Mastering Data-Intensive Collaboration and Decision Making

FP7 - Information and Communication Technologies

Grant Agreement no: 257184
Collaborative Project
Project start: 1 September 2010, Duration: 36 months

D4.2.2 – The Dicode Decision Making Support Services (enhanced version)

Due date of deliverable: 31 August 2012
Actual submission date: 31 August 2012
Responsible Partner: CTI
Contributing Partners: CTI, FHG, UOL

Nature: ☒ Report ☒ Prototype ☐ Demonstrator ☐ Other

Dissemination Level:
☒ PU : Public
☐ PP : Restricted to other programme participants (including the Commission Services)
☐ RE : Restricted to a group specified by the consortium (including the Commission Services)
☐ CO : Confidential, only for members of the consortium (including the Commission Services)

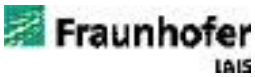
Keyword List: Decision making support services, recommendation mechanism, semantic-driven collaboration monitoring mechanism, decision making support, abstract service description, interfaces, operations, development, multi-criteria decision making

The Dicode project (dicode-project.eu) is funded by the European Commission, Information Society and Media Directorate General, under the FP7 Cooperation programme (ICT/SO4.3: Intelligent Information Management).

The Dicode Consortium

Computer Technology Institute & Press "Diophantus" (CTI) (coordinator), Greece

University of Leeds(UOL), UK



Fraunhofer-Gesellschaft zur Foerderung der angewandten Forschung e.V. (FHG), Germany

Universidad Politécnica De Madrid(UPM), Spain

Neofonie GmbH(NEO), Germany

Image Analysis Limited(IMA), UK

Biomedical Research Foundation, Academy of Athens(BRF), Greece

Publicis Frankfurt Zweigniederlassung der PWW GmbH (PUB), Germany

Document history			
Version	Date	Status	Modifications made by
1	02-08-2012	First draft version	Manolis Tzagarakis (CTI) Spyros Christodoulou (CTI)
2	10-08-2012	Second draft version	Natalja Friesen (FHG) Spyros Christodoulou (CTI)
3	14-08-2012	Third draft version (sent for internal review)	Fan Yang-Turner (UOL) Manolis Tzagarakis (CTI) Nikos Karacapilidis (CTI)
4	17-08-2012	Internal review	Max Jakob (NEO) Jörg Kindermann (FHG)
5	24-08-2012	Reviewer's comments incorporated (version sent to the Dicode SC)	Manolis Tzagarakis (CTI)
6	31-08-2012	Final version (approved by SC, sent to Project Officer)	Manolis Tzagarakis (CTI) Nikos Karacapilidis (CTI)

Deliverable manager

- Manolis Tzagarakis, CTI

List of Contributors

- Spyros Christodoulou, CTI
- Manolis Tzagarakis, CTI
- Nikos Karacapilidis, CTI
- Natalja Friesen, FHG
- Fan Yang-Turner, UOL

List of Evaluators

- Jörg Kindermann, FHG
- Max Jakob, NEO

Summary

This deliverable is a progress report on the enhanced version of the Dicode decision making support services, which are designed and implemented in the context of WP4. Decision making support services developed in Dicode concern the building of machine-interpretable knowledge in order to actively support various decision making tasks. In this deliverable, the technical specifications of the decision making support services being developed in the context of Tasks 4.2, 4.3 and 4.5 are presented. The intended audience of this document are designers and developers of the Dicode project. The document informs them on which decision making support services have been developed or updated (with respect to the initial version of the decision making support services that have been described in deliverable D4.2.1) and how they can be used. Future steps towards the full development of the envisioned services are also discussed. The enhanced versions of the decision making support services are presented using a formalized and project-wide adopted service description template.

Table of Contents

1 Introduction

1.1 Context

This deliverable presents the enhanced version of the decision making support services that have been designed and developed in the context of WP4 ("Data-Intensive Collaboration & Decision Support Services") of the Dicode project. Specifically, it reports on the progress of work being carried out in Tasks 4.2, 4.3 and 4.5, by describing the current (enhanced) version of the associated services.

Deliverable D4.2.2 is the second of a series of three deliverables reporting on the progress of work related to the development of decision making support services in the context of WP4. While the focus of D4.2.1 was on the initial version of these services, this deliverable reports on their enhanced version. The final version of the decision making support services will be reported in deliverable D4.2.3 (due in month 33).

1.2 Objectives

The purpose of this document is to present the enhanced version of the developed decision making support services, as they originated from the functional specifications outlined in deliverable D2.2 ("The Dicode Approach – User requirements, conceptual integrative architecture, agile methodology and functional specifications"), and the updated Dicode approach described in deliverable D2.3 ("The Dicode Approach Revisited").

The developed services are presented from a technical perspective, broken down to the level of individual operations, in order to make clear their role and use and facilitate their assessment with respect to the derived functional specifications. The operations presented are those which are available to clients to be invoked and executed without going into detail about how exactly these can be invoked or executed. In particular, the presented operations can be executed by various technologies such as REST (Fielding, 2000) or Web Services (W3C - Web Services Architecture, 2004), but such issues are not in the focus of their description. The description of services takes an operation-oriented approach listing the available operations and detailing their aim and purpose.

The enhanced versions of the services are presented using the service description template, which has been derived and used in the context of deliverables D3.1.1 ("The Dicode Data Mining Framework")) and D4.2.1 ("The Dicode Decision Making Support Services (initial version)"), called the Abstract Service Description. The Abstract Service Description template provides a technical specification of services by providing an overview of the supported interfaces and the relevant operations. For each operation, a description together with major input and output information is presented.

2 Dicode Decision Making Support Services

The aim of the decision making support services is to turn information and knowledge machine interpretable, thus allowing the active participation of the system in collaborative activities, and ultimately aiding the overall decision making process. In the context of Dicode, there are three different ways to generate knowledge for the decision making process:

- Automatically generating knowledge about the preferences of a single user, which can for example be used to recommend documents the user might be interested in.
- Automatically generating knowledge about the interaction of a whole group of users, e.g. by appropriately structuring an ongoing discussion and identifying related discussions.
- Allowing users to actively participate in the knowledge generation process, which will allow to fully capture and describe all knowledge that is relevant in a decision making process.

Towards these overall goals, a number of relevant services are being developed that can fully address the relevant requirements of Dicode use cases, as outlined in deliverables D2.2 and D2.3.

This document reports on the enhanced version of the services that have been developed in the context of the following tasks:

- Task 4.2: Recommendation mechanism, which is concerned with the design and development of recommender mechanisms to support the user in the analysis of large and heterogeneous data;
- Task 4.3: Semantics-driven collaboration monitoring mechanism, which will derive a semantics-driven model of collaborative processes in a community;
- Task 4.5: Decision making support services, which are concerned with the formalization of collaborative decision making issues to intelligently support stakeholders in such activities.

In the following sections, we present the service description of the enhanced version of the services being developed in the abovementioned tasks of WP4. For each service, a short description related to its aim and purpose is given, followed by the Abstract Service Description template. Only services that have been updated or added, in comparison to the initial version of the decision making support services (described in deliverable D4.2.1), are presented.

2.1 Overview of changes in the enhanced version of collaboration services

The following table gives an overview of changes made in the enhanced version of the decision making support services, which will be presented in detail in the rest of this deliverable. The value "Updated" in the "Status" column means that the data type, interface or operation has been updated (compared to the initial version reported in D4.2.1), while the value "New" means that a data type, interface or operation has been newly introduced in the enhanced version of the decision making support services.

