Combining and Uniting Business Intelligence with Semantic Technologies

Acronym: CUBIST
Project No: 257403

Small or Medium-scale Focused Research Project
FP7-ICT-2009-5
Duration: 2010/10/01-2013/09/30

Visualisation Requirements Document

Abstract: Based on the directives provided by D.1.1.1, individual and integrated prototypes, this report describes the User Interface / Visualisation requirements for CUBIST.

Type	Other
Document ID:	CUBIST D4.1.1
Workpackage:	WP9
Leading partner:	CRSA
Author(s):	Cassio Melo (CRSA)
Dissemination level:	CO
Status:	Final
Date:	28 October 2011
Version:	1.0

Versioning and contribution history

Version	Description	Contributors
0.1	First draft	Cassio Melo (CRSA)
0.2	Review of section 2	Marie-Aude Aufaure (CRSA)
0.3	Incorporated review from INN	Cassio Melo (CRSA)
1.0	Incorporated review from SAS and SHU	Cassio Melo (CRSA)

Table of contents

1 Introduction

This deliverable will provide concrete requirements for the Formal Concept Analysis (FCA) visualisation tools to be developed in WP4. The report will describe how known techniques will be adapted and the new techniques to be implemented. This is a compiled and reviewed list of requirements concerning the graphical interface, potential visualisations and interaction styles, which emerged from the following assets:

	***Use Case documents*** *Use case documents from each CUBIST partner (D.7.1.1), (D.8.1.1), (D.9.1.1).*
	***General Requirements document*** *From the general requirements document D.1.1.1 it's possible to directly infer many of the actual visualisation requirements.*
	***Participatory design sessions notes*** *Participatory design sessions aimed to get user feedback from early phases of the project. In those sessions, user plays an active role in discussing the interface, workflow and interaction styles.*
	***User Mock-ups*** *Users were encouraged to create mock-ups illustrating their case (D. 9.1.2), (D. 1. 1. 3), those were discussed and combined in the integrated mock-up (D. 8. 1. 2).*
	***Similar software benchmarks*** *Many features presented in FCA-related software are also used to guide some of the interface requirements.*

This document is organized as follows: Section 2 discusses some of existing strategies to cope with visualisation and FCA; Section 3 describes the visualisation requirements, using modified *Volere* template[1]. Finally, Section 4 outlines the open issues to be discussed whether they are in the project's scope.

[1] Volere Template. http://www.volere.co.uk/template.htm

2 FCA Visualisation

A concept lattice is traditionally represented by a Hasse diagram illustrating the groupings of objects described by common attributes. A Hasse diagram is a graph where concepts appear as vertices on the plane connected by line segments or curves [1]. Lattices visualisation becomes a problem as the number of clusters grows significantly with the number of objects and attributes. Interpreting the lattice through a direct visualisation of the line diagram rapidly becomes impossible therefore more synthetic representations are needed.

In order to reduce the complexity of lattices, simplified diagrams can be produced by displaying only concepts with a sufficient relevance [2]. Visualisations can also be restricted to portions of the data [3], and concept number reduction is possible by incorporating conditions into the data mining process [4]. Conceptual measures can be applied to identify the most relevant concepts and filter outliers [5]. In CUBIST, these simplifications by *InClose*[2], a Formal Concept Analysis miner software in development by SHU.

To deal specifically with the visual complexity of Hasse diagrams, several approaches allow users to dynamically explore and reveal specific parts of the diagram, using visual query languages [6-8]. However these techniques do not provide a clear view of the entire lattice.

Other FCA visualisation approaches map the distances between concepts to visual variables, in order to highlight patterns. For example, in [9], similar concepts are represented as similarly coloured pixels placed in the 2D space along a Peano-Hilbert curve, so that similar concepts are placed close to each other. Nevertheless, detailed relationships between concepts are lost in these representations. Finally, systems often provide users with hybrid/combined lattice visualisation [10], e.g. showing both a general Hasse diagram and a tag cloud for representing the neighbours of a specific concept (for a review see [11]).

2.1 Visualisation and Interaction Techniques for FCA

Some visualization/interaction techniques envisioned by CUBIST are described below.

