EU COMMUNITY

ICT-2013.5.4 ICT for Governance and Policy Modelling

EU COMMUNITY MERGES ICT AND SOCIAL MEDIA NETWORKING WITH ESTABLISHED ONLINE MEDIA AND STAKEHOLDER GROUPS TO CULTIVATE TRANSPARENCY, ENHANCE EFFICIENCY AND STIMULATE

Deliverable D3.2.1

Reputation Management Module

(Software/ Documentation)

(first version)

Editor(s):	Manolis MARAGOUDAKIS, Demetrios PYRENIS, Miltiadis KOKKONIDIS
EU COMMUNITY:	FUNDACIO PER A LA UNIVERSITAT OBERTA DE CATALUNYA, University of the Aegean-Research Unit, INTRASOFT INTERNATIONAL SA, EURACTIV.COM PLC
Status-Version:	Final - V1.0
Date:	07.11.2014
EC Distribution:	PU

Project Number:	611964
Project Title:	EU COMMUNITY

Title of Deliverable:	Reputation Management Module (Software/ Documentation) (first version)
Date of Delivery to the EC:	07/11/2014

Workpackage responsible for the Deliverable:	WP3 – Opinion Mining and Reputation Management Component
Editor(s):	Manolis MARAGOUDAKIS, Demetrios PYRENIS, Miltiadis KOKKONIDIS
Contributor(s):	AEGEAN, UOC, INTRA
Reviewer(s):	UOC, EURACTIV
Approved by:	All Partners

Abstract:	The key task of the Reputation Management Component is to assign a reputation score to each person of interest per topic. This reputation score is a synthetic score computed as a combination of scores for a number of reputation criteria. Some of these criteria were specified in D2.4 Community Requirements and specifications. Additional criteria are examined here. Synthetic reputation scoring also relies on a way of combining these individual scores. Both the method envisaged in D2.4 and additional ones are to be made available in the EU Community platform. In addition, a high degree of administrator/user customization will also be supported. In order to ensure the usability of the user

	customisation feature, the necessary computations are separated into off-line and on-line computations. The data schema is appropriately designed to support this division. The present document describes the EU Community Reputation Management System, presents options for its design, and specifies the features implemented in its first version.
Keyword List:	Reputation Management, Synthetic Reputation Score, opinion mining, policy process, Crawlers Component, Europa WhoIsWho, Twitter, LinkedIn, social graph.

Document Description

Document Revision History

Version	*Date*	*Modifications Introduced*	
		Modification Reason	*Modified by*
0.1	18.8.2014	Initial draft version on the basis of the Request for Proposals prepared by Aegean and the proposal sent by HSE. Circulated to INTRA and HSE for comments and contributions.	AEGEAN
0.2	20.09.2014	Input from HSE and INTRA.	HSE, INTRA
0.3	30.09.2014	Consolidated draft version – 1st review	
0.31	03.10.2014	Revised, overall improvements by AEGEAN, HSE and INTRA.	AEGEAN, HSE, INTRA
0.4	14.10.2014	Final edits. Submitted for 2nd review round.	AEGEAN, HSE, INTRA
0.5	30.10.2014	Revised in accordance with partners' comments. Also added mention to the need to handle Linkedin connections in a different manner due to a legal restriction reported by INTRA.	AEGEAN, HSE, INTRA
1.0	07.11.2014	Minor edits. Submitted to the EC.	AEGEAN

Contents

List of Figures

List of Tables

Definitions, Acronyms and Abbreviations

Table 1: Acronyms and Abbreviations

Acronym	Title
JSON	JavaScript Object Notation
RMS	Reputation Management System
SQL	Structured Query Language

Table 2: Definitions

Acronym	Title
Uni-Dimensional RMS	A Reputation Management System that provides a single reputation criterion
Multi-Dimensional RMS	A Reputation Management System that provides multiple reputation criteria
Synthetic Reputation Criterion	A Reputation Criterion whose value is a function of the values of more primitive reputation criteria and a number of configuration options
Synthesis-Enabled RMS	A Reputation Management System that includes one or more Synthetic Reputation Criteria
Synthetic RMS	A Uni-Dimensional Reputation Management System that provides a single synthetic reputation criterion (computed in terms of more primitive reputation criteria which are only used internally and not exposed)
Objective Reputation Criterion	A reputation criterion that has the same value irrespective of the observer i.e. a criterion that will have the same value for all users.
Subjective Reputation Criterion	A reputation criterion the value of which depends on factors including who the observer is i.e. a user-sensitive criterion.
Personalisation (in the context of RMSs)	The ability of an RMS to support subjective criteria and/or per-user configurability of (some of) its options.

1 Executive Summary

This deliverable outlines the design of the Reputation Management System. The Reputation Management System of the EU Community platform combines a number of interesting characteristics:

- It is a synthetic reputation scoring system. Synthetic reputation scoring is typically used in cases where a single actionable score is needed, either for providing a ranking (e.g. web search results) or making a decision (e.g. should a loan be issued to a particular bank customer or not). In EU Community, synthetic reputation scoring is used for both. In particular, it is used for (threshold-based) decision making (deciding which persons of interest have a sufficiently high reputation to appear in the list of experts on a particular topic (or at all) in EurActory, which documents appear on the PolicyLine, and so on and so forth). It is also used for per-topic expert ranking. The computed synthetic reputation score is the result of synthesis of a number of individual `reputation' scores. Whereas the individual scores could be used directly for the same purposes, the resulting decision trees or sorting criteria would either be too simplistic (and fail to capture the same information the synthetic score does), or too complex (resorting to ad hoc reputation synthesis operations, which, typically, both increases complexity and reduces transparency). The synthetic reputation score computed by the RMS provides a convenient unified measure of reputation per person per topic which makes both decision-making and ranking much easier for the various EU Community components than a multitude of atomic reputation scores would.

- The synthetic reputation score is the result of synthesis of individual reputation scores each based on a different reputation criterion and reputation source. These include self-assessment and peer assessment criteria as well as document-related and network-graph criteria. Reputation sources include the EU Platform itself as well as crawled sources.

- A synthesis function is responsible for combining the individual reputation scores for a given person and a given topic into a single synthetic score for that person-topic combination. The synthesis function specified in D2.4, as part of the requirements gathering process, was a simple weighted average function with fixed weights. The Reputation Management Component moves beyond that baseline specification, allowing users not only to specify custom weights for the available reputation criteria, but also to choose among a number of available synthesis functions as well. Administrators can make such choices for the default synthesis function.

- In order to allow such level of user-customisability, while providing real-time feedback on the effects of the user's options and supporting large-datasets, a separation of the processing stages and their corresponding input and output is necessary, so that a significant part of the computational workload is carried out off-line leaving behind data that can then be used for effectuating a user's choices in real-time.

- Interestingly, EU Community's reputation scoring is subjective, meaning that for different users, the score for the same person-topic combination

may be different. This is due to the inclusion of subjective criteria which reflect the fact that a user is likely to attach more significance to his/her own network connections, document evaluations and so on and so forth.

Three versions of the Reputation Management System have been scheduled for delivery. The present document describes the functionality offered by the first version.

2 Introduction

2.1 Objectives and Purpose

This report, entitled 'Reputation Management Module (first version)' is the first of a series of three deliverables of **EU Community** Work Package 3 (WP3). The purpose of each report in the series (D3.2.1, D3.2.2, D3.2.3) is to outline the design of the corresponding version of the Reputation Management System (part of the Opinion Mining and Reputation Management Component).

This deliverable is to fulfil the following objectives:

- Provide general background on the topic of Reputation Management.
- Present and classify the various pre-defined reputation scoring criteria.
- Present the various pre-defined synthesis functions.
- To present the user and administration customisability capabilities of the RMS.
- To present the data schema and the precise algorithms of the first-version of the RMS.

2.2 Relation to other Work Packages/Deliverables

This deliverable is the direct output of Task 3.2 'Reputation Management Design & Component', one of the main tasks of WP3.

