#users
HIV	5	2
risk	4	2
factor	3	1
tennis	9	4
cinema	5	2

User Profile	
	#usage
HIV	4
risk	3
factor	3
tennis	2
#average	3

Figure 7: Example of a User Profile and a Group Profile.

We defined two metrics: *user profile similarity* and *group profile similarity*.

6.2.1 *User Profile Similarity*

The goal of this metric is to assess the degree of similarity of a query to the user profile. For each keyword belonging to the query, we compare its usage frequency a_i to the average usage frequency of all the keywords b_i. More formally, we define the user profile similarity metric *Mu* as:

$$M_u = \frac{\sum_i \frac{a_i - b_i}{a_i + b_i}}{\#keywords}$$

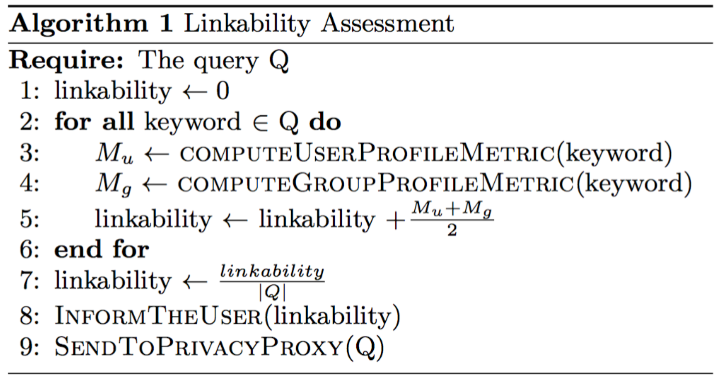
The value of this metric varies from -1 (no part of query was already issued) to 1 (the query was already issued multiple times). If we compute this metric for the keyword "HIV" using the user profile of Figure 7 (the usage frequency being 4 and the average usage frequency of all keywords being 3.75), we deduce that the user profile similarity metric of "HIV" is 0.14.

6.2.2 Group Profile Similarity

The goal of this metric is to assess if a query is popular among the other users in the group. Concretely, it consists in determining for each keyword of the query, how many times c_i, the keyword was used and how many users d_i used it. More formally, the group profile similarity *Mg* is defined as:

$$M_g = \frac{\sum_i \frac{a_i - c_i/d_i}{a_i + c_i/d_i}}{\#keywords}$$

If we consider the keyword "HIV" and the profiles of Figure 7, we obtain a group profile similarity of 0.23.



Algorithm 1 Linkability Assessment

```
Require: The query Q
linkability ← 0
for all keyword ∈ Q do
    $M_u$ ← COMPUTEUSERPROFILEMETRIC(keyword)
    $M_g$ ← COMPUTEGROUPPROFILEMETRIC(keyword)
    linkability ← linkability + $\frac{M_u+M_g}{2}$
end for
linkability ← $\frac{linkability}{|Q|}$
INFORMTHEUSER(linkability)
SENDTOPRIVACYPROXY(Q)
```

The linkability is defined as an average between *Mu* and *Mg* (see Algorithm 1). A low value means that the query is not linkable to a specific public profile whereas a high value means that only one public profile corresponds to the query (consequently, a potential attacker is able to link the anonymous query to its requester).

We consider that a query with a linkability value over the threshold 0 (i.e., a query which is not popular and close to the user profile) is sensitive. Consequently, before sending a query, the system could inform the user about this value. For instance, if we consider the keyword "HIV" and the profiles of Figure 5, we obtain a linkability of 0.18. This value is over 0 and thus the query "HIV" is potentially linkable. The system could advise the user to modify its query.

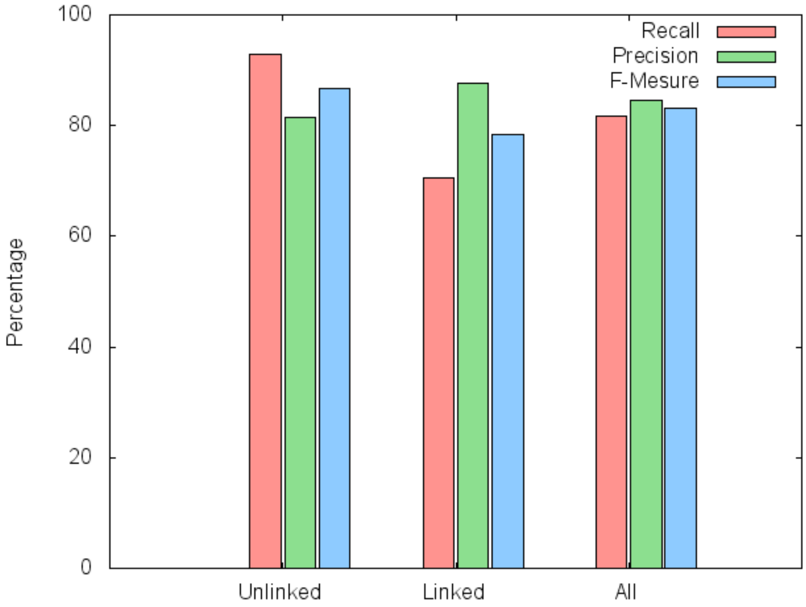


Figure 8: Linkability assessment (preliminary results).