Iceberging

The full topology of a lattice seems to be little help in the analytical process [16]. The display of the traditional Hasse diagram should be only partial for large lattices using techniques such as the “Iceberging”. Iceberging is a technique by which only a portion of a lattice is generated and visualised, centering on a particular node or node-cluster of interest. An FCA defined 'region' of the data can then be queried in more detail.

Sub-contexts

Large data sets may be sub-divided using specific properties or values: data classes may be considered as sub-contexts, or data might be naturally divided into chronological sections (e.g. monthly data).

Lattice as Trees

One approach consists in representing lattices not as Hasse diagrams, but as trees. Different criteria may be used to extract trees from lattices, i.e., how to select a single parent for each concept node, and visualise the resulting trees. Trees are inherently simpler hierarchical structures than Hasse

[2] InClose FCA miner - http://sourceforge.net/projects/inclose/

diagrams and due to their applicability in many domains, there is a plethora of tree representations. These include: indented outline trees, sometimes called a “tree list” (common in file browsers such as *Microsoft Windows Explorer*), traditional layered node-link diagrams in 2D or 3D (e.g. ConeTrees [12]), spatially transformed tree diagrams (e.g. Radial [13]) as well as several space optimization (Space Optimized trees [14]) and space-filling tree visualisation techniques (e.g. TreeMaps [15]).

Concept clustering

Concept clusters are nodes formed by grouping together closely related concepts. They are defined by a specified commonality of attributes and/or objects. They are closely related to the notion of 'rough sets', and, as such, may also be usefully part of the warehouse Triple Store.

Query graphs

Using FCA analysis, we will inform users as to how their queries are being interpreted by the system (e.g. representing the sub-ontology or cluster of data that the query has been related to by the system). Thus the user will be presented with a "query graph" placing the query within FCA. The user will then be able to interact with the graph to either refining the query (by navigating the query graph) or explore alternative FCA interpretations.

Lattice navigation

Traditional FCA-software enables panning and zooming by drag-and-drop the lattice. An interesting alternative for navigation is combination of a key and mouse pointer, for instance, user holds the “*shift*” key to gain an interactive control for scrolling through layout's anchor point in response to the mouse movement, as proposed in [19].

Focus+Context

Optionally it is possible to enable the *Fish Eye* [17] distortion - it makes the concept node appear bigger when the mouse pointer is over and display its subsequent relatives in a decreasing scale size. This technique is well known in graph navigation literature, known as *focus+context distortion* [18]. It allows the user to focus on items he or she selects without losing the context around (e.g. the concept hierarchy).

2.2 Visualisation Libraries Candidates

Recent adoption of *HTML5* and performance improvements in *JavaScript* engines by popular web browsers led to the development of sophisticated visualisation libraries that employ these technologies. On the other hand *Adobe Flex* is still an industry standard for *Rich Internet Applications (RIA)* and makes the interface prototyping faster than most technologies. The following visualisation packages are among the best candidates for CUBIST visualisation:

- **The Prefuse/Flare**[3] *Flash* package is an extensible visualisation framework that handles different data formats, graph options, operations and transitions. It is mainly focused on graphs visualisation but it has a handful of other layout options;
- **D3**[4] is a *JavaScript* visualisation framework based on *CSS*, *HTML* and *SVG*, successor of the popular InfoVis toolkit. It has a number of visualization layouts and can handle different data structures (e.g. tree, graphs);

[3] Flare - https://github.com/prefuse/Flare

[4] D3 - http://mbostock.github.com/d3/

- Compared to D3, **Rgraph**[5] is more analytics oriented, providing several chart types and extensions in *HTML* and *SVG*.

Although *Prefuse/Flare* allows faster interface prototyping as opposed to *JavaScript* libraries mentioned above, as the web moves to more open and interactive applications, one should consider the migration to open standards such as *HTML* and *JavaScript*.

[5] Rgraph - http://www.rgraph.net/

3 Visualisation Requirements

Each requirement is described in the table below including its code (Req. #); where it came from (Originator), feature, description, rationale and priority (Mandatory | Important | Desirable).