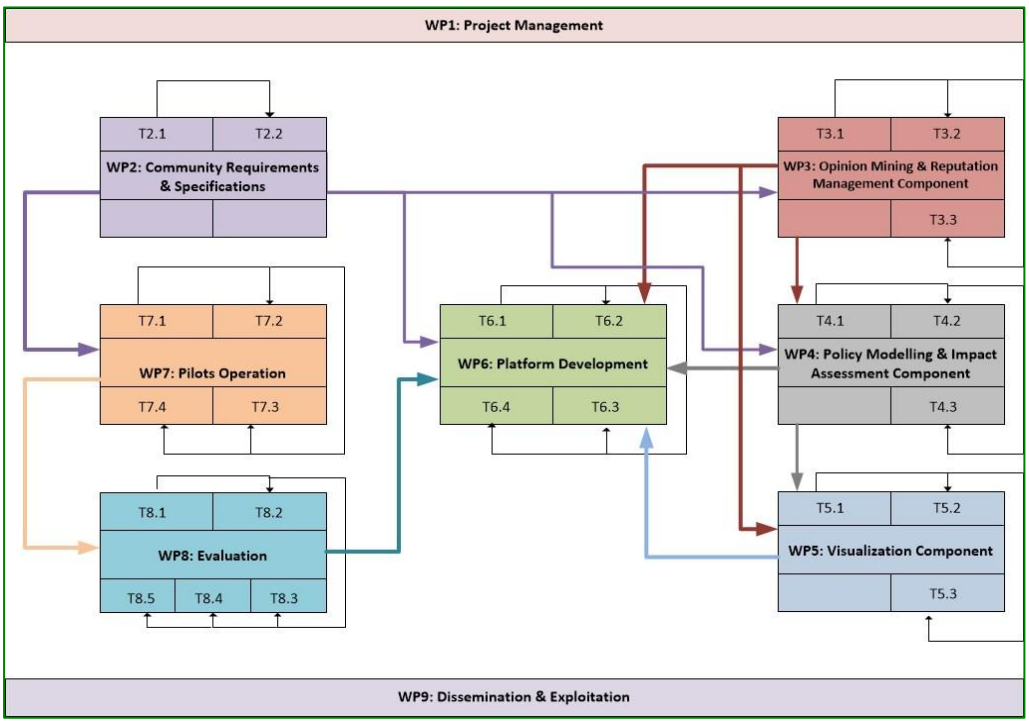

Figure 1: WPs Interdependencies and relations

WP3 covers the design and development of not only the Reputation Management Component, but also of the Crawlers Component and of the Opinion Mining Component. Both the Reputation Management Component and the Opinion Mining Component rely heavily on the Crawlers Component to supply them with the data they will process. The Reputation Management Component additionally relies also on the Opinion Mining Component as some reputation criteria take sentiment into consideration. These are the WP3-internal dependencies of the Reputation Management Component.

The Reputation Management Component also supports a small number of criteria involving input from the users of EU Community; such user input is expected to come from EurActory (WP6). The output of the Reputation Management Component is used by the user-facing components of the EU Community platform, namely EurActory (WP6) and PolicyLine (WP5).

As illustrated in Figure 2:

- EurActory and the Crawlers Component provide data to CurActory (the former by recording user input, the later by importing external content)
- The Opinion Mining Component adds information to CurActory such as document sentiment information.
- The Reputation Management Component pre-computes and stores processed reputation data in CurActory; this is done off-line on a daily basis.
- Whenever reputation scores using custom user-selected synthesis functions are requested, the Reputation Management Component computes on-demand, in real time a synthetic reputation score on the basis of the users' selections and the pre-processed reputation data in CurActory.

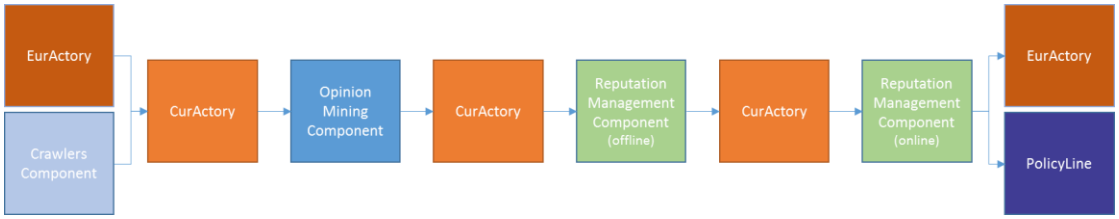

Figure 2: Component Interdependencies and relations

Initial requirements for the Reputation Management System were outlined in D2.4 Community Requirements and Specifications – Research Strategy. The current deliverable and the follow-up deliverables D3.2.2 and D3.2.3 represent / will represent the Consortium's evolving understanding of how to best approach the complex subject of reputation management.

2.3 Structure of the Document

The structure of the deliverable is the following:

Chapter 1 provides an executive summary of the deliverable.

Chapter 2 provides a general introduction to the deliverable placing it in the context of the Work Package and the Project.

Chapter 3 provides a brief outline of the requirements set out for the EU Community Reputation Management system and a comparison with three noteworthy, for different reasons, extant Web 2.0 reputation management systems.

Chapter 4 outlines the Reputation Management System's architecture and its advantages over an initially considered alternative.

Chapter 5 specifies the reputation criteria found in the first version of the EU Community Reputation Management System.

Chapter 6 specifies the synthesis functions found in the first version of the EU Community Reputation Management System.

Chapter 7 specifies the behaviour of the topic- and observer-sensitive reputation scoring function when one of these two important parameters, topic and observer (i.e. current user) is missing.

Chapter 8 specifies the interfaces the EU Community Reputation Management System provides for integration purposes.

Chapter 9 contains the conclusions and directions for future work.

3 Background and Terminology

3.1 Approaching Reputation

The first and possibly most difficult concept to define in the context of a document describing a Reputation Management System is that of `reputation' as it is not a well-defined concept resulting in the term being used to mean relatively different things.

Still, it is possible to attempt to provide a framework for how reputation is obtained. According to a plausible framework for reputation, an individual's reputation depends on:

1. Volume of work
2. Visibility of work (direct/indirect)
3. Quality of work

For instance, Dostoyevsky has a reputation as one of the great authors of all time due to having authored many famous well-received works. Had he authored a single great piece of literature, its reputation would overshadow his. Had his work not been published and read by the masses (direct visibility) and widely talked about (indirect visibility), he would not have had the reputation he currently has. Had his work been mediocre, but visible for other reasons, he could perhaps have achieved fame/infamy but not a reputation as a great author.

One shortcoming of the above reputation breakdown framework is that it fails to quantify reputation and the effect of the three proposed reputation constituents.

Another obvious shortcoming of this framework is that, though it is widely applicable, it does not cover cases where reputation stems from position (head of state, judge, CEO of Microsoft etc.) or other factors. In investigating what reputation is, one needs to keep a very open mind, adjusting any theoretical framework to the needs of the analysis rather than trying to fit the data into the framework.

In the context of Reputation Management Systems such as the one described herein, such theory/data mismatch issues do not / cannot exist: reputation is whatever the Reputation Management System says it is. In other words, each RMS defines its own sense of reputation. The question then is whether this is a *useful* notion in the context that motivated the existence of the RMS in the first place. So, an RMS can only be evaluated with respect to how well suited it is for achieving a specific *purpose*. Moreover, reputation metrics are quantitative by nature and if there is a synthesis of one or more factors this is clearly specified (although the algorithm is not necessarily visible to end users). The following section gives examples of RMSs in Web 2.0 applications.

3.2 Three Example Web 2.0 Reputation Management Systems

Before presenting the reputation management system of EU Community, we examine three well-known Web 2.0 reputation management systems, for comparison purposes.

3.2.1 The eBay RMS

One of the key features of what has come to be known as Web 2.0 is the ability of web applications to empower users to perform tasks where they are, as opposed to requiring their physical presence at specific points-of-service. With that came an opportunity to cater not only to multiple customers but also multiple suppliers, making sites such as eBay, Amazon, and Alibaba trading hubs, whereby hitherto unknown to each other buyers and sellers enter into business agreements.

The first Web 2.0 Reputation Management System to be examined here is that of eBay because it has been and remains emblematic of e-commerce hub RMSs, because eBay is such a big player in the internet-enabled sales arena and because the eBay RMS is such an important part of what is eBay today for its users (buyers and sellers alike).