Dicode Service	Interface Method	Status
Recommender Service		
	Recommender Service	Updated
	Query	Updated
Cluster Discussion Threads		
	ThreadsClusterer	Updated
	readDiscussions	Updated
Identify Discussion Forum Topics		
	TopicsIdentifier	Updated
	generateTopicClouds	New

Collaboration Service		
	MultiCriteriaDecisionMaking	New
	setWSMWeights	New
	calcWSMScores	New
	showWSMPlot	New
	setAHPParameters	New
	calcAHPScores	New
	showAHPPlot	New
	setLexicographicParameters	New
	calcLexicographicScores	New
	showLexicographicPlot	New

2.2 Recommendation mechanism

Task 4.2 concerns recommender mechanisms that intend to facilitate the reuse, retrieval and exchange of scientific data such as research objects, datasets or experiment plans. Deliverable 4.2.1 presented an initial version of the Recommender Service which is based on similarity learning technique. The recommendation in a scientific context is significantly different from the standard case of simple product recommendation. Many common techniques, such as collaborative filtering, content based filtering and knowledge-based cannot be applied for the domain of scientific items (Friesen & Rüping, 2011). The biggest issues are the representation of complex objects, a smaller heterogeneous set of users and a lack of information about the user's preferences. Similarity learning addresses these problems by providing the following features:

- Generality: can be applied to a wide range of items from different application domains including complex objects, such as Dicode items, e.g. datasets, or medical reports. Similarity learning does not require that the instances are represented in some feature space. It only assumes that meaningful local distances can be defined adequately.
- Ability to provide personalized recommendation according to the user needs. It is one of the most promising techniques in cases where the notion of the similarity is user-dependent. It enables to learn a user specific similarity function.

We refer to Deliverable D3.2.2 ("The Dicode Data Mining Services (enhanced version)") for more details.

2.2.1 Recommender Service

A prototype of this service has been implemented and described in D4.2.1 (for more details refer to D3.2.1: "The Dicode Data Mining Services (initial version)"). The service intends to provide a user with new and relevant items that are similar to the one the user is interested in. In the first step, the service computes a similarity between a reference item and all available items. In the second step, the algorithm orders all items according to their similarity and returns the k most similar ones. The computation of similarities is based on a similarity model, which is learned using the similarity service presented in D3.2.1.

The initial versions of the Recommender and the Similarity Learning Services have been extended as follows:

- In the Recommender Service, we have integrated a component for creation of an efficient data structure. This significantly reduces computation time.
- In the Similarity Learning Service, we integrated an approach that enables learning an accurate model with small labeling costs. We developed an intelligent sampling strategy that selects the most 'interesting' pairs from a pool of unlabeled data to show them to the user.

The updated description of the Dicode Recommender Service is presented below.

Name	*Recommender Service (**Updated**)*
Standards	See deliverable D4.2.1
Description	All details about the basic functionalities, described in Deliverable D4.2.1, are still valid. Additionally, the service has been extended by a component for creation of an efficient data structure that intends to significantly reduce time of search for similar items.
Interface	*ServiceCapabilities*
getCapabilities	See deliverable D4.2.1
Interface	*Recommendation (**Updated**)*
*Query (**Updated**)*	Computation time is reduced due to the integrated algorithm for building an efficient data structure.
feedback	See deliverable D4.2.1
Example usage	See deliverable D4.2.1
Comments	We demonstrate the feasibility of the service in the example of Recommender System for Gene Expression Omnibus (GEO) datasets, presented in D3.2.2 ("The Dicode Data Mining Services (enhanced version)").
Conformance classes	As described in Dicode deliverable D4.2.1
Implementation rules	As described in Dicode deliverable D4.2.1
Implementation status	The algorithm is yet provided as a service and can be used as a general framework for recommender tasks.
UML model	Not available.

2.3 Semantic-driven collaboration monitoring mechanism

The Community Modelling and User Profiling components comprise a set of services that address the challenges of the semantics-driven collaboration monitoring mechanism task (Task 4.3) of WP4.

Subsequently, an abstract, coarse description of the Community Modelling and User Profiling Services is provided in textual format according to Dicode Service Description Template. The description presents the basic functionality (in terms of interfaces and operations) of each of the services whose prototypical version has been implemented. These "abstract service descriptions" are meant as a high level introduction to the Community Modelling and User Profiling Services and will be complemented by formal abstract specifications and/or implementation specifications when the services become operational. In addition to the abstract service description for each of the services, we also report briefly on an experiment which has been carried out in order to demonstrate the current version of the services described in this section.

The Community Modelling and User Profiling services have been updated to reflect the knowledge gained from sensemaking models. Searching for relevant information is one of the important sensemaking operations that can be seen in online discussion forums. The literature has identified many problems in sensemaking support (see D2.3: "The Dicode approach revisited").

Although many sensemaking support tools have been designed for both individuals and groups to search over document collections and the web, these tools still have very little contribution to enable the users to make sense of the vast amounts of information they encounter every day. In many situations, little sensemaking support is observed from such tools. In particular, there is still lack of support to help users who participate in a discussion forum to:

- identify the discussions that are truly relevant to their interests or needs, as well as to search for the information they seek within only these relevant discussions instead of searching within the whole discussion forum.
- refine the discussions found in the forum by keeping only those discussions that are relevant to their interests or needs.
- abstract or summarize the discussions by visualizing the main topics (or main theme) for each group of similar discussions threads in the forum.

In addition to the above issues, there are no existing and well known sensemaking-support tools that exploit machine learning and data mining techniques to support the several sensemaking operations, such as *summarizing* and *linking*.

To support the user sensemaking (summarizing and linking) in data-intensive online collaborative spaces, a Forum Summarization Service (FSS) is developed to help users identify the popular topics and find relationships among those topics (with FSS we refer to the set of functionalities provided by the "Cluster Discussion Threads" service and the "Identify Discussion Forum Topics" service). In summary, the updates of this service since the previous related deliverable (D4.2.1) are:

- A pure black-box Java-based version of the Forum Summarization Service is implemented, which runs without the need of using the RapidMiner platform.

- An input data mechanism has been implemented to take standard format of discussion threads of forums as input to the Forum Summarization Service.
- Two Internet Forums have been selected from stackexchange.com as input to the service for the user trial of different use cases. One is statistical analysis (for Use Case 1) and the other is web applications (for Use Case 2 or Use Case 3).
- A sample lightweight front-end interface website has been developed in PHP to visualize the topic clouds and browse the discussions using the topics in the clouds.
- The Forum Summarization Service has been integrated into the Dicode workbench following the guideline provided in D5.1.1: "Standards and guidelines for development".

2.3.1 Cluster Discussion Threads

The "Cluster Discussion Threads" service is aimed to support discussion-related recommendations, by taking as input the community discussions that exist in the "forum view" of the Dicode collaboration space (See Section 7.4 of deliverable D2.2) and grouping them into distinct clusters based on their textual content. The service also produces the centroid of each cluster, which is a vector of the average term weights that exist in each cluster. The output of the service (assignment of discussions to clusters and cluster centroids) represents the foundation for a content-based recommendation service that compares the user characteristics with cluster characteristics based on the cluster centroids and then recommends to the user the discussions that belong to the cluster whose characteristics are the closest to the user characteristics.