Req. #	Originator	Interface feature	Description	Rationale	Priority
VIS001	D.1.1.2	Lattice Visualisation	Display the concept lattice visually (e.g. *Hasse* diagram).	Concept data should be visualized by the user.	Mandatory
VIS002	General requirements	Multiple Layouts	Provide different layouts for visualizing the concept lattice.	Alternative visualisations may be more suitable for some analysis.	Mandatory
VIS003	SAP008, Integrated prototype	Partially show the lattice	Concept lattice can be displayed partially according to the selected attributes, for instance.	It hides unimportant concepts for the current analysis.	Mandatory
VIS004	INN018	Search with auto-complete	Allow searching for attributes/objects with auto-complete.	It enhances input performance and avoids misspellings.	Desirable
VIS005	Software benchmark	Concepts labelling	Provide different options for labelling concepts (e.g. non-repetitive, attributes number, etc).	Labels are used to identify concepts.	Mandatory
VIS006	Software benchmark	Pan and zoom canvas	Pan by drag-and-drop visualisation and zooming, similar to Google maps.	It facilitates navigation and visualisation.	Mandatory
VIS007	Integrated prototype	Use visual variables to display information	Shape, color and size can be used to represent values of a given dimension (e.g. stability, ratio between attributes).	It allows visual perception of measures and facilitates comparison among concepts.	Mandatory
VIS008	General requirements	Faceted Navigation	Display a tree of facets from concepts in the lattice and allow navigation through facets.	Facets represent more intuitively the concept hierarchy.	Mandatory
VIS009	General requirements	Sync selected concepts in visualisation with data table	When selecting one or more concepts, the selection should reflect/highlight	It allows users to perceive links between data and generated	Desirable

			related data as well.	concepts.	
VIS010	General requirements	Smooth transitions	Provide animated transitions between different layouts or transformations in the current visualisation.	It helps user keep track of changes in the visualisation.	Desirable
VIS011	General requirements	Select concepts	User can click-and-select one or more concepts.	It allows performing operations on specific concepts.	Mandatory
VIS012	General requirements	Search within selection	It should be possible to restrict the search space to only current concepts.	It allows drill-down analysis of data.	Desirable
VIS013	General requirements	Lattice overview	Display stats and interesting facts about the concept lattice.	It provides an overview of the current lattice.	Mandatory
VIS014	General requirements	Concept overview	Display stats about the selected concept(s).	It provides an overview of current selected concept(s).	Mandatory
VIS015	SAP008	Filtering	Hide/show concepts based on certain criterion such as attributes, attributes values, thresholds (e.g. support). Any change should take effect in the lattice immediately.	It constraints data visualisation based on specific characteristics.	Mandatory
VIS016	SAP010	Preferred visualisation for each attribute	It should be possible to select a particular graph type (e.g. pie chart, bar chart, histogram, timeline, etc) as default visualisation for each attribute.	It provides data distribution visualisation for each attribute.	Important
VIS017	Integrated prototype	Visual filtering	Provide visual filtering capabilities from the preferred visualisation for each attribute.	Visual data filtering of attribute values is usually more intuitive than categorical filtering.	Important
VIS018	Integrated prototype	Filter history	An ordered list of filters applied to this visualisation.	It helps user keep track of visualisation states.	Mandatory

VIS019		Layout options	Provide drawing options for each layout.	It provides user customization for the current visualisation.	Mandatory
VIS020	General requirements	Concepts comparison	Selected concepts can be compared by showing their attribute values, correlations, and other interesting information they may have.	Used to compare and discover facts about specific concepts.	Desirable
VIS021	General requirements	Display association rules graphically	The association rules should be encoded visually when possible.	Association rules can be better understood if displayed graphically.	Important
VIS022	Integrated prototype	Combine visualisations	Combine visualisations (e.g. pie-charts in a timeline) for reflecting different attribute-values selection.	It allows the association of concepts with attribute values.	Important
VIS023	Software benchmark	Highlight concept connections	Concept relations with other concepts should be highlighted from the rest of the lattice.	Neighbour concepts are usually more relevant than non-related concepts.	Mandatory
VIS024	HWU mock-up	Full screen visualisation	Allow full screen mode of the current visualisation.	Allow user to focus on the visualisation by expanding its area and hiding other interface components.	Important
VIS025	SAP005	Nesting of visualisations	Embed visualisations of different attributes in a concept	It facilitates drill-down analysis.	Desirable
VIS026	SAP mock-up	Dual lattice browsing	Displays lattices for two (eventually more) attributes side-by-side. Selection of one concept reflects on the other lattice.	Facilitated browsing and easy comparison between two attributes.	Desirable
VIS027	Software benchmark	Keep track of calculated metrics	Display a history of already calculated metrics.	The interface should prevent user from calculating twice	Important