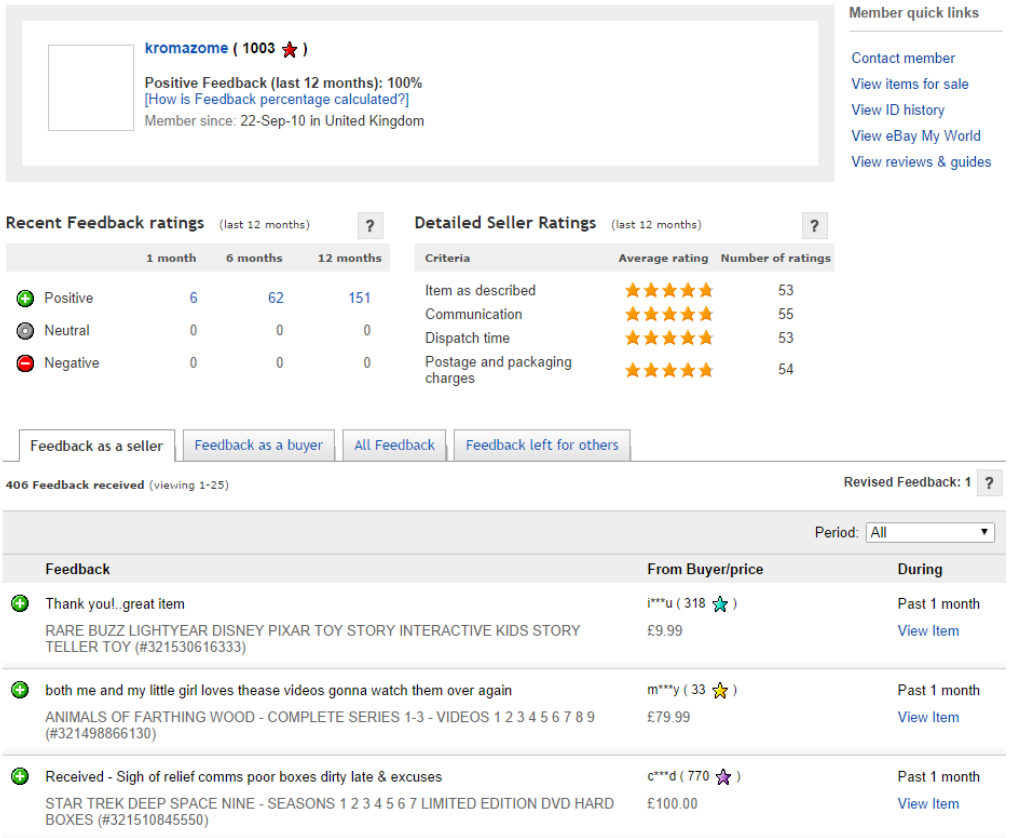

Figure 3: eBay Member Feedback and Reputation Page

The Reputation Management System in eBay supports (see Figure 3):

1. A cumulative feedback score: The number of positive (+1), negative (-1), and neutral (0) feedback ratings a member has received over time.
2. An optional star icon: Stars are badges of honour awarded on the basis of the cumulative feedback score (see Figure 4).
    a. A positive feedback, a negative feedback and a neutral feedback counter for feedback received during the past 1 month.
3. A positive feedback, a negative feedback and a neutral feedback counter for feedback received during the past 6 months.
4. A positive feedback, a negative feedback and a neutral feedback counter for feedback received during the past 12 months.
5. A percentage of positive feedback computed as the ratio of the above positive feedback counter value divided by the sum of the above positive and negative feedback counters (i.e. the percentage of positive feedbacks considers only feedback obtained during the past 12 months).
6. Short feedback messages accompanying (and explaining) the positive, neutral, or negative feedback:
7. Detailed seller ratings: A feature not originally found in eBay, detailed seller rating include 0 to 5 star ratings (computed as averages of relevant feedback received during the past 12 months) on the following four categories:
    - Item as described
    - Communication
    - Dispatch time
    - Postage costs

It is interesting to note that items 1 through 6 in the list above correspond to a single feedback action involving a choice between positive, neutral and negative. Item 7 requires the user giving feedback to add a short explanation for the feedback left and Item 8, an afterthought in the design of the eBay RMS, requires the user to provide a five star rating for up to four aspects of a transaction (some may be deactivated if non-applicable; for example Postage costs does not apply when shipping is free).

Clearly, the eBay RMS relies on eBay users to both provide reputation information and to interpret that information, paying attention to how such information is organised (feedback for the past 1/6/12 months, feedback as a seller/as a buyer/all feedback/feedback left for others). Yet, despite the wealth of available raw reputation data available for perusal, it is the computed positive feedback score percentage together with the computed accumulative feedback score that are the most important tools for judging the reputation of sellers.

Interestingly, though both these computed scores are misleading in the case of non-professional sellers, as they include feedback they collected as buyers. It is somewhat surprising perhaps but eBay's Reputation Management System does

not differentiate between sellers and buyers as much as one might expect, promoting the idea of a good eBayer over that of a good buyer (or seller). What the reputation system of eBay is geared to is creating a community (of eBayers, not just buyers or sellers) and it has to a large extent succeeded in doing that. The fact that the term "eBayer" is quite commonly used by eBay's users is a sign of its success in that respect. Moreover, it encourages buyers to become sellers thus fuelling eBay's growth as a platform for the auctioning of practically everything under the sun and the company's financial growth (as more sales lead to more income from commissions).

Star	Color	Number of ratings
	Yellow	10 to 49
	Blue	50 to 99
	Turquoise	100 to 499
	Purple	500 to 999
	Red	1,000 to 4,999
	Green	5,000 to 9,999
	Yellow shooting star	10,000 to 24,999
	Turquoise shooting star	25,000 to 49,999
	Purple shooting star	50,000 to 99,999
	Red shooting star	100,000 to 499,999
	Green shooting star	500,000 to 999,999
	Silver shooting star	1,000,000 or more (Wow!)

Figure 4 The eBay Reputation Star System

3.2.2 The Google Search URL RMS

Users of the Google search engine may not be faced with numerous metrics and may not be requested to provide any feedback, yet they are provided with a service that relies very heavily on a Reputation Management System that ranks the results of a search query not only by relevance but by a combination of relevance (how closely the query matches a document found at a certain URL and even how closely the URL itself matches the search query) and prominence (how important the URL and its content are deemed to be).

In many ways, the Google Search URL RMS is the exact opposite of the eBay RMS. Google requires no user input nor does it disclose the information it uses in its algorithms which themselves are in turn complex and for the most part secret. Reputation data are collected primarily by means of Google's crawlers. There is no association with financial transactions, a characteristic of the eBay RMS that protects it from manipulation, and due to the high value of placing high on Google's results list the business of Search Engine Optimisation flourishes. Some SEO techniques are welcome (ensuring the textual content of webpages is easily extracted and appropriate navigation is provided) whereas others amount to efforts to maliciously trick Google into assigning a higher prominence (e.g. link spamming) and/or relevance (e.g. hidden text relevant to certain popular queries meant to be crawled and indexed, but not seen by to human visitors) for certain queries and Goggle' staff is in a constant state of alert for detecting and thwarting such, so-called, black hat SEO machinations. Moreover, Google supports so-called personalised searches by means of taking into consideration past search history[1].

This is how Google themselves describe how Google search ranks URLs (blurring the distinction between prominence and relevance to a query, and leaving room for personalisation):[2]

When a user enters a query, our machines search the index for matching pages and return the results we believe are the most relevant to the user. Relevancy is determined by over 200 factors, one of which is the PageRank for a given page. PageRank is the measure of the importance of a page based on the incoming links from other pages. In simple terms, each link to a page on your site from another site adds to your site's PageRank. Not all links are equal: Google works hard to improve the user experience by identifying spam links and other practices that negatively impact search results. The best types of links are those that are given based on the quality of your content.

Unlike eBays's RMS for its members, Google's RMS for the URLs it crawls is not at all about transparency and community building, but rather about company trade secrets and ranking manipulation avoidance. PageRank may be patented and known to the public, but this is only a piece of a large secret puzzle.

3.2.3 The Klout RMS

While Klout remains a small company, it has drawn considerable attention as they offer something relatively unique: a score for a user representing what they call 'influence', which in fact is a measure of how likely others are likely to react (within the context of social media) to that user's postings. They have an interesting business model whereby they provide their service to users for free and additionally arrange for top influencers to receive Perks from companies they work with. The idea is that these top influencers will then have nice things to say and that people will listen. Companies pay Klout for Perks campaigns, whereby they offer free services or products to select Klout users.

[1] https://support.google.com/websearch/answer/54068?hl=en Retrieved: 9-9-2014.

[2] https://support.google.com/webmasters/answer/70897?hl=en&ref_topic=4558960 Retrieved: 9-9-2014.