The abstract description of this service is given below:

Name	Cluster Discussion Threads
Standards	
Description	The Cluster Discussion Threads service takes a collection of discussion threads and clusters the threads into groups, where each group contains a subset of the original threads. The discussion threads that belong to a particular group are relatively similar in content. The Cluster Discussion Threads service provides its functionality through the *ThreadsClusterer* interface. The purpose of this interface is to receive a collection of discussion threads as input, perform text pre-processing on the received threads, build and train a clustering model, generate clusters of discussion threads, assign each of the discussion threads to one of the generated clusters, and retrieve the centroid vector for each generated cluster.
Interface	ThreadsClusterer (***Updated***)
readDiscussions (***Updated***)	The *readDiscussions* operation reads the dataset from a storage resource. In the prototypical version of the service, the storage resource is an Excel sheet containing the required attributes of the dataset. In the updated version, a MySQL database is used to store the original discussion threads to enable end users to view the generated topic clouds in a

	graphical way.
dataToDocuments	See deliverable D4.2.1
processDocuments	See deliverable D4.2.1
trainModel	See deliverable D4.2.1
validateModel	See deliverable D4.2.1
clusterThreads	See deliverable D4.2.1
getCentroids	See deliverable D4.2.1
Example usage	When the users of the collaborative workspace log on into the discussion "forum view", they may want to view other discussion threads that meet their interests or needs. The Cluster Discussion Threads service groups the existing discussion threads into clusters (groups) based on their content. A recommendation service can then compute a similarity score between the user preferences vector and the centroid vector of each discussion thread group. Subsequently, the service can then recommend to the user to read and participate in the discussion threads that belong to the group whose centroid has the highest similarity score with the user preferences.
Comments	The user does not interact with the Cluster Discussion Threads service directly (it constitutes a support service (internal mechanism)). The purpose of this service is to group similar discussion threads found in a discussion forum into distinct groups. The discussion forums that belong to a particular cluster are relatively similar in content. **Current restrictions**: There are no available discussion threads in the Dicode "forum view" by the Use Case users (Use Case 1 and 2). Therefore, the current prototypical version of the service is demonstrated using discussions collected from an online technical discussion forum. See Appendix A in deliverable D4.2.1 for a detailed demonstration of the service. **Future Extensions and Enhancements**: Future versions of the service will be applied to the discussion threads made by the users of Use Case 1 and 2 within the "forum view" of the Dicode multi-view workspace approach. Future versions will also apply different clustering models based on different clustering algorithms (e.g. K-means, Hierarchical clustering (Abonyi and Feil, 2007)). The different models will be compared and the model that derives the best clusters (based on cluster validity evaluation techniques) will be selected for the final implementation. Feature extraction pre-processing will be performed before generating the clusters using Latent Semantic Analysis (LSA) (Landauer and Dumais,

	1997) to improve the clustering efficiency and the centroids of the generated clusters.
Conformance classes	Not available.
Implementation rules	Not available.
Implementation status	Prototypical version implemented.
UML model	Not available.

2.3.2 Identify Discussion Forum Topics

The "Identify Discussion Forum Topics" service is aimed to enable a user belonging to a Dicode community to make more sense of the discussions that exist in the "forum view" of the Dicode collaboration space by taking a cluster of discussions generated by the "Cluster Discussion Threads" service (described in Section 2.3.1) as an input and produces a list of identified topics with their weights, which the discussions in that cluster are about. The output of the service (topics and topic weights) is expected to help the users of the forum to make more sense of the existing discussions. By showing the users a "topic cloud" that depicts the identified topics in different font sizes (which are proportional to the computed topic weights), the user is expected to identify the main theme of the discussion in that cluster. Therefore, the user will be able to focus his reading and participation in the discussions that belong to the cluster that he identified, based on the identified topics, as the most suitable to his interests.

The following abstract description of the service is given:

Name	Identify Discussion Forum Topics
Standards	
Description	The Identify Discussion Forum Topics service takes a cluster of discussion threads as input from the Cluster Discussion Threads service and identifies the most prominent terms (topics) in the discussion threads that belong to that cluster. The identified topics can be used by the users to identify the main theme the discussion threads in that cluster are about. The service provides its functionality through the *TopicsIdentifier* interface. The purpose of this interface is to receive a collection of discussion threads that belong to a particular cluster, perform text pre-processing on the received threads, and identify the topics of these threads.

Interface	TopicsIdentifier (***Updated***)
readDiscussions	See deliverable D4.2.1
preprocessThreads	See deliverable D4.2.1
identifyTopics	See deliverable D4.2.1
generateTopicClouds ***(New)***	The *generateTopicClouds* operation generates an HTML output file containing a collection of Topic Clouds, where each cloud corresponds to one of the generated discussion clusters. Each cloud depicts (i) the theme of discussion for the discussions that belong to that cluster, (ii) the number and percentage of discussions that belong to that cluster, and (iii) 10 topics having the highest 10 weights for the discussions that belong to that cluster.
Example usage	When the users of the collaboration workspace log on to the "forum view" and focus on the discussion threads that belong to a particular cluster (i.e. the cluster of discussions that is closest in content to their preferences), they may want to summarize these discussion threads or identify the main theme that these discussion threads are about. Invoking the Identify Discussion Forum Topics service identifies the main topics these discussion threads have. A visualization service can take the identified topics and their frequencies and show to the users a "topic cloud" that depicts the identified topics in different font sizes proportional to the frequency of occurrence of these topics. The user will be able to identify how important each identified topic is to the discussion threads based on the font size of that topic, and therefore be able to summarize these discussions and identify the main theme of the discussions. The user may also be able to interact with the identified topics through another visualization service that links each identified topic with a page that shows a list of the discussion threads that contain discussions mainly related to that particular topic. The user can click on the topic link to browse the discussions related to that topic.
Comments	The user does not interact with this service directly. This is a support service (internal mechanism) where the purpose of this service is to identify the most prominent terms (topics) in the discussion threads that belong to a particular cluster of discussion threads. **Current restrictions**: There are no available discussion threads in the Dicode Discussion Forum view by the Use Case users (Use Case 1 and 2). Therefore, the prototypical version of the service is demonstrated using discussions collected from an online technical discussion forum. See Appendix B in deliverable D4.2.1 for the detailed demonstration of the service. **Future Extensions and Enhancements**: Future versions of the service will

	be applied on the discussion threads made by the users of Use Cases 1 and 2 within the "forum view" of the Dicode collaboration workspaces. Future versions of the service will implement more advanced models – than the frequency of occurrence model currently adopted - to support topic identification and topic modeling (e.g. models based on the Latent Dirichlet Allocation (LDA) algorithm (Blei et al., 2003)). The aim is to compare these models and adopt the model that derives the best topics (based on human evaluation) for the final implementation.
Conformance classes	Not available.
Implementation rules	Not available.
Implementation status	Prototypical version implemented.
UML model	Not available.

2.3.3 Input Data

To simulate Dicode use cases, we selected two forums from stackexchange.com, a fast-growing family of forum web sites, which have over 1.5 million users on diverse topics from programming to cooking. The selected forums are: (i) Web application forum - to simulate use case 3: data analysts in consultant company, and (ii) Statistical analysis forum - to simulate use case 1 researchers using statistics in their work. These two forums are technically comparable in terms of their forum features[1] summarized below:

	Questions	Answers	Questions Answered	Users	Visits/day	Questions sampled
Web application	6.8k	12k	89%	13k	12k	5728
Statistical analysis	8k	14k	83%	7.5k	6.7k	4854

Table 2: Forum Selection for Forum Summarization Service Evaluation

2.3.4 User Interface of Service

A sample lightweight front-end interface website has been developed in PHP to visualize the topic clouds and browse the discussions using the topics in the clouds. The interfaces can be accessed from the following URLs:

- Cross Validated Statistics Forum, available at http://imash.leeds.ac.uk/dicode/wp4/Stats2-Forum/ (for Use Case 1)

[1] http://stackexchange.com/sites

- Web Applications Forum, available at http://imash.leeds.ac.uk/dicode/wp4/Webapps2-Forum/ (for Use Case 2 or Use Case 3)

To provide a visual output of the Forum Summarization Service (FSS), we have chosen to present the clustered groups and identified topics in a list of "Topic Clouds" (tag/word cloud, a weighted word list) as this format is useful for quickly perceiving the most prominent terms and to determine its relative prominence.[2] The output of FSS includes 10 groups of topics, each of which is presented in a topic cloud, showing the most frequent terms (topics) mentioned in the text content for that group. Each topic cloud has a title, which shows the most discussed topic in that group, other topics listed in the word cloud are presented in different font sizes, which reflect the number of discussions.