				the same metric,	
VIS028	HWU030, SAP013	Save/Load visualisation/filters	Allow saving and loading visualisations states.	Save visualisations states for further analysis later or to analyze in a particular mode (e.g. filters for embryonic cases)	Mandatory
VIS029	SAP014	Export visualisation to image or flash object	Export the actual visualisation state to image or flash object.	Files can be shared and embedded in other formats.	Mandatory
VIS030	SAS010, HWU067	Progress bar	Display progress bar whenever the computer response takes more than 3 seconds.	It gives to user an estimated time to complete or at least, that some processing is being done.	Mandatory
VIS031	HWU061	Tips and infobars	Constant feedback on user operations.	It helps novice users to master different functionalities of the tool.	Important
VIS032	General requirements	Themes	Personalised colour/layout scheme for each partner.	It keeps the visual identity of the company.	Important

4 Discussion Items

The following items are combinations of two or more visualisation functionalities mentioned above.

Display trend information (HWU023, HWU024, HWU025)

It has been proposed by HWU to display trend information, i.e., how to enable temporal data browsing and analysis in a concept lattice. This functionality can be partially accomplished by VIS022, for instance, combining timeline visualisations with concept visualisation.

Suggest visualisation and scales (SAP010)

It may be possible to infer data type, and consequently data visualisation, for each attribute in the context table. For instance, many-valued attributes can be displayed as pie or bar charts whereas date and time can be represented by calendar and timelines respectively (VIS016).

Location based lattice (INN005)

INN005 expresses the need for geographical filtering of concepts. It is not clear if this filtering should occur during the data preparation step or during the analytical process. Geographical filtering through interface can be addressed by VIS016 and VIS017, for instance, defining a map visualisation for each geographical attribute and allowing filtering by place + radius.

Concept Node Zooming (SAP007)

Navigating through successive subsets of information can be accomplished by the activation of filters (VIS015 and VIS018) and nesting of visualisations (VIS025).

Highlight inconsistencies in data (HWU022)

Identify and visualise unusual patterns in data may require both pre- and post- processing of data in order to inform to the graphical interface which data is considered “inconsistent”. Visual variables, such as colour can be used to highlight an unusual pattern in data (VIS007).

5 References

[1] Di Battista, G.; Tamassia, R. (1988), "Algorithms for plane representation of acyclic digraphs", Theoretical Computer Science 61: 175–178.

[2] Stumme, G., Taouil, R., Bastide, Y., Pasquier, N., and Lakhal, L. Computing iceberg concept lattices with Titanic. In Data & Knowledge Engineering, Volume 42, Issue 2, pp. 189-222, 2002.

[3] Ducrou, J., Eklund, P., and Wilson, T. An Intelligent User Interface for Browsing and Searching MPEG-7 Images Using Concept Lattices. In S. Ben Yahiaet al. (Eds.): CLA 2006, LNAI 4923, pp. 1–21, Springer-Verlag Berlin Heidelberg 2008.

[4] Zaki, M.J., Hsiao, C-J. Efficient Algorithms for Mining Closed Itemsets and Their Lattice Structure. In IEEE Transactions on Knowledge and Data Mining, Vol. 17, No. 4, IEE Computer Soc., 2005.

[5] Le Grand, B., Soto, M., Aufaure, M.-A. (2009) “Conceptual and Spatial Footprints for Complex systems Analysis: Application to the Semantic Web”, in 20th International Conference on Database and Expert Systems Applications 2009, pp.114-127.

[6] Blau, H., Immerman, N.,and Jensen, D.. A Visual Language for Querying and Updating Graphs. University of Massachusetts Amherst, Computer Science Department Tech: Report 2002-037. 2002.

[7] Cruz, I. F., Mendelzon, A. O., and Wood, P. T.. A Graphical Query Language Supporting Recursion. In Proc. of the Association for Computing Machinery Special Interest Group on Management of Data, pages 323–330. ACM Press, May 1987.