The company is not particularly forthcoming with details about the way the scores are computed. This is what they present as a "detailed description of how a Klout Score is determined":[3]

The majority of the signals used to calculate the Klout Score are derived from combinations of attributes, such as the ratio of reactions you generate compared to the amount of content you share. For example, generating 100 retweets from 10 tweets will contribute more to your Score than generating 100 retweets from 1,000 tweets. We also consider factors such as how selective the people who interact with your content are. The more a person likes and retweets in a given day, the less each of those individual interactions contributes to another person's Score. Additionally, we value the engagement you drive from unique individuals. One-hundred retweets from 100 different people contribute more to your Score than do 100 retweets from a single person.

We know how important it is to maintain the integrity of the Klout Score, so we closely monitor activity across the signals we measure for inauthentic behaviors—spambots and the like. The Score will continue to evolve and improve as we add more networks and more signals.

Attempts to understand how Klout score works, for social networks such as Facebook, Twitter, Google+, Instagram, Foursquare, and LinkedIn it seems that what counts the most is how much others interact with one's postings rather than how many friends/followers/ etc. he/she has. This is consistent not with a system that attempts to estimate reputation in some generic sense but rather the likelihood than a Klout user will be an effective communicator of any positive opinions on a certain company that may happen to be offering Klout Perks. As mentioned earlier, the exact nature of reputation an RMS computes is very much dependent on the purpose for which the RMS was built and Klout's RMS has a specific business model to serve. However, in order to deal with criticisms about divergence from real-world reputation, as an afterthought, Wikipedia and Bing are now also used as reputation sources.

3.3 Key EU Community RMS Requirements

1. **Reputation Criteria:** After many discussions between the partners and with users during CreActiv workshops, the below criteria have been decided to evaluate the reputation and influence of EU stakeholders.

    1. Self-evaluation: input given by the candidate him/herself.

    2. Organization reputation: rating of all members of that organisation.

    3. Position ranking: level of the candidate title (junior/medium/senior). The consortium can provide an initial matrix of titles.

    4. Document assessment: measure of the positive or negative evaluations that a document produced by the candidate has received, such as Facebook shares.

[3]

5. Proximity Trust: the level of connection of the candidate with the user, such as in social media.

6. Network Value: the rating of the candidate close connections, such as in social media.

7. Past measurements: the rating of the candidate for previous months.

2. **Weighted Average Synthesis Function:** Another requirement for the RMS is that the independent criteria scores be combined by means of taking their weighted average.

3. **Configurability:** Unlike other RMSs, the EU Community RMS must allow configurability, both by system administrators *and ordinary users* in the way it computes reputation scores, at the very least by allowing the weights of the various criteria to be configured (see Figure 5). Administrators can configure the behaviour of the system reputation scoring, whereas ordinary users are allowed to provide custom settings (which only apply to them).

4. **Performance:** It should be possible for reputation scores to be computed instantly given a new set of configuration options.

5. **Transparency:** The reputation algorithm also must have a critical feature: it must be open to everyone so if there is an objection regarding the calculation of someone's reputation, he will be able to examine the way the reputation was calculated.

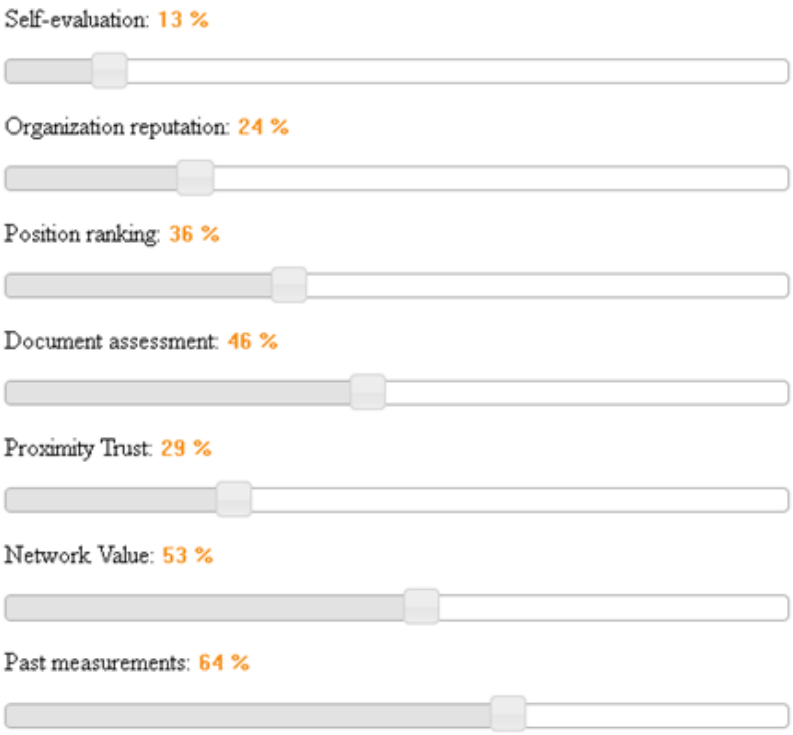

Figure 5: Weight Configuration Mockup Screen

3.4 Comparisons and Conclusions

The three Web 2.0 reputation management systems examined earlier were purposefully chosen in order to frame the discussion of the EU Community RMS within a context of two very familiar albeit very different RMSs (eBay, Google Search) plus an RMS with which it has many similarities (as well as interesting differences) (Klout). The key similarities and differences of the four systems are summarised in **Error! Reference source not found.**. In more detail, this is how the EU Community RMS compares to the eBay, Klout and Google Search RMSs:

- Like the RMSs of Klout and eBay, EU Community's RMS also focuses on the reputation of individuals.
- Like Klout and Google, EU Community relies heavily on crawled content, but unlike them, not exclusively. In its current, first, version it takes into consideration a self-evaluation score users assign to themselves for topics where they think they have a certain level of expertise. It would be interesting to see peer evaluation or other forms of reputation-related user-input in future versions of EurActory and/or PolicyLine; any such input would, of course, be taken into account by means of additional criteria in subsequent versions of the RMS.
- Given that it measures the reputation on persons not necessarily included in its user base, relying heavily on crawled reputation data, the EU Community RMS is very similar to Klout. Yet, it is unlike Klout in that it offers per topic reputation scores.
- Like Google Search, and unlike Klout and eBay, EU Community supports subjective reputation criteria (or in Google's terminology, it supports personalisation). However, while Google Search relies on past search information, the current version of the EU Community RMS relies on crawled social graph relations (see §5.5).
- The EU Community RMS is unlike all three other systems examined in offering significant customisability to users. Google Search does offer customisation but it is relatively limited in scope. Klout and eBay offer no customisability whatsoever.
- Transparency is explicitly required of the EU Community RMS. All its algorithms are open-source; Google and Klout's algorithms are considered trade secrets and especially in the case of Google secrecy is also considered a means of avoiding more black hat SEO attempts. Out of the three extant RMSs examined, eBay's RMS stands out in that respect both because it is fully transparent but also because it is particularly simple.

In summary, reputation calculations in eBay are transparent and use user-feedback, Google Search is personalisable and customisable and while Klout lacks those features, found in the EU Community RMS, it is the closest of the three examined RMSs to it in term of its goals (estimate the reputation of people that are not necessarily users of the site heavily relying on crawled data). Given that EU Community

1. targets a specific community of experts,
2. supports a mix of user-input and crawled reputation data,
3. computes reputation on a per-topic basis,
4. implements a variety of criteria not all of which linked to social media

it becomes obvious that the EU Community RMS has qualitative characteristics, in theory, capable of setting it apart from systems like Klout whereby Jean-Claude Juncker may score lower than Justin Bieber or some blogger or other. As this deliverable precedes any systematic effort to test, evaluate and improve the EU Community RMS, it is not possible to have an accurate view of its performance (as these will depend on the characteristics of the group of people of interest to EU Community and their online presence), but it is certain that it brings an entirely new and exciting approach to the table.