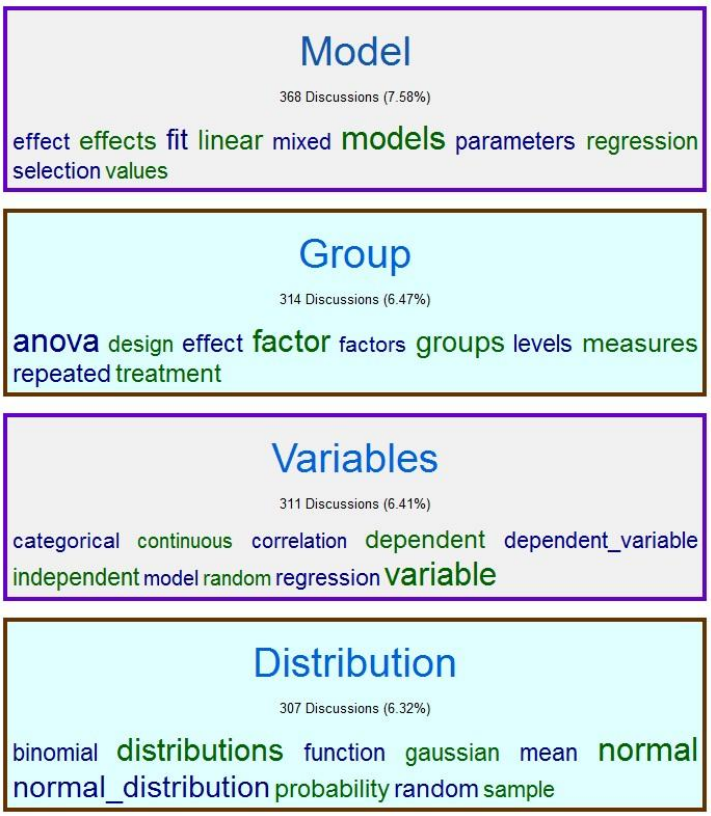


Figure 1: Topic Clouds of Forum Summarization Service

For example, in Figure 1, the four groups of topics are the output from the statistical analysis forum "Cross Validated". The topic "Distribution" was extracted from 307 discussions and related topics are shown in this cloud, such as "normal", "probability" and "sample" etc. By clicking on a topic in the topic cloud (group title or words in the group), users can see all discussions from the group that include this word. By clicking on the title of the question, users can read the full detail of the question: the question and its answers, if any.

This user interface has been integrated into the Dicode workbench following the guidelines provided in D5.1.1: "Standards and guidelines for development (initial version)".

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

2.4 Decision making support services

Dicode's decision making support services is the focus of Task 4.5; they aim towards a meaningful formalization of the collaboration to intelligently support stakeholders in decision making activities by enabling the use and exploitation of various reasoning mechanisms.

Based on the functional specifications outlined in deliverable D2.2, a set of operations supporting collaboration within Dicode have been designed and developed. In particular, in the initial version of the decision making support services, the "formal view" of collaboration workspaces has been implemented as outlined in deliverable D4.2.1. In the enhanced version of the decision making support services, apart from the "formal view" of the Collaboration Service, the support for the decision making process has been augmented with algorithms coming from the field of Multi-Criteria Decision Making (MCDM or Multi-attribute decision making - Triantaphyllou et al., 1999).

Elementary concepts of the MCDM algorithms are:

- The *alternatives* which refer to the different choices a decision maker has (actually, all the possible solutions to the problem under consideration), and
- The *attributes* that represent the different dimensions from which the alternatives may be viewed.

The main objective of a MCDM algorithm is, based on the attributes of each alternative, to calculate a corresponding score. An alternative with a higher score than another one is considered to be a better solution to the problem under consideration. The output of the MCDM algorithm is a ranked list of all the alternatives with respect to their (calculated by the algorithm) scores.

In Dicode, in the enhanced version of the decision making support services, three MCDM algorithms have been implemented. In order to calculate the alternatives' scores, these methods use the attributes of the collaboration objects in the "mind-map view" as input. More details about the implementation and application of the MCDM algorithms, as well as the attributes of the collaboration objects used in our approach, are included in Appendix A.

2.4.1 Decision making support

The enhanced version of the decision making support services, apart from providing the necessary operations to implement the "formal view", implements the "multi-criteria decision making view" of the collaboration workspaces to offer the Dicode user additional decision making algorithms.

As already stated in D4.2.1, the "formal view" of a collaboration workspace permits only a limited set of discourse moves for a limited set of message types whose semantics is fixed and system defined. In addition, this view can be associated with reasoning algorithms that are able to calculate which proposed solution is currently prevailing or which position has been defeated.

The "multi-criteria decision making view" of collaboration workspaces is the fourth option (the other three are the "forum view", the "mind-map view" and the "formal view") in which collaboration workspaces can be operated. Concerning the collaboration items, it is a read-only view (in the sense that it does not allow addition or updating of items) and its main purpose is to support the decision making process based on the attributes of the collaboration items coming from the "mind-map view". For the time being, three multi-criteria decision making algorithms (more to be added in the final version of the decision making support services) have been implemented, namely: (i) the Weighted Sum Model – WSM (Triantaphyllou et al., 1999), (ii) the Analytical Hierarchy Processing – AHP (Coyle; Saaty ,1977, 1980, 1983), and (iii) the Lexicographic Decision Making rule (Campbell et al., 2006) (refer to Appendix A for details on the theoretical background of the algorithms and their implementation in Dicode).

Since all the abovementioned views constitute an integral part of the collaboration workspace concept, the decision making support operations have been implemented within the Collaboration Service, which has been initially presented in deliverables D4.1.1 and D4.2.1. Two different interfaces provide the relevant decision making operations, when the workspace is operated in the "formal view" and the "multi-criteria decision making view".

In the table below, we present the associated changes in the enhanced version of the Collaboration Service. We annotate each change with "New" or "Updated" to indicate the type of change that occurred in the enhanced version (compared to the initial one). For completeness, we have also included the interfaces and operations reported in the initial version of the service, without their description which can be found in deliverable D4.2.1.