[8] Consens, M., and Mendelzon, A. Hy+: a Hygraph-based query and visualisation system. SIGMOD Record, 22(2):511–516, 1993.

[9] Michel Soto, Benedicte Le Grand, Marie-Aude Aufaure, "Spatial Visualisation of Conceptual Data," International Conference Information Visualisation, pp. 57-61, 2009.

[10] Eklund, Peter, Villerd, Jean. A Survey of Hybrid Representations of Concept Lattices in Conceptual Knowledge Processing Formal Concept Analysis. Lecture Notes in Computer Science 2010, Springer Berlin/Heidelberg, pp. 296- 311

[11] C. Roth, S. Obiedkov, D. G. Kourie. "Towards Concise Representation for Taxonomies of Epistemic Communities", CLA 4th Intl Conf on Concept Lattices and their Applications. 2006.

[12] George Robertson, Jock D. Mackinlay, Stuart Card. Cone Trees: Animated 3D Visualisations of Hierarchical Information. In Proceedings of the ACM CHI 91 Human Factors in Computing Systems Conference, pages 189-- 194, Association for Computing Machinery. 1991.

[13] P. D. Eades. Drawing free trees. Bulletin of the Institute for Combinatorics and its Applications, 5:10–36, 1992.

[14] Quang Vinh Nguyen; Mao Lin Huang. A space-optimized tree visualisation, Information Visualisation, 2002. INFOVIS 2002, pp. 85- 92, 2002

[15] B. Johnson, B. Shneiderman: Tree maps: A Space-Filling Approach to Visualisation of Hierarchical Information Structures. IEEE Visualisation 1991: 284-291

[16] Carpineto, C., & Romano, G. (1995). Ulysses: a lattice-based multiple interaction strategy retrieval interface. In B. Blumenthal, J. Gornostaev & C. Unger (Eds.), Human-Computer Interaction,LNCS 1015-Springer, 91-104.

[17] Sarkar, M. & Brown, M. H. (1992). Graphical Fish–eye views of graphs. In Human Factors in Computing Systems, CHI ’92 Conference Proceedings, ACM Press, pp. 83–91.

[18] Kreuseler, M. and Schumann, H. (1999). Information visualization using a new Focus+Context Technique in combination with dynamic clustering of information space, Proc. NPIV'99, Missouri, pp. 1-5, 1999, ACM Press.

[19] Cassio A. Melo, Marie-Aude Aufaure, Bénédicte Le-Grand. Towards a Visual Analytics for Formal Concept Analysis, in XI Conférence Internationale Francophone sur l’Extraction et la Gestion des Connaissances (EGC 2011) / IX Atelier Visualisation et Extraction de Connaissances. Brest, France, 2011.

Annex A - Requirements Related to Visualisation

The following table of requirements was excerpted from the Generalized Requirements Document D.1.1.2 related to the graphical interface.

Partner	Originator:	Req. #:	Priority	Description	Rationale	Fit Criterion
MANDATED						
HWU	Kenneth McLeod	HWU011	1 mand.	Must not require installation of plug-in, extension etc.	Many biological users will not control their computer & thus cannot install software	CUBIST runs perfectly in a newly installed browser
FUNCTIONAL						
Inno	Hazzaz Imtiaz	INN001	1 mand.	use FCA to discover hidden relations within the data	This is one of the objectives of the DoW.	Test the use case prototypes
Inno	Hazzaz Imtiaz	INN002	1 mand.	Produce semantic visual analytic widgets for representation of data	This is one of the objectives of the DoW.	Test the use case prototypes
Inno	Hazzaz Imtiaz	INN005	1 mand.	Radias job search to search for jobs around a town, county or postcode or part of a postcode by number of miles	Location search with mere city or county name is not very useful	UAT testing with Innovantage sales team with a range of radias searches
Inno	Hazzaz Imtiaz	INN018	1 mand.	User should search and view job vacancies matching complicated criterions: Title of the job vacancy, Description, Advertiser, Location , Date posted, Sector, Source, Job category etc. Search is essentially conducted in	core functionality in recruitment	UAT testing with Innovantage sales team with a range of searches fulfillin various criteria