Table 3: RMS Comparison Matrix: eBay, EU Community, Google Search, Klout

FEATURE	**YES**	**NO**
REPUTATION FROM USER INPUT	eBay EU Community	Google Search Klout
REPUTATION FROM CRAWLED DATA	EU Community Google Search Klout	eBay
FOCUS ON PEOPLE REPUTATION	eBay EU Community Klout	Google Search (URLs)
SYNTHETIC REPUTATION CRITERIA	eBay (very simple) EU Community Google Search Klout	
COMPLEX COMPUTATIONS IN SYNTHETIC CRITERIA	EU Community Google Search Klout	eBay
SECRET ALGORITHMS FOR SYNTHETIC REPUTATION CRITERIA	Google Search Klout	eBay (simple, documented computations) EU Community (complex, open source algorithms)
PERSONALISATION / SUBJECTIVITY	Google Search EU Community	eBay Klout
CUSTOMISABILITY	Google Search EU Community	eBay Klout

4 Architecture, Performance, Communication and Integration

4.1 Off-Line and On-Line Processing

The Reputation Management System has to satisfy two important and opposing requirements:

1. Support criteria involving complex computations on large quantities of raw data.
2. Provide immediate real-time feedback to users configuring the reputation rating parameters.

In order to satisfy both these requirements the Reputation Management Component needs to perform:

- Off-line pre-computations of the ratings for individual criteria using pre-defined or administrator-defined logic.
- On-line computations using default or user-supplied options for the synthesis function, the result of which is computed on the basis of pre-computed reputation criteria ratings.

4.2 Integration with EurActory: Stored Procedures Collection vs. Communicating Component Design

4.2.1 The Stored Procedure Approach

The chosen approach is to:

(i) Encode the core RMS business logic as SQL stored procedures (which has the advantage of immediate read and update access to the relevant data).

(ii) Re-use the existing Drupal RESTful Web API for exposing the reputation scores (and separate calls for auxiliary intermediate scores).

This design preference is based on consideration of the following facts:

- The majority of primary reputation data come from data gathered by the Crawlers Component. The two exceptions are user-supplied data (e.g. self-assessments), gathered by EurActory, and sentiment analysis data, produced by the Opinion Mining and Sentiment Analysis Component. So, no single component whence some reputation data originates has all required data in order to calculate the reputation scores, but all three of them send their data over to CurActory, the common data repository.
- The single biggest overhead in an RMS that would communicate either directly with CurActory or via the project's Drupal RESTful API to obtain

the aforementioned data (excluding the unavoidable database server file I/O) would be the network communication overhead for getting data and storing back results; the requisite computations (per person, per document, per organisation, or any links between people and people, documents and documents, documents and people, and people and organisations) are trivial (they typically involve a very small number of primary arithmetic operations such as addition, multiplication and division). So for performance reasons alone, the obvious choice is to implement the core RMS business logic in the form of SQL stored procedures (CurActory is a database on a MySQL server). This is common practice for high-performance applications: keeping simple data-centric computation as close to the data as possible. Any solution involving taking data from CurActory, processing it elsewhere, and storing it back would be significantly less efficient.

- There are requirements for extreme customisation by administrators. Since SQL is by design an interpreted language (pre-processing and JIT optimisations aside), these become very easy to adhere to, if criteria scoring functions and synthesis functions are implemented in the common SQL server. In fact, it would be easy to even have the administrator define the so-called pre-defined functions mentioned here. Any auxiliary stored procedures and functions devised for use by the pre-defined criteria and synthesis functions would also be documented and available for use in the implementation of custom pre-defined criteria and synthesis functions.

- There is a requirement that reputation scores are refreshed daily. This can easily be achieved using the MySQL scheduler which can be set to run the main update loop every morning at 4am CET.

4.2.2 Another Possible Architectural Choice: The Separate Component Design

A different approach which was eventually rejected was to create a new external component which would:

(i) take input from CurActory (possibly using the Drupal WebAPI rather than connecting directly to it)
(ii) process it
(iii) provide a web service for exposing its results ("The RMS must be inteconnectable, via web services, for providing reputation results per selected expert per topic [...].").

This approach has some serious disadvantages:

(i) It makes components that use the RMS output invoke two web methods, one on the RMS web service to get reputation data and one on a separate web method to the Drupal Web API to collect the remaining information (or forces the duplication of CurActory data on the RMS database); currently all existing or planned components store data on and retrieve data from CurActory (directly or through the Drupal Web API) alone, so the RMS would be the anomaly defying established norm

(ii) If the RMS does not post data back to CurActory but instead stores it itself, we get into a scenario where there no longer is a single database, but two, one for the RMS and one for everything else; again this seems contrary to the overall design philosophy of the project

(iii) If the RMS does post data back to CurActory, the network overhead is not one but two data transfers that could have been avoided if the RMS core logic was placed in CurActory in the form of stored procedures and functions.

4.3 Implementation Technologies

As per the chosen architectural proposal, since CurActory is implemented as a MySQL database, MySQL Stored Procedures are being used to deliver the core RMS business logic. This provides for great flexibility and high performance due to direct access to the data the RMS needs to process both for the off-line and the on-line computations.

5 Reputation Criteria

The Reputation Criteria implemented in the first version of the Reputation Management System are listed below. For each reputation criterion, a (non-technical) description of the criterion is given, followed by classification information, a list of the sources from which raw data used in the computation originate, and information pertaining to the off-line and on-line pre-processing of CurActory data.

5.1 Criterion A - Self-evaluation

Description
It is a basic requirement of the EU Community platform that the user will be able to evaluate his/her own expertise in any number of topics. The user interface for this functionality will be provided by EurActory. Typically, users will evaluate themselves on the topics most relevant to them, i.e. they will not provide a self-assessment for each and every topic. The absence of user-provided self-evaluation on a particular topic is to be interpreted as a zero-valued self-evaluation i.e. if the user does not claim to have any expertise in a topic, the system will understand this as a self-assessment of zero expertise on that topic. Non-users for whom the Crawlers have provided data cannot have any per topic expertise self-assessments; in their case, there are two options: (i) that this criterion will not affect their reputation or (ii) that the value of the criterion will be 0. The latter option offers an advantage to users of the platform as opposed to non-users. Version 1 of the Reputation Management Component takes this route.
Classification
▪ Per-person per-topic criterion ▪ Objective
Sources
▪ EurActory
Off-line Precomputation
No pre-computation is required. Self-evaluation scores are stored by EurActory. Their range is 0 – 100.
On-line Computation (for person *P*, topic *T*)

1. Find the Person_Topic record for the key (*P,T*)
2. Return Criterion_A

5.2 Criterion B - Organization Reputation

Description

According to this reputation criterion a person receives the same reputation rating as the most reputable organisation he/she is a member or employee of.

Note: In general, the effect of the reputation of an organization affects the reputation of its members or employees, but the opposite is also true. For the purposes of this project, it was decided it is sufficient (and desirable) to focus solely on the effect the reputation of organisations has on the reputation of their members / employees and not vice versa.[4]

Classification

- Per-person criterion
- Objective

Sources

- CurActory (pre-sets for well-known organisations)

Off-line Precomputation

For each person *P:*

1. Let *M* be the maximum reputation of organisations in the table RM_Organisation for organisations found in the table Person_Position where PersonId=*P*
2. Find the RM_Person record for *P*
3. Set CiterionB=*M*

Note: In the current, first version of the RMS, organisation reputation is assigned only to well-known organisation manually added in the RM_Organisation table. In subsequent versions, the aim is to define means for automatically populating the RM_Organisation table as new, hitherto unknown companies, are found in the Person_Position table (and for periodically updating it).

[4] It is perhaps interesting to note that while a reputable person moving into a new and unknown organization may, under this assumption, not affect its reputation, it can still affect the reputation of its members/employees by means of Criterion F (i.e. network connections).

On-line Computation (for person *P*)
1. Find the RM_Person record for the key (*P*) 2. Return Criterion_B

5.3 Criterion C – Position Ranking

Description
This criterion assigns a reputation rating to a person on the basis of the level of the most high-ranking job title in the most reputable organisation he/she is associated with. Three job title ranks are recognised: junior, medium, and senior. This is a rough classification of the significance of a person within their organisation. There are many possible positions and job titles and a more fine-grade classification would be impractical, but given this classification titles containing keywords such as President, Vice-President, CEO, Director, Secretary General can be mapped to the senior level, titles not containing those words but containing words such as Senior, Manager, Consultant etc. can be mapped to the medium level and the rest to the junior level. This is the key idea of our proposal on how to deal with this criterion. Additional logic and/or regular expressions and a thorough examination of the possible job titles/positions would be needed to deal with cases such as 'Assistant to the Secretary General' or 'P.A. to the CEO' in order not to map them onto the senior but onto the junior level.
Classification
▪ Per-person criterion ▪ Objective
Sources
▪ Europa WhoIsWho ▪ LinkedIn
Off-line Precomputation
For each person *P:* 1. Let *M* be the maximum value of SeniorityToScore(SeniorityLevel(*J)*) for all job titles found in the table Person_Position where PersonId=*P* 2. Find the RM_Person record for *P* 3. Set CriterionC=*M*

SeniorityLevel(J): Determining the Seniority Level of a Job Title

Given J, a textual description of a job title:

1. If the job title matches a Junior ranking pattern, return 0
2. If the job title matches a Senior ranking pattern, return 2
3. If the job title matches a Middle ranking pattern, return 1
4. Return 0 if none of the above conditions are met

SeniorityToScore(S): Determining the Score Assigned to a Seniority Level

Given S, a job title seniority level (in the range 0 to 2):

Return the corresponding score.