Name	Collaboration Service
Standards	REST (Fielding, 2000)
Description	Apart from what is described in deliverable D4.1.2, the Collaboration Service also provides operations that enable users to conduct formal argumentative discourses and multi-criteria decision making analysis. Formal argumentative discourse is provided by the "formal view" of collaboration workspaces. The service provides operations to allow users configure the "formal view" and use them to engage into collaborative activities towards decision making. The "multi-criteria decision making view" incorporates a set of three algorithms to further augment the decision making process. The algorithms take as input the alternative collaboration objects (as these appear in the "mind-map view") together with their respective attributes, and calculate a score for each alternative. The output of these algorithms is a ranked list of the alternatives with respect to their scores. The alternative with the highest score is

	considered to be the best solution. The Collaboration Service provides functionality for supporting the "formal view" and the "multi-criteria decision making view" of collaboration workspaces through the following interfaces: • *FormalCollaboration Interface:* Its purpose is to provide all operations related to the *configuration* and use of the "formal view" of workspaces. • *MultiCriteriaDecisionMaking Interface:* Its purpose is to provide multi-criteria decision making algorithms to augment the decision making process. As in the case of the *FormalCollaboration Interface*, the *MultiCriteria-DecisionMaking Interface* does not provide operations for creating and managing collaboration workspaces, as such tasks are handled through the *Workspace Interface* that has been described in deliverable D4.1.1.
Interface	*FormalCollaboration*
postIssue	See deliverable D4.2.1
postAlternative	
postPosition	
addPreference	
setReasoningEngine	
Interface	*MultiCriteriaDecisionMaking (**New**)*
*setWSMWeights (**New**)*	Used to set the weights of the four factors/criteria (number of Likes and Dislikes, Creator Rating, number of relationships in favour/against, item's rating) used in the WSM algorithm. These weights are used by the *calcWSMScores operation*.
*calcWSMScores (**New**)*	Calculates the score of each alternative using the WSM algorithm (based on an alternative's performance with respect to the four factors/criteria). Takes as inputs the workspace ID, the alternative collaboration objects (and their attributes) from the "mind-map view" and the output of the *setWSMWeights* method. For each alternative, the method calculates its corresponding score based on the alternative's attributes and produces a ranked list of the alternatives based on their scores.
*showWSMPlot (**New**)*	Used to create the bar chart with the alternatives scores. Takes as input the ranked list of the alternatives (with their respective scores-calculated by the *calcWSMScores method*) and produces a bar chart of the alternatives scores in ascending order. Alternatives with the top 20 scores appear on the bar chart.
*setAHPParameters (**New**)*	Used to set the parameters of the AHP algorithm. Takes as inputs the workspace ID, the alternative collaboration objects (and their

	attributes) from the "mind-map view" and the four factors/criteria (number of likes/dislikes, Creator Rating, number of relationships in favour/against, Item rating) used for the Decision Making. It provides all the necessary structures for the user to define the relative importance of one factor over another and the relative importance of one alternative over another (with respect to each one of the four factors). The output of the method is a data structure with all the relative importance weights among each pair of the factors and each pair of the alternatives (to be used as input to the *calcAHPScores).*
calcAHPScores ***(New)***	Calculates the score of each alternative using the AHP algorithm. Takes as inputs the relative importance weights from the *setAHPParameters* method and produces a ranked list of the alternatives based on their scores.
showAHPPlot ***(New)***	Used to create a bar chart with the alternatives scores. Takes as input the ranked list of the alternatives (with their respective scores-calculated by the *calcAHPScores method*) and outputs a bar chart of the alternatives scores in ascending order. Alternatives with the top 20 scores appear on the bar chart.
setLexicographicParameters ***(New)***	Used to order, in terms of importance, the four factors (number of likes/dislikes, Creator Rating, number of relationships in favour/against, item's rating) used in the decision making process. Takes as input the four factors and provides all the necessary structures for the user to order them in descending order of importance. The output of the method is the ordered list of the four factors (to be used as input to the *calcLexicographicScores operation).*
calcLexicographicScores ***(New)***	Calculates the score of each alternative using the Lexicographic decision making rule. Takes as inputs the ordered list of the four factors in terms of importance (set by the *setLexicographicParameters* operation), the workspace ID and the alternative collaboration objects (and their attributes) from the "mind-map view". It outputs a ranked list of the alternatives with respect to their scores.
showLexicographicPlot ***(New)***	Used to create a bar chart with the alternatives scores. Takes as input the sorted list of the alternatives (output of the *calcLexicographicScores method*) and creates as output a bar chart of the alternatives scores in ascending order. Alternatives with the top 20 scores appear on the bar chart.
Example usage (***Updated***)	A web application will provide the necessary user interface through which the previously mentioned operations can be executed by end users. In general, all Dicode use cases that require support for decision-making will be able to use the above operations. In particular, users of the Dicode use cases can use the

	operations provided by the workspace interface (presented in deliverables D4.1.1 and D4.1.2) to create and configure new workspaces where the collaboration will take place. Depending on their needs, they may deploy the workspace either in "forum view" or "mind-map view", where they are able to upload and process the available collaboration items via the respective interfaces. If decision making functionalities are required, users may transform a workspace into the "formal view" or use the "multi-criteria decision making view" to assist decision making through the execution of multi-criteria decision making algorithms.
Comments	The enhanced version of the service provides a proof-of-concept implementation of all the presented operations, which nevertheless may have bugs and malfunctions. Such issues will be addressed in future versions of the service, along with a more thorough evaluation of their usefulness.
Conformance classes	Not available.
Implementation rules	Not available.
Implementation status	Prototypical version implemented. The source code of the initial version of the collaboration service, implementing the above operations, can be found at the following Subversion repository https://anivas.cti.gr/websvn/listing.php?repname=ftel&path=/dicode/trunk/src/dll/
UML model	Not available.

In Appendix A, a more detailed description of the three decision making algorithms used in the "multi-criteria decision making view" is provided along with the way they are applied in Dicode use cases.

3 Development Process

The enhanced version of the previously described services have been developed using the guidelines outlined in deliverable D5.1.1: "Standards and guidelines for development". In particular, the Java Guidelines coding convention has been used to format the authored code. In addition, design patterns have been deployed to solve common software design problems.

To manage code changes, the involved project partners have installed the Subversion code repository on their site, which hosts the code of successive versions of the services they develop.

4 Future Work

This deliverable presents the enhanced version of the decision making support services that have been designed and developed in the context of Tasks 4.2, 4.3 and 4.5, representing a revised/enhanced implementation of the functional specifications outlined in deliverable D2.2 and revisited in deliverable D2.3. Future work will concentrate on improving and enriching the above services in order to fully address the needs of the Dicode project, taking into account the similarities and differences of the three Dicode use cases. Specifically, with respect to the implementation of the Decision Making Support services, future work will focus on a number of issues which include:

- Correcting bugs and malfunctions of the enhanced version of the implemented services.
- Assessing the enhanced version of the developed services against the functional specifications. The purpose of this action is to see how the enhanced version of the services must be changed in order to properly support the Dicode use cases. This includes identifying which operations must change their functionality, as well as which operations must be added in order to fully address the needs of the use cases. Much feedback is also expected from the project's second evaluation round.
- Appropriate tuning of the implemented services to better serve interoperability and integration issues (work towards this issue also concerns the related tasks of WP5).

5 References

Abonyi J. and Feil B. (2007). Cluster Analysis for Data Mining and System Identification. Birkhäuser Verlag AG, Berlin.

Campbell, D., Hutchinson, G.W. & Riccardo Scarpa, R. (2006). "Lexicographic Preferences in Discrete Choice Experiments: Consequences on Individual-Specific Willingness to Pay Estimates," Working Papers 2006.128, Fondazione Eni Enrico Mattei.

Coyle, G. Practical Strategy. Open Access Material, http://www.booksites.net/ download/ coyle/download.htm

Dicode Deliverable D2.2. (2012) – The Dicode Approach

Dicode Deliverable D2.3. (2012) – The Dicode Approach Revisited

Dicode Deliverable D3.1.1. (2012) – The Dicode Data Mining Framework

Dicode Deliverable D3.2.2. (2012) – The Dicode Data Mining Services (enhanced version)

Dicode Deliverable D4.1.1. (2011) – The Dicode Collaboration Support Services (initial version)

Dicode Deliverable D4.2.1 . (2011). The Dicode Decision Making Support Services (initial version)

Dicode Deliverable D5.1.1 . (2011). Standards and guidelines for development (initial version).