				the visual analytics frontend with drilling down functionalities		
SAP	Frithjof Dau	SAP005	1 mand.	different visualisations can be combined and "nested"	having only one concept lattice for the visualisation is insufficient: Different attributes may require different types of visualisations, and if different attributes are to be explored, theses visualisations must be combined	it is for example possible to display a concept lattice (outer scale) where each node in turn is a dedicated visualisation (e.g. a pie chart or sunburst diagram) of another attribute
SAP	Frithjof Dau	SAP007	1 mand.	it is possible to zoom into a concept node	a given concept node corresponds to a specific subset of the information space, and a user might want to explore only the given subset	CUBIST allows to zoom into a concept node and to conduct further analysis (again with visual analytics) for the objects and attributes of the given concept node
SAP	Frithjof Dau	SAP008	1 mand.	it is possible to select a part of the overall concept lattice to be displayed	the overall concept lattice might be too large to be displayed in total	CUBIST offers possibilities to zoom into parts of the concept lattice, e.g. by displaying for a given node only nodes in the neighboorhood, by using a fisheye view for segment

						of the concept lattice
SAS	Alexander Mikhailian	SAS003	1 mand.	CUBIST software shall use FCA for suggesting new patterns in data	This is one of the objectives of the DoW.	Via testing of the use case prototypes v1 and v2.
SAS	Alexander Mikhailian	SAS004	1 mand.	CUBIST software shall produce visual representatio n of data	This is one of the objectives of the DoW.	Via testing of the use case prototypes v1 and v2.
SAS	Alexander Mikhailian	SAS005	1 mand.	CUBIST software shall store existing data in an RDF store and provide an extended SPARQL query interface to the data.	This is one of the objectives of the DoW.	Via testing of the use case prototypes v1 and v2.
SAS	Alexander Mikhailian	SAS007	1 mand.	CUBIST software shall try to avoid (where possible) terminology of the underlying technology	Users come from different backgrounds	Via the review of text strings deployed in software.
SAS	Alexander Mikhailian	SAS010	2 des.	Make users aware of the progress for tasks that take > 1sec.	Users must realise that system is working	By review of design
HWU	Kenneth McLeod	HWU022	2 des.	Locate errors/inconsi stencies in underlying data	Biological data is naturally inconsistent & incomplete. Users find it hard to deal with these issues	Suggests inconsistency "solutions" to user
HWU	Kenneth McLeod	HWU023	2 des.	Display trend information for expression level	Level changes over time	Ask human expert to verify three examples
HWU	Kenneth McLeod	HWU024	2 des.	Display trend information for expression pattern	Pattern changes over time	Ask human expert to verify three examples

HWU	Kenneth McLeod	HWU025	2 des.	Display trend information for co-expression level	Co-expression changes over time	Ask human expert to verify three examples
HWU	Kenneth McLeod	HWU027	2 des.	Identify genes with similar expression patterns	Provide spatially orientated co-expression information	Ask human expert to verify three examples
HWU	Kenneth McLeod	HWU028	2 des.	Describe similarities of genes involved in same process	Indicates what other processes, functions etc the genes may be involved in	Ask human expert to verify three examples
HWU	Kenneth McLeod	HWU030	2 des.	Queries & results may be saved & reloaded later	Allows users to return to previous state	A test query can be exported, then later imported successfully
HWU	Kenneth McLeod	HWU031	1 mand.	Filter results (before lattice is computed)	Users may find it helpful to deal with a subset of the full result set	Result set can be reduced by setting a range of parameters
HWU	Kenneth McLeod	HWU032	2 des.	Flexible presentation of results	Different personas have different computing background & experience	User can switch to different visualisation of same information
SAP	Frithjof Dau	SAP001	2 des.	Full text search on displayed concept lattices	Users are familiar with full-text search, and finding concepts in huge concept lattice manually is too time-consuming	A search box is implemented, and found concept nodes are highlighted
SAP	Frithjof Dau	SAP011	2 des.	CUBIST offers different layout algorithms for the layout of a Hasse diagram	there exist different layout algorithms, and depending on the use case and user context, one or the other is more appropiate	CUBIST offers to choose from different layout algorithms (e.g. spring algorithm or a chain decompositio n)
SAP	Frithjof Dau	SAP013	2 des.	visualisations can be stored and reloaded	For good visualisations, users might want to be able to store them in oder to reuse them	load and "save" functionality available for visualisations