Note: In future versions the mapping of seniority levels to scores could be administrator configurable (or even user configurable, in which case its calculation would need to become part of the on-line score calculation). The current, fixed, mapping is:

Seniority Level	Reputation
0	0
1	40
2	100

On-line Computation (for person *P*)

1. Find the RM_Person record for the key (*P*)
2. Return Criterion_C

5.4 Criterion D – Document assessment

Description

This criterion assigns a reputation rating on the basis of the document reputation indicators that documents produced by the person have received. For instance, if a certain expert's tweets are constantly

receiving a high number of retweets, this is very indicative of his/her reputation and influence, at least in the Twitter-realm, and possibly beyond. Someone's raw document assessment value score is computed as the sum of the adjusted values of that person's document reputation indicators. An adjusted document reputation indicator value is a document reputation indicator value multiplied by the adjustment factor corresponding to the document reputation indicator.

Document Reputation Indicator	**Adjustment Factor**
Retweet count (for tweets)	100%
Favourite count (for tweets)	70%

The number of comments a tweet (or another document) gets is not considered a reputation metric, as comments could be negative. On the other hand, Retweeting, in the Twitter community, is practically always a mark of approval and hence positive reputation. The number of times a tweet has been marked as a favourite is clearly a mark of positive reputation.

Note 1: It may be surprising that the `favourite' count, a dedicated document-reputation feature mirroring Facebook `likes', is deemed less important than the `retweet' count functionality, yet to state that one likes something carries less value than actively going out and broadcasting it; the different adjustment factors reflect that.

Note 2: Adjustment factors are currently fixed, but they could become administrator and/or user configurable.

Note 3: Criterion D is meant to be a simple straightforward criterion for incorporating document reputation as an indication of author reputation. Different ways of doing that will be considered in designing subsequent versions of the RMS, probably resulting in an additional document-related criterion.

In order, to normalize scores, a three-stage process is used, the first step being computing the individual scores, the second being normalising them by dividing them by the maximum and multiplying them by 100, and the third combining the normalised scores as a weighted sum.

Note: This normalization method can be applied to any criterion with a variable range of raw scores.

Classification

- Per-person criterion (in version 1 of the RMS)

- Objective

  Note**:** Criterion D in the 1st version of the RMS described here is not a per person per topic criterion as the relevant logic that associates documents with topics is under parallel development. It is planned to either make it topic-specific in version 2 of the Reputation Management System by only considering documents related to a specific topic when computing per person per topic reputation scores or introduce a separate topic-sensitive criterion.

Sources

- Twitter

  Note: Currently only the reputation of tweets is being considered. It is expected that the scope of Criterion D will be expanded in subsequent versions of the RMS.

Off-line Precomputation

For each person *P*:

1. Let *D1* be the sum of the retweet counts of tweets by P
2. Let *D2* be the sum of the favourite counts of tweets by P
3. Update the RM_Person record for the key *P*, setting the fields Criterion_D1=*D1,* Criterion_D2=*D2*

For each topic *T*:

1. Let *MD1* be the maximum value of Criterion_D1 in RM_Person
2. Let *MD2* be the maximum value of Criterion_D2 in RM_Person
3. Update all entries for topic *T* in RM_Person as follows: Criterion_D1 = 100×Criterion_D1/*MD1*, Criterion_D2 = 100×Criterion_D2/*MD2*

On-line Computation (for person *P*)

1. Find the RM_Person record for the key *P*
2. Return (ρ ×Criterion_D1+ φ × Criterion_D2)/(ρ+φ)

   where ρ=100%, φ=*50%*

5.5 Criterion E – Proximity Trust

Description

The Proximity Trust Criterion is not an objective reputation criterion, but rather a subjective relevance criterion that allows a user's assumed personal bias towards being interested more in results involving his/her own acquaintances to be taken into consideration by the system. It is (reasonably) assumed that the user is more interested in the documents and profile changes of his/her acquaintances, the closer these acquaintances are to him/her, as can be determined by social graph distances (LinkedIn, Twitter) or organizational charts (WhoIsWho/LinkedIn).

Different scores can be given to connections according to the proximity and the type of connection. The 1st version of the RMS employs an proximity trust point system, computed on-line, awarding respectively 35/5 points for a 1st/2nd degree connection on LinkedIn, 30 for a 'follows' relation and 5 for a `is-followed-by' relation on LinkedIn and 30 for being colleagues (under the broad definition of working for the same company or institution according to WhoIsWho or LinkedIn).

Classification

- Per-person criterion
- Subjective

Sources

- Europa WhoIsWho
- LinkedIn

Off-line Precomputation

None.

On-line Computation (for persons *P, U*)

Note: P is the person about whom the score is being computed, whereas *U* is the person for whom the score is being reported i.e. *U* is the current user. Technical detail: *U* is person id, not a user id.

RETURN

(If *P* and *U* are 1st degree connections on LinkedIn THEN 35 ELSE 0) +

(If *P* and *U* are 2st degree connections on LinkedIn THEN 5 ELSE 0) +

(If *U* follows *P* on Twitter THEN 30 ELSE 0) +

(If *P* follows *U* on Twitter THEN 5 ELSE 0) +

(If *U* and *P* are colleagues according to WhoIsWho or LinkedIn THEN

30 ELSE 0)

Note 1: As specified, the bulk of the computation of this criterion happens on-line. This will change in the following version of the RMS, which will additionally solve also another problem.

Note 2: Due to a legal constraint in the LinkedIn API Terms of Use prohibiting storage of crawled connections, unless special permission is gained no LinkedIn connections will be available for the RMS to take into consideration. This is a recently discovered issue which is expected to be resolved in the subsequent version (D3.2.2) of the present deliverable (see §9).

5.6 Criterion F – Network Value

Description

This is an objective reputation criterion that allows the system to take into consideration the influence of one's connections on his/her reputation. For instance, if a certain energy expert is followed on Twitter by over half the current MEPs, this is very indicative of his/her reputation and influence. Someone's raw network value score is computed as the sum of the adjusted reputation values of that person's network connections. An adjusted reputation value is a reputation value multiplied by the adjustment factor corresponding to the relation between the two individuals.

Relation Type	**Adjustment Factor**
Twitter 'is-followed-by' relation	100%
LinkedIn '1st degree connection' relation	50%

Note 1: The Twitter 'follows' relation is not taken into consideration in order to avoid reputation-metric manipulation by means of following of influential figures. Besides, as anyone can follow anyone, the `follows' relation does not say much about the reputation of the follower.

Note 2: The Twitter 'is-followed-by' relation is considered more important than the LinkedIn 1st degree connection both because it is asymmetric and because it depicts an interest in what the person being followed has to say, as opposed to a real-life social connection.

Note 3: Due to a legal constraint in the LinkedIn API Terms of Use prohibiting storage of crawled connections, unless special permission is

gained, no LinkedIn connections will be available for the RMS to take into consideration. This is a recently discovered issue which is expected to be resolved in the next version of the RMS specification (D3.2.2) (see §9).

Note 4: The LinkedIn 2nd, 3rd (and 4+th) degree connections (as well as common group membership connections) are taken to be possibly incidental, so only 1st degree connections are to be taken into consideration, if possible, as only they represent a conscious statement of both parties about their social connection.

Note 5: The adjustment factors could be administrator and user configurable, but in the current version they are not.

Note 6: As in the case of a two-way connection between persons A and B, the network value of A can influence the network value of B and vice versa, if the reputation metrics used in the computation are the vanilla reputation metrics incorporating this criterion, each daily reputation value update will result in higher and higher raw network values for both A and B; in order to avoid that, a reputation metric excluding Criterion F must be used. This problem will be addressed in the 2nd version of the reputation management system.