Friesen, N. and Rüping, S. (2012). Distance Metric Learning for Recommender Systems in Complex Domains, Mastering Data-Intensive Collaboration through the Synergy of Human and Machine Reasoning (dicoSyn 2012). A workshop at CSCW 2012, February 12, 2012, Seattle, WA

Kniaz, K., Analytic Hierarchical Process.NET, http://www.kniaz.net/software/ahp.aspx, visited 07/08/2012

Landauer, T., and Dumais, S. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. Psychological Review, 104 (2), 211-240.

Saaty, T.L. (1977). "A scaling method for priorities in hierarchical structures," Journal of Mathematical Psychology, 15, 57-68.

Saaty, T.L. (1980). The Analytic Hierarchy Process, McGraw-Hill International, New York, NY.

Saaty, T.L. (1983). "Axiomatic foundations of the Analytic Hierarchy process," Management Science, 32(2), 841-855.

Triantaphyllou, E., Shu, B., Nieto Sanchez, S. and Ray, T. (1999). "Multi-Criteria Decision Making: An Operations Research Approach," Encyclopedia of Electrical and Electronics Engineering, (J.G. Webster, Ed.), published by John Wiley & Sons, New York, NY, Vol. 15, pp. 175-186

Appendix A - Multi-Criteria decision making view

In the enhanced version of the decision making support services, the decision making process (apart from the "formal view" of collaboration workspaces) has been enhanced with a new view, the "multi-criteria decision making view". This new view exploits and incorporates algorithms coming from the field of Multi-Criteria (or Multi-Attribute) decision making (MCDM) (Triantaphyllou et al., 1999). Basic concepts of these methods are the *alternatives* and the *attributes* (or criteria).

Alternatives refer to the different choices (usually finite) available to the decision maker, while the *attributes* represent the different dimensions from which the alternatives may be viewed. Most of the MCDM methods require attributes to be balanced using an attribute weight (absolute or relative) to reflect the attribute importance in the decision making process. Each alternative is assessed based on the given attributes and, for each alternative, a corresponding score is calculated. An alternative with a higher score than another is perceived to be a better "solution" to the problem under consideration.

In Dicode terms, an alternative is represented on the "mind-map view" by a knowledge item of a specific knowledge type (which may vary according to the use case or workspace under consideration), while the following four attributes/criteria are used in the evaluation of the alternatives (see also deliverable D4.1.2):

- *Likes/Dislikes*. The number of an item's "Likes" minus the item's "Dislikes".
- *Creator Rating*. For the time being (subject to change in future versions), the Creator Rating is calculated as the sum of all Likes/Dislikes corresponding to the items the creator has contributed on the workspace.
- *Relationships in favour/against*. The algebraic sum of an item's "in favour" relations (depicted with green arrows on the "mind-map view" connecting knowledge items) minus the item's "against" relations (depicted with red arrows on the "mind-map view").
- *Item rating*. The total rating corresponding to the users' preferences (a user expresses his preference on an object through an "one-to-five star" rating scale).

After calculating the corresponding attribute values for all alternatives, all four values are normalized to the range [0 .. 100], and then a score is calculated for each alternative.

Based on the criteria already described, in the "multi-criteria decision making view" (Fig. 2), three algorithms have been implemented to get the alternatives' scores and support the decision making process in the context of Dicode:

- the Weighted Sum Model (WSM),
- the Analytical Hierarchy Processing (AHP), and
- the Lexicographic Decision Making rule.

For each algorithm, the user has to set all the necessary algorithm parameters. The algorithm is then executed and an ordered list of alternatives (the output of the algorithm) in descending score order is returned. The Dicode user may browse through the detailed results of the algorithm (to realize the reason why an alternative performs better than another one) or just view the plot with the scores of the alternatives.

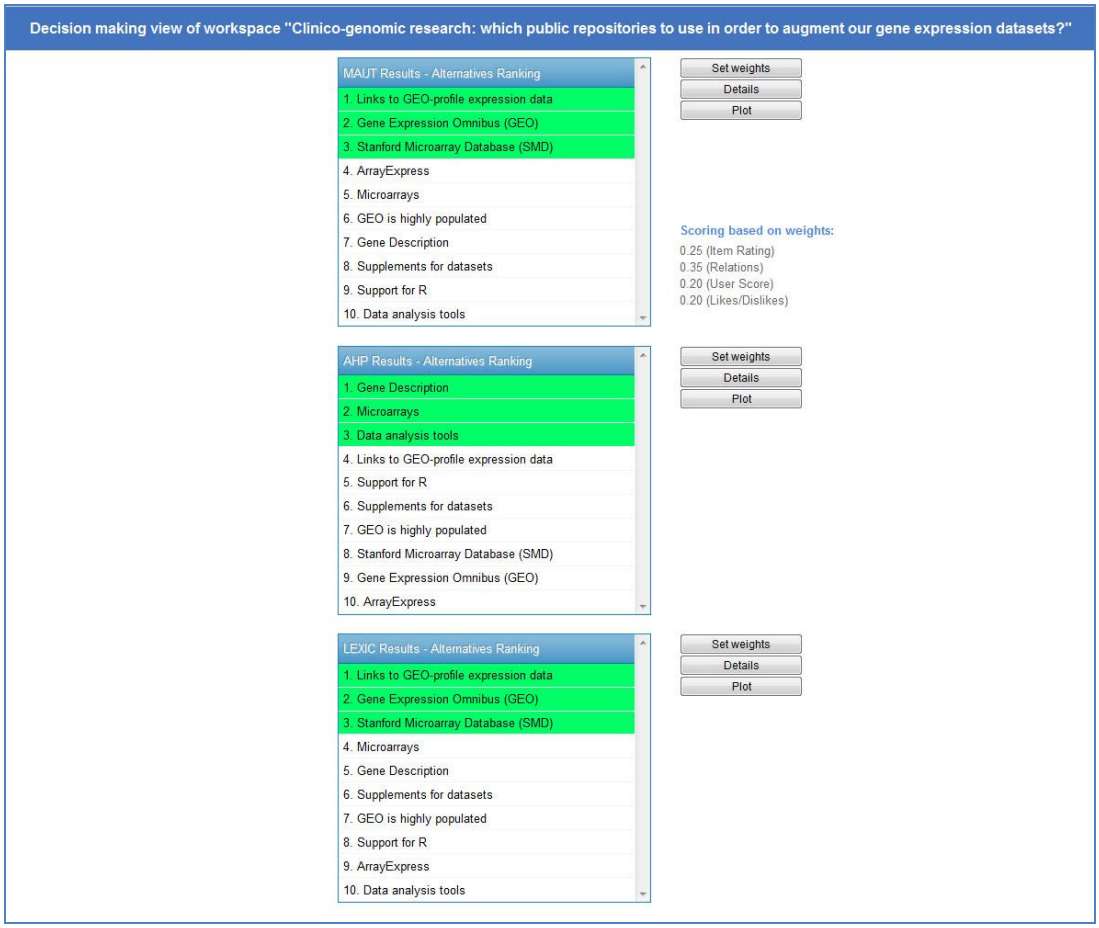


Figure 2: The Multi-Criteria decision making view with the tree results list, one for each algorithm.

More details about the implementation of the three algorithms are given below.