Figure 8 shows the performance of the linkability assessment. We compare its classification (linkable and unlinkable) with machine learning results (linked and not linked). For instance, if a query was flagged linkable by the linkability assessment but was not linked to the user public profile by the machine learning attack (presented in section 5.1), we consider this result as a false positive. Consequently, we compute, for both linked queries and unlinked queries: the precision, the recall and the f-score (defined as the harmonic mean of precision and recall).

As we can see on this figure, the linkability assessment correctly classifies a high percentage of queries (more than 80% in average). However, even if 70.7% of linked queries are retrieved, 29.3% of linked queries are classified as non-linkable and thus not detected by our system. This can endanger the user by leaking personal information. As we want to maximise the protection of the user, we will focus our future work on improving the recall of linkable queries. This work will consist in modifying the current metrics to find a better way to evaluate the linkability.

7 Conclusions

7.1 Summary of the contributions

The main contributions presented in this document are:

- First implementation of the privacy proxy (described in section 3). This contribution fulfils the requirements I.1, I.5 and I.9 described in section 4.5 of the deliverable D1.1.
- Development of graphic user interfaces for the Google Chrome extension (described in section 4). This contribution fulfils the requirements II.2 and II.3.
- Description of two approaches to integrate the notion of reputation in the system in order to deal with the privacy issues that are related to the different data providers.
- Additional developments (described in section 5) give promising results to inform the user about a potential leakage of information.

7.2 Future work

Apart from the development of the next release of the privacy proxy, the next months will be dedicated to work on user awareness. The idea is to send feedback to the users in order to inform them when their activity (i.e. the queries they issue) together with their privacy settings may lead to a privacy breach. For instance, data mining techniques may allow the identification of a user's gender from her browsing history [Wahlstrom, 2009]. We believe that giving more information to a user about her profile will allow her to make better decision regarding her privacy setting. Ultimately, it will allow the recommendation of appropriate privacy policies. This work will fulfil requirements I.4 and II.1.

We also plan to work on the concepts of accuracy and privacy preservation. These two concepts are strongly related: increasing the privacy guarantees may tend to reduce the accuracy of the recommender system. The objective is to find the best trade-off in order to ensure a good quality of services (i.e., an appropriate level of privacy and accurate recommendations). This work will fulfil requirements I.7 and II.4.

In addition, we will work on obfuscation techniques to hide any information that could reveal the identity of the user. This will introduce some noise in the request. Consequently, we plan to work on a post filtering to remove irrelevant recommendations. This work corresponds to requirements I.2 and I.6.

Some of these developments will be integrated in the next deliverable (D6.3). They correspond to requirements I.8, II.1 and II.4. The other developments will be integrated in the final deliverable (D6.4).

8 References

[Hall, 2009] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. “The WEKA data mining software: an update.” ACM SIGKDD explorations newsletter, 11(1), 10-18, 2009.

[Hasan, 2010] Omar Hasan. “Privacy preserving reputation systems for decentralized environments.”, 2010.

[Hasan, 2012] Omar Hasan, Lionel Brunie, and Elisa Bertino. “Preserving Privacy of Feedback Providers in Decentralized Reputation Systems.” Computers & Security, 31(7), 2012.

[Jøsang, 2002] Audun Jøsang and Roslan Ismail. “The beta reputation system.” Proceedings of the 15th bled electronic commerce conference. 41-55, 2002.

[Jøsang, 2007] Audun Jøsang, Roslan Ismail, and Colin Boyd. “A survey of trust and reputation systems for online service provision.” Decision support systems 43.2, 618-644, 2007.

[Page, 1999] L. Page, S. Brin, R. Motwani, and T. Winograd. “The PageRank citation ranking: Bringing order to the web.”, 1999.

[Peddinti, 2011] Peddinti, Sai Teja, and Nitesh Saxena. “On the effectiveness of anonymizing networks for web search privacy.” Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. ACM, 2011.

[Resnick, 2002] Paul Resnick and Richard Zeckhauser. "Trust among strangers in Internet transactions: Empirical analysis of eBay's reputation system.” Advances in applied microeconomics 11, 127-157, 2002.

[Wahlstrom, 2009] Kirsten Wahlstrom, John F. Roddick, Rick Sarre, Vladimir Estivill-Castro, and Denise De Vries. “Legal and Technical Issues of Privacy Preservation in Data Mining.”, 2009.

9 Glossary

Terms used within the EEXCESS project.

Partner Acronyms

JR-DIG	JOANNEUM RESEARCH Forschungsgesellschaft mbH, AT
Uni Passau	University of Passau, GE
Know	Know-Center - Kompetenzzentrum für Wissenschaftsbasierte Anwendungen und Systeme Forschungs- und Entwicklungs Center GmbH, AT
INSA	Institut National des Sciences Appliquées (INSA) de Lyon, FR
ZBW	German National Library of Economics, GE
BITM	BitMedia, AT
KBL-AMBL	Kanton Basel Land, CH
CT	Collection Trust, UK
MEN	Mendeley Ltd., UK
WM	wissenmedia, GE

Abbreviations

EC	European Commission
EEXCESS	Enhancing Europe's eXchange in Cultural Educational and Scientific resource

Acronyms

API	Application Programming Interface
GUI	Graphical User Interface
JSON	JavaScript Object Notation
REST	Representational State Transfer
WSDL	Web Services Description Language
SOAP	Simple Object Access protocol

Acknowledgement: The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 600601.