Note 7: In many Reputation Management Systems, including Klout and PageRank (a component of the synthetic Google Search RMS but possible to also examine as an autonomous RMS), there is a concept of reputation dilution. In the context of this criterion, if person *A* had a reputation *R* and followed person *B,* whereas currently this would benefit *B*'s raw score by 100%×*H,* if reputation dilution was implemented the question of how many others *A* follows would become a factor. In the simplest implementation of reputation dilution, if person *A* had a reputation *R* and followed *F* accounts, including person *B's* account*,* this would benefit *B*'s raw score by 100%×*H/F.* A reputation dilution option will be available in subsequent versions of the EU Community RMS.

As the raw network value does not necessarily fall in the range of 0 to 100, the same normalization process described for Criterion D is used for obtaining the normalised network value from the raw network value; it is the normalised network value that is the value of Criterion F.

Classification

- Per-person , per-topic criterion
- Objective

Sources

- Twitter
- LinkedIn

Off-line Precomputation
For each person *P* and each topic *T*: 1. Let *F1* be the sum of the system reputations of *P*'s Twitter followers 2. Let *F2* be the sum of the system reputations of *P*'s 1st degree LinkedIn connections 3. Update the RM_Person_Topic record for the key (*P*,*T*) setting the fields Criterion_F1=*F1,* Criterion_F2=*F2* For each topic *T*: 1. Let *MF1* be the maximum value of Criterion_F1 in RM_Person_Topic 2. Let *MF2* be the maximum value of Criterion_F2 in RM_Person_Topic 3. Update all entries for topic *T* in RM_Person_Topic as follows: Criterion_F1 = 100×Criterion_F1/*MF1*, Criterion_F2 = 100×Criterion_F2/*MF2*
On-line Computation (for person *P* and topic *T*)
1. Find the RM_Person_Topic record for the key (*P*,*T*) 2. Return (τ×Criterion_F1+ λ ×Criterion_F2)/(τ+λ) where τ=100%, λ=*50%*

5.7 Criterion G – Past Measurements

Description
Other criteria may have values that change from one day to another. If that change is drastic there will be abrupt fluctuations in a person's reputation. Criterion G allows historical reputation values to smoothen the fluctuation and reflect the intuition that a person's reputation today is not the result of the current state of affairs only (e.g. current position) but also of his/her reputation in the past. Note: Exactly which time frame interests the consortium is a matter to be discussed, but it seems that a reasonable proposal would be to take into consideration the reputation of January 1st of four years ago, of twelve months ago, of the 1st of six months ago, and of the 1st of the previous month, attach different (user/admin configurable) weights to each and take the weighted average as the value for this criterion.
Classification
▪ Per-person, per-topic criterion

- Objective

Sources

- CurActory / Reputation Management System

Off-line Precomputation

The system reputation scores are kept for the 1st day of each month for each person and topic.

For each person *P* and each topic *T:*

1. Let *G1* be the archived system reputation of *P* on *T* on January 1st of four years ago
2. Let *G2* be the archived system reputation of *P* on *T* on the 1st of twelve months ago,
3. Let *G3* be the archived system reputation of *P* on *T* on the 1st of six months ago,
4. Let *G4* be the archived system reputation of *P* on *T* on the 1st of last month
5. Update the RM_Person_Topic record for the key (*P*,*T*) setting the fields Criterion_G1=*P1,* Criterion_G2=*G2,* Criterion_G3=*G3,* Criterion_G4=*G4.*

On-line Computation (for person *P* and topic *T*)

1. Find the RM_Person_Topic record for the key (*P*,*T*)
2. Return

$$\frac{\text{Criterion\_G1} * \text{CriterionG1Weight} + \cdots + \text{CriterionG4} * \text{CriterionG4Weight}}{\text{CriterionG1Weight} + \cdots + \text{CriterionG4Weight}}$$

6 Synthesis Functions

A synthesis function is a function that takes the individual per-criterion metrics and produces a single reputation metric. It is possible for administrators to add synthesis functions of their choosing. The pre-defined synthesis functions are listed below.

6.1 Default

Description
The weighted average of the criteria scores.
User Configuration
• Weights per criterion **Note:** It is envisaged that the user will have available a user interface for de-activating the weighting of criteria. This will not be supported by the RMS in terms of an additional flag parameter to the synthesis function as the same effect can be obtained by using 1 as the weight of all criteria.
Implementation
$\frac{\text{CriterionA}*\text{CriterionAWeight}+\cdots+\text{CriterionG}*\text{CriterionGWeight}}{\text{CriterionAWeight}+\cdots+\text{CriterionGWeight}}$

6.2 Best *N* (e.g. Best 3)

Description
The weighted average of the best N weighted criteria scores. Example: Suppose the user chooses N to be 3 and disables the weighting of criteria (see User Configuration below). For person X, the 3 best criteria may be (D:40) (A:35) and (G:15) (average: 30) whereas for person Y the 3 best criteria may be (C:10) (B:20) and (D:30) (average: 50). So person Y will have a higher reputation than person X using this synthesis function even if on average person X fairs better in terms of individual per-criterion scores.
User Configuration
▪ *N* ▪ Weights per criterion **Note:** It is envisaged that the user will have available a user interface for

de-activating the weighting of criteria. This will not be supported by the RMS in terms of an additional flag parameter to the synthesis function as the same effect can be obtained by using 1 as the weight of all criteria.
Implementation
1. Map each criterion *i* to a row containing the score for Criterion *i*, the weight for Criterion *i* and the product of the score for Criterion *i* × the weight for Criterion *i* 2. Put those rows in ascending order of the third column i.e. he score for Criterion *i* × the weight for Criterion *i* 3. Calculate S the sum of the third column of *N* rows starting with row 1 4. Calculate W the sum of the second column of *N* rows starting with row 1 5. Return S / W

6.3 Worst N (e.g. Worst 3)

Description
The weighted average of the worst N weighted criteria scores.
User Configuration
▪ *N* ▪ Weights per criterion **Note:** It is envisaged that the user will have available a user interface for de-activating the weighting of criteria. This will not be supported by the RMS in terms of an additional flag parameter to the synthesis function as the same effect can be obtained by using 1 as the weight of all criteria.
Implementation
1. Map each criterion *i* to a row containing the score for Criterion *i*, the weight for Criterion *i* and the product of the score for Criterion *i* × the weight for Criterion *i* 2. Put those rows in descending order of the third column i.e. he score for Criterion *i* × the weight for Criterion *i* 3. Calculate S the sum of the third column of *N* rows starting with row 1 4. Calculate W the sum of the second column of *N* rows starting with row 1 5. Return S / W

6.4 Middle N (e.g. Middle 3)

Description

The average of the middle N weighted criteria scores.
User Configuration
▪ *N* ▪ Weights per criterion **Note:** It is envisaged that the user will have available a user interface for de-activating the weighting of criteria. This will not be supported by the RMS in terms of an additional flag parameter to the synthesis function as the same effect can be obtained by using 1 as the weight of all criteria.
Implementation
1. Map each criterion *i* to a row containing the score for Criterion *i*, the weight for Criterion *i* and the product of the score for Criterion *i* × the weight for Criterion *i* 2. Put those rows in descending order of the third column i.e. he score for Criterion *i* × the weight for Criterion *i* 3. Let *F*=floor((7-*N*)/2) + 1 4. Calculate S the sum of the third column of *N* rows starting with row *F* 5. Calculate W the sum of the second column of *N* rows starting with row *F* 6. Return S / W

7 Missing Inputs

Whereas the RMS is designed to compute a reputation score for a person that differs from topic to topic and from observer to observer (whereby observers are the users), namely a subjective per person per topic score, on occasion that specificity is not merely unnecessary but unwanted. What follows, is a specification pertaining to how the cases of having no current user (specified by passing 0 as the current user person id) or no topic (specified by passing 0 as the topic id) as inputs are dealt with by the RMS.

7.1 No Current User

The computation of the system reputation value for historical record keeping, as required by Criterion G, is topic sensitive, but not user-sensitive.

When no current user person id is specified, the value of the criterion A score is taken to be 0.

This would normally affect the range of possible scores e.g. if a weight of 10% was assigned to Criterion A by the system administrator, scores would be in the range 0 to 90. However, when no current user is supplied, the weight of Criterion A is also set to 0 (irrespective of what weight may have been passed as a parameter to the RMS scoring function) thus ensuring Criterion A is not considered.

A similar approach could also be used when no topic is specified (ignore all topic-sensitive reputation criteria). However, this would leave out important information from consideration of a person's reputation. So a different solution is used there, as described below.