The Weighted Sum Model (WSM)

The Weighted Sum Model is the most popular MCDM approach. For a number of *M* alternatives and *N* criteria, the best alternative is the one with the top score calculated as

$$A^*_{wsm} = max_i \sum_{j=1}^{N} q_{ij} w_j \quad for\, i = 1,2,3,\ldots,M$$

where A^*_{wsm} is the score calculated for the best alternative, *N* is the number of criteria, q_{ij} is the subscore of the *i*-th alternative with respect to the *j*-th factor and w_j is the weight (predefined with respect to the problem under consideration or assigned by the user) of the *j*-th factor. The output of the algorithm is a list of the alternatives (Fig. 3) in descending score order based on their scores.

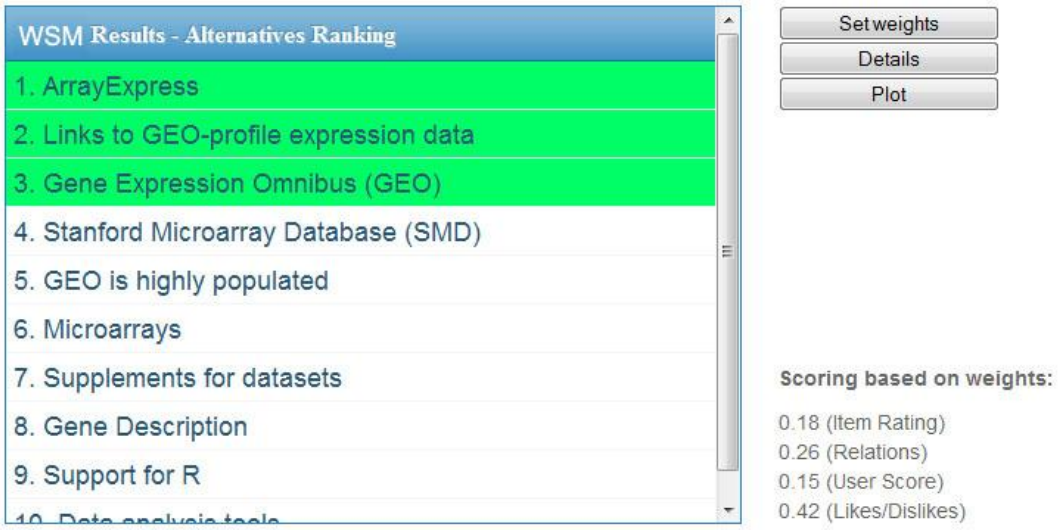


Figure 3: WSM alternatives list, top three alternatives in green background

The user may change the predefined weights of the four factors (Fig. 4), browse through each alternative's score and sub-scores (Fig. 5, each sub-score corresponds to one of the four factors) or view the plot of the results (Fig. 6).

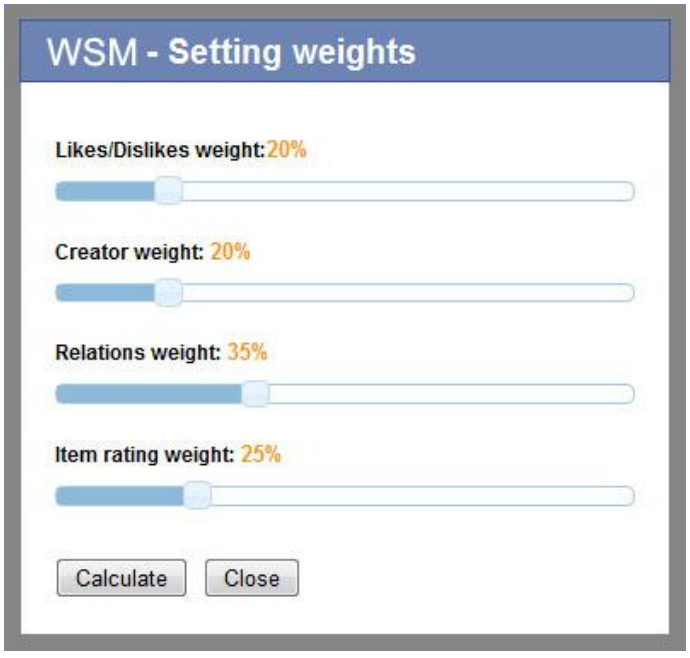
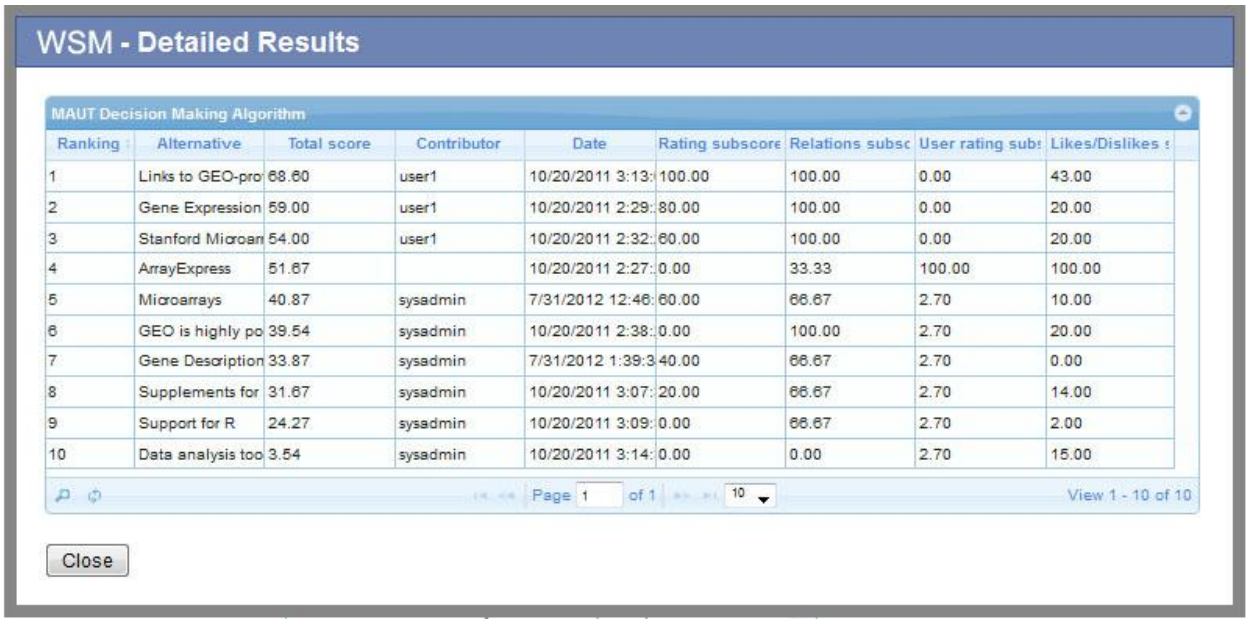


Figure 4: Setting WSM weights



Ranking	Alternative	Total score	Contributor	Date	Rating subscore	Relations subsc	User rating sub	Likes/Dislikes
1	Links to GEO-pro	68.60	user1	10/20/2011 3:13:	100.00	100.00	0.00	43.00
2	Gene Expression	59.00	user1	10/20/2011 2:29:	80.00	100.00	0.00	20.00
3	Stanford Microan	54.00	user1	10/20/2011 2:32:	60.00	100.00	0.00	20.00
4	ArrayExpress	51.67		10/20/2011 2:27:	0.00	33.33	100.00	100.00
5	Microarrays	40.87	sysadmin	7/31/2012 12:46:	60.00	66.67	2.70	10.00
6	GEO is highly po	39.54	sysadmin	10/20/2011 2:38:	0.00	100.00	2.70	20.00
7	Gene Description	33.87	sysadmin	7/31/2012 1:39:3	40.00	66.67	2.70	0.00
8	Supplements for	31.67	sysadmin	10/20/2011 3:07:	20.00	66.67	2.70	14.00
9	Support for R	24.27	sysadmin	10/20/2011 3:09:	0.00	66.67	2.70	2.00
10	Data analysis too	3.54	sysadmin	10/20/2011 3:14:	0.00	0.00	2.70	15.00