					later	
SAP	Frithjof Dau	SAP014	2 des.	visualisations can be exported	For good visualisations, users might want to export them (e.g. to png or gif) on order to print or distribute them	export functionality available for visualisations
SAS	Alexander Mikhailian	SAS008	2 des.	Provide a FCA-based visual outlook of the data in the triple store	Space control centre operations run on tight schedule. A graphical UI for the otherwise numeric data will simplify cognitive operations	
SAS	Alexander Mikhailian	SAS009	2 des.	Provide an interactive environment to users	Users need to navigate the data and to receive instant feedback from software.	By review of design
SAP	Frithjof Dau	SAP010	3 opt.	CUBIST suggests scales and visualisations for given attributes	unexperience d users might be overstrained with choosing appropiate scales or visualisations	CUBIST offers a "sugges visualisation" button for a selected attribute
LOOK AND FEEL						
SAP	Frithjof Dau	SAP009	1 mand.	it is possible to manually move nodes in the display of the concept lattice	the automatic layout of the concept lattice might be inappropiate or insufficient, and the user want to manually correct the layout	CUBIST allows nodes to be manually moved
HWU	Kenneth McLeod	HWU051	2 des.	Interface should be clean and uncluttered	Makes interface easier to use	

Inno	Hazzaz Imtiaz	INN016	2 des.	Look and Feel of the UI in general needs to be lively, funky which includes colour of the panels and fonts, font needs to be appropriate	Should have a user friendly interface	Test with Innovantage sales team
Inno	Hazzaz Imtiaz	INN017	2 des.	Start page can have a basic search interface on left pane and current jobs on right pane, i.e. reed.co.uk.	Should have a user friendly interface	Test with Innovantage sales team
SAP	Frithjof Dau	SAP006	2 des.	the frontend can be customized	different users and use case partners have different priorities and preferences for the look&fell of the frontend	CUBIST offers a set of options for the display of concept lattices
HWU	Kenneth McLeod	HWU052	3 opt.	Looks like prominent web apps	Mimicking resource will make CUBIST seem friendlier & easier to use	See description
USABILITY						
HWU	Kenneth McLeod	HWU001	1 mand.	Provide analytical features on top of existing data	Lots of data, yet no analytical features	CUBIST provides a series of analytical features for EMAGE data
HWU	Kenneth McLeod	HWU060	3 opt.	Use biological metaphors when designing interface	Biologists understand biological metaphors but not computing ones	
HWU	Kenneth McLeod	HWU061	2 des.	Provide constant feedback to users	Biologists are often not computer experts & need lots of support. Feedback improves the transparency too.	Users are able to describe what CUBIST is doing whilst using the system

HWU	Kenneth McLeod	HWU062	1 mand.	Flag data when unsure of auto correction (HWU022).	System needs to be transparent and provide constant feedback to users	Users can tell when CUBIST has changed the underlying context (e.g. with fult tolerance algorithms)
HWU	Kenneth McLeod	HWU064	2 des.	Users will be able to use system once they have watched the screencast	Users are unwilling to read manual or work through a tutorial	Users can use CUBIST when their only introduction is the screencast
HWU	Kenneth McLeod	HWU065	2 des.	Users should only need to watch screencast once	Users are unwilling to use tools that seem hard to use	Users can use CUBIST 2 weeks later without any help or reminders
HWU	Kenneth McLeod	HWU066	2 des.	Use standard web metaphors	Users are familiar with web	
HWU	Kenneth McLeod	HWU067	2 des.	Make users aware of progress	Users must realise that system is working	
HWU	Kenneth McLeod	HWU068	2 des.	Interface should be simple	Biologist are not comfortable with complex interfaces, and traditionally ignore them	Evaluation shows biologists describe the interface as "simple"
HWU	Kenneth McLeod	HWU069	2 des.	Visualisations should be detailed and convey large amounts of information (for personal computational scientist)	If they don't might as well scan web pages	Computational biologists indicate all "key" information is in visualisation
HWU	Kenneth McLeod	HWU070	2 des.	Visualisations should be simple (for persona biologist)	Biologists are not used to dealing with lattices and other forms of visualisation CUBIST is likely to employ	Biologists can interpret visualisations