7.2 No Topic

No topic is provided by EurActory when no topic has been selected by the user. If experts are to be ordered by reputation for display purposes with no topic selected, ignoring the per topic reputations is less than ideal.

Instead, the RMS supports a special dummy topic id, 0, which gets, for each expert, the maximum reputation he/she has in the other (i.e. the real) topics, for each reputation field in the RM_Person_Topic table. This is implemented as a final step in the pre-computation of reputation scores. As a result, the on-line computation need not to treat the case of no topic (topic=0) in any special manner.

8 Integration

The Reputation Management System is not a user-facing component; rather, its purpose is to provide reputation information to such components of the EU Community Platform, namely EurActory and PolicyLine in a convenient and efficient manner.

8.1 RMS-EurActory Integration

EurActory is the one user-facing component that has direct access to CurActory, the common EU Community Platform database. By the RMS's design, EurActory remains entirely agnostic to the RMS's design, data and operations except for the fact that the RMS makes available an SQL function called RM_SCORE_WITH_FULLPARAMETERS.

RM_SCORE_WITH_FULLPARAMETERS		
Input Parameter 1 Name _PersonId	**Input Parameter 1 Type** LONG	**Input Parameter 1 Documentation** This is the id of the person for which the reputation score for the topic specified by the second argument will be computed.
Input Parameter 2 Name _TopicId	**Input Parameter 2 Type** LONG	**Input Parameter 2 Documentation** This is the id of the topic for which the reputation score for the person specified by the first argument will be computed. Together, parameters 1 and 2 specify the two key inputs for the reputation scoring question in the EU Community RMS: (i) Whom are we evaluating the reputation of? (ii) On what topic are we evaluating his/her

		reputation? When a per-person rather than a per-person per-topic score is required, the value 0 must be passed to the _TopicId parameter, as discussed in §7.1.
Input Parameter 3 Name _CurrentUserPersonId	**Input Parameter 3 Type** LONG	**Input Parameter 3 Documentation** This is the id of the user on behalf of whom the reputation score is being requested. This parameter is required because the EU Community Reputation Management System is capable of taking into consideration subjective (i.e. user-sensitive) factors (such as social network proximity – see §5.5) in computing person-topic reputation scores. When requesting an objective (i.e. non user-sensitive) person-topic reputation score, the value 0 must be passed to the _UserId parameter, as discussed in § 7.2.
Input Parameter 4 Name _Synthesis	**Input Parameter 4 Type** VARCHAR(30)	**Input Parameter 4 Documentation** This parameter determines the synthesis function to be used for computing the synthetic reputation score. It may take one of the following values: ▪ '$DEFAULT' ▪ '$BEST-N' ▪ '$MIDDLE-N' ▪ '$WORST-N'

		These values correspond to the synthesis functions specified in Section 0. For more details, see the documentation of the _N_Criteria parameter.
Input Parameter 5 Name _N_Criteria	**Input Parameter 5 Type** INTEGER	**Input Parameter 5 Documentation** This parameter specified the number N i.e. the number of criteria that will be taken into consideration by the synthesis function specified via the _Synthesis parameter when the latter is not '$DEFAULT', but rather one of the following: ▪ '$BEST-N' ▪ '$MIDDLE-N' ▪ '$WORST-N' Together parameters _Synthesis and _N_Criteria specify the synthesis function to be used in determining the synthetic reputation of _Person in _Topic as perceived by _User.
Input Parameters 7-13 Names CriterionAWeight CriterionBWeight CriterionCWeight CriterionDWeight CriterionEWeight CriterionFWeight CriterionGWeight	**Input Parameters 7-13 Type** DOUBLE	**Input Parameter 7-13 Documentation** The weights to be given, respectively, to the reputation scores of Criteria A to G in determining the synthetic reputation of _Person in _Topic as perceived by _User in accordance with the synthesis function specified by _Synthesis and _N_Criteria. Example: If for the given

		_Person the reputation score according to Criterion A in _Topic as perceived by _User is 80 and _CriterionAWeight is 0.1, then the Criterion A score that will be considered by the synthesis function will be 80×0.1=8. Note: While there is no algorithmic requirement for weights to fall in any particular range, it is implicitly assumed throughout this document that they will be in the range 0 to 1. This ensures that since every individual criterion score is guaranteed to be in the range 0 to 100, the synthetic score will also be in the same range.
	Output Type DOUBLE	**Output Documentation** The synthetic reputation score of _Person in _Topic as perceived by _User in accordance with the synthesis function specified by _Synthesis and _N_Criteria, taking into consideration the _CriterionAWeight … _CriterionGWeight weight parameters.

8.2 RMS-PolicyLine Integration

In accordance with the project plan, no version of PolicyLine is available for integration work with the RMS to begin at the time the present deliverable is being submitted. Details of integration with PolicyLine of versions 2 and 3 of the RMS will be provided in D3.2.2 and D3.2.3, respectively.

9 Conclusions & Future Work

The primary aim of the first version of the Reputation Management System was to provide a basis for integration with the various currently available components of the EU Community platform. In that respect, getting the architecture right and creating appropriate interoperability mechanisms and interfaces were the two key objectives for the first version of the RMS.

Both those two key objectives have been met. The decision to implement the RMS within CurActory as a set of stored procedures and functions is optimal both in terms of system performance and of integration effort, as:

1. it eliminates the need to have primary data transferred to the RMS (from CurActory and/or directly from the Crawlers Component and the Opinion Mining Component) and reputation scores sent back to CurActory (from the RMS)

2. it eliminates the need to provide and optimise the respective interoperability mechanisms

3. it eliminates the need of other components such as EurActory and PolicyLine to be gathering information from an additional component; instead the EurActory/CurActory Web API can serve as the interoperability mechanism of the RMS also, providing additional reputation information in an easy-to-consume manner without requiring the consuming components to be performing the joining of the respective information.

At the same time, at the level of data representation and data processing specifications, a solution has been outlined in the present document (and is implemented in the RMS software) that allows two opposing requirements to be satisfied, namely the requirement that users do not notice any delays due to reputation calculations (which is possible if reputation scores are computed off-line on, say, a daily basis) and the requirement that users be able to customise the reputation calculation (which is not possible if reputation scores are computed off-line). The key two ingredients of the solution are: (i) a careful design of reputation scoring customisability and (ii) a division of labour in reputation scoring between an off-line pre-computation stage and an on-line light-weight computation stage utilising both reputation data prepared off-line and a user's reputation scoring preferences.

Moreover the first version of the RMS has managed to go well beyond its original targets with regards to coverage of the original requirements, as the full set of required reputation criteria have been implemented, albeit with room for improvement in certain cases (see below).

This document presents work in progress. The next deliverable in the series, D3.2.2, describing the corresponding, second, version of the RMS, will, at the very least:

- Reflect any developments in the Consortium's understanding of how to best capture aspects of reputation relevant to the project in terms of reputation criteria

- Specify an improved calculation for Criteria B (Organisation Reputation) and C (Position Ranking)[5].
- Specify either a per person per topic version of Criterion D or another document-related per person per topic criterion, making use of the document categorisation functionality of components currently under development.
- Provide an option for reputation dilution (as discussed in the specification of Criterion F, § 5.6).
- Move part of the pre-computation of LinkedIn-related criteria to the Crawlers thus bypassing the legal limitation that LinkedIn relations cannot be stored.
- Specify a mechanism whereby a full analysis of the computed reputation score will be prepared in the form of an HTML string which can be used by EurActory and/or PolicyLine for transparency purposes.
- Assuming the relevant user-interfaces are in place in the respective components, provide details not of the envisaged (Figure 5) but of the actual user-interface customisation options for reputation management in EurActory and PolicyLine.
- Specify the Web API for integration with PolicyLine, a component that is at the time of writing under development.

[5] Individual improvements aside, there needs to also be a way of ensuring a closer integration of the two criteria to prevent a certain problem the current implementation has when dealing with people with multiple positions in multiple organisations. The problem is that for both Criterion B and for Criterion C the result is the maximum reputation that can be gained from, respectively, the organisations and from the positions the person is associated with. So a person who holds a junior position in highly reputable organisation (Criterion B reputation 100) and a senior position (Criterion C reputation 100) in an unimportant organisation will be awarded the highest possible marks for both Criteria B and C (as if he/she held a senior position in the reputable organisation). The details of how this is to be achieved have been worked out, but have not been implemented in the current version of the RMS.