Page 1 of 1 10 View 1 - 10 of 10

Figure 5: WSM detailed results with alternatives scores and subscores

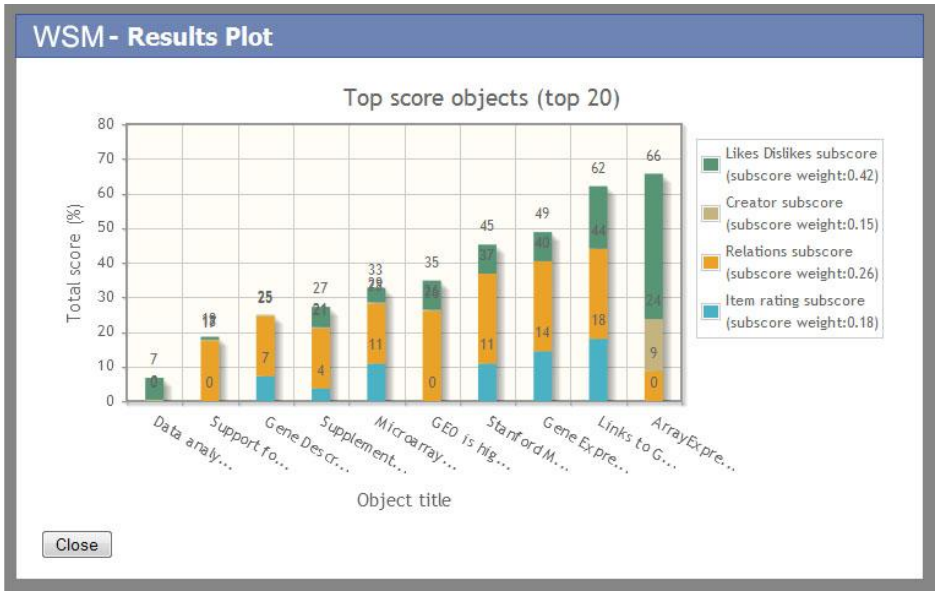


Figure 6: WSM alternatives scores and subscores plot (each colour corresponds to a subscore)

Analytic Hierarchy processing – AHP

Analytical Hierarchy processing (AHP) is based on decomposing a problem into a system of hierarchies. Its first step includes constructing a *NxN* matrix (*N* is the number of attributes) expressing the relative values of a set of attributes. Setting value *x* to the a_{ij} element of this matrix states that attribute *i* is *x* times more important than attribute *j*. The next step includes constructing *N* matrices of dimension *MxM* (*M* is the number of alternatives), where setting the value *y* in the element b_{ij} of the matrix states that alternative *i* is *y* times more important than alternative *j* (with respect to a specific criteria). Values of *x*

and y are taken from a common scale (the Saaty rating scale, see Table 1) used to declare the relative importance of an attribute (or an alternative) with respect to another.

Intensity of importance	Definition	Explanation
1	Equal importance	Two factors contribute equally to the objective
3	Somewhat more important	Experience and judgement slightly favour one over the other.
5	Much more important	Experience and judgement strongly favour one over the other.
7	Very much more important	Experience and judgement very strongly favour one over the other. Its importance is demonstrated in practice.
9	Absolutely more important.	The evidence favouring one over the other is of the highest possible validity.
2,4,6,8	Intermediate values	When compromise is needed

Table 1: The Saaty rating scale

Based on the previously described matrices, the *Relative Value Vectors* (RVV) and the *Option Performance Matrix* (OPM) are calculated using the eigenvectors of each table. The vector (VFM) including the corresponding alternatives scores is calculated as:

$$VFM = OPM*RVV$$

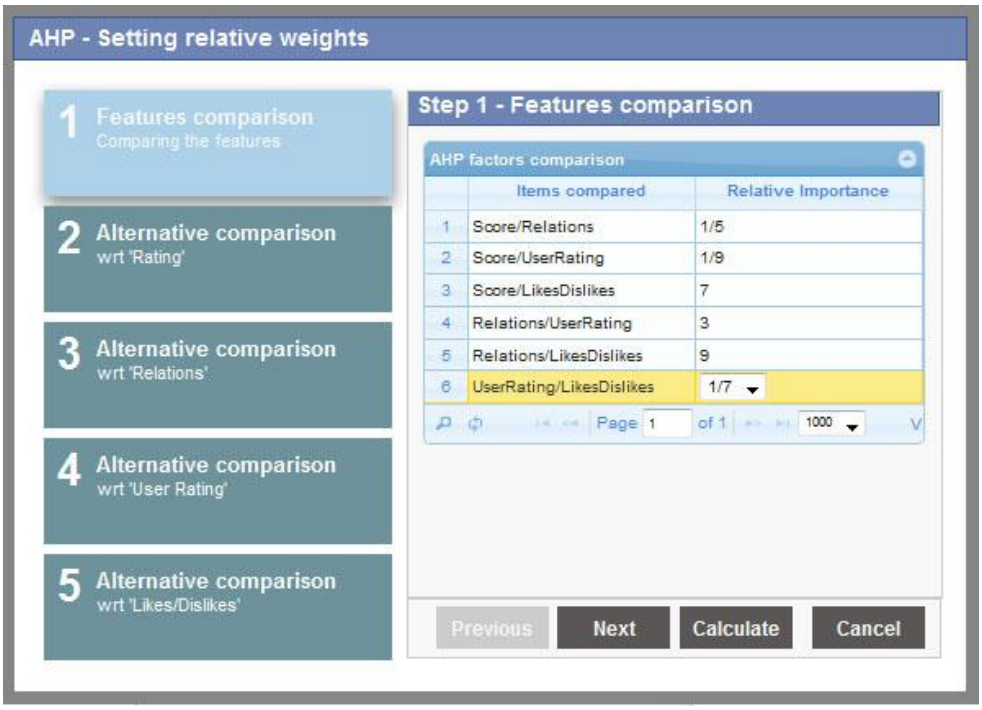


Figure 7: Setting the relative weights in AHP

In the context of Dicode, an open source library (Kniaz, 2012) has been used to conduct all the matrix calculations needed for the AHP algorithm. Concerning the implementation of the algorithm in Dicode, a wizard (Fig. 7) is used to perform all the basic steps of the AHP (that include inputting all the necessary values stating the relative importance among all pairs of criteria and alternatives).

Lexicographic decision making rule

The Lexicographic decision making rule (Campbell et al., 2006) is a decision rule based on ranking the attributes of the decision making process in terms of their importance. No compensation is allowed between the attributes and, consequently, a "good" sub-score of an alternative concerning one attribute cannot compensate for a "bad" sub-score concerning another attribute. The best alternative is the one with the best sub-score with respect to the most important attribute.

Figure 8: Sorting attributes with respect to their importance in the Lexicographic DM rule

In terms of Dicode, the Dicode user has to rank the four attributes based on their importance (Fig. 8). Calculating the rank of alternatives is based on the partial score of the most important attribute. If there are two or more equal sub-scores with respect to the most important attribute, the algorithm moves to the next more important attribute, compares the respective sub-scores and the procedure is repeated until all alternatives are distinguished and ranked (or the attributes are finished, in the case of alternatives with identical sub-scores).