



Project no. 033104

MultiMatch

Technology-enhanced Learning and Access to Cultural Heritage
Instrument: Specific Targeted Research Project
FP6-2005-IST-5

D6.1.2 Designing the User Interface for the First Prototype

Start Date of Project: 01 May 2006

Duration: 30 Months

Organization Name of Lead Contractor for this Deliverable: USFD

Final version

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)

Document Information

Deliverable number:	D6.1.2
Deliverable title:	Designing the User Interface for the First Prototype
Due date of deliverable:	September 2007
Actual date of deliverable:	September 2007
Main Author(s):	Paul Clough, Jennifer Marlow, James Carmichael, Juan Manuel Cigarran Recuero, Javier Artiles, Julio Gonzalo, Daniela Petrelli
Participant(s):	USFD, UNED
Workpackage:	1
Workpackage title:	User Interaction and Interface Design
Workpackage leader:	Paul Clough
Est. person months:	4
Dissemination Level:	PU (Public)
Version:	Final
Keywords:	User interfaces, user interaction, information seeking behaviour

Abstract

This document describes the design of the interface for the first MultiMatch prototype. This system aims at providing various user groups with multilingual access to a rich variety of cultural heritage resources (e.g. images, videos, web pages and structured texts). Specifically, this document describes the interface design process, a variety of relevant literature, the results of initial interviews with expert cultural heritage users and the current implementation strategy. This deliverable (D6.1.2) is the final version of D6.1 (previously released as a preliminary version – D6.1.1 – at 12 months and presented as input for the first year annual review).



Table of Contents

Document Information	1
Abstract	1
Executive Summary.....	3
1 Introduction	4
1.1 Relation with the MultiMatch architecture.....	4
2 Design Methodology	4
2.1 User-centred design approaches	5
2.2 The MultiMatch design process	6
2.3 Design inputs.....	8
3 User interface designs.....	26
3.1 Description of initial interface for Prototype 1.....	27
3.2 Next steps	33
4 Implementation and architecture	34
4.1 AJAX coding technology	35
4.2 MultiMatch user interface technology.....	37
4.3 MultiMatch User Interface Architecture	39
4.4 MultiMatch development environment	43
5 Discussion and summary.....	44
5.1 Planned experiments and user testing.....	44
5.2 Functionality of final prototype.....	45
References	52
APPENDIX A – Sites used for competitor analysis.....	56
Cultural Heritage related sites	56
Multimedia search engines	58
Browsing capabilities	58
APPENDIX B – SCREENSHOTS OF GOOD EXAMPLES	59
APPENDIX C – RESPONSES TO WIND SURVEY	66
APPENDIX D – PRELIMINARY DESIGNS	69
Screen shot from kick-off meeting	69
UNED Version 1	70
USFD Version 1	71
UNED Version 2	72
USFD Version 2	73
Version 3	74
Version 4	75
APPENDIX E: INITIAL INTERFACE FOR PROTOTYPE 1	76
APPENDIX F	83
USER INTERFACE COMPONENT /WIDGETS LIST.....	83



Executive Summary

This deliverable describes the design and implementation processes for the user interface to MultiMatch, a multilingual/multimedia search engine in the cultural heritage domain. This deliverable (D6.1.2) is the final version of D6.1 (previously released as a preliminary version – D6.1.1 – at 12 months and presented as input for the first year annual review).

A user-centred design approach has been adopted in the design stage that incorporates the feedback and views of users at various stages within the design lifecycle. This report describes the design methodology used to research and create the design for the initial interactive version of MultiMatch (the first prototype). This includes a summary of user-centred design for user interfaces and the approach used in MultiMatch, a summary of findings from research inputs which have influenced and guided the design (e.g. past literature, comparison between existing cultural heritage sites and portals and interviews of professional users as part of a task analysis). The report goes on to describe a series of initial interface designs before concluding with a description of the current MultiMatch interface design for the first prototype.

In addition to describing the interface design process, we also discuss and present our current interface implementation strategy. To fit in with the overall MultiMatch architecture and provide maximum flexibility, we have used the Google Web Toolkit (GWT) to develop the specified functionality that provides a framework in which to realise the interface design, and provide a set of Application Programming Interfaces (APIs) that can provide MultiMatch functionality within custom applications. We describe the GWT web development approach, together with justification for why we have selected this implementation strategy. The code associated with the components described in this deliverable has been stored on the MultiMatch integration server hosted by OCLC PICA

Finally we conclude this report by describing the user tests which we are planning to carry out in order to evaluate the user interface, and a brief description of the functionality that we believe the final MultiMatch interface will offer users.

1. Introduction

The MultiMatch user interface will provide end users with multilingual/multimodal access to rich cultural heritage information consisting of texts, images, audio and video. The interface will enable different classes of users (identified in WP1) to search and browse the system synergistically, making use of additional semantics and ontological information provided by WP4 and WP2 respectively. User evaluations are instrumental in determining the usability of proposed techniques in WP5, and recommendations based on usability findings (efficiency, effectiveness and user satisfaction) are fed back into the design process. The interface will support the typical search tasks of users (identified in WP1), and provide effective information access to rich multimedia and multilingual data based on empirical user evaluations and usability assessment.

A well-designed search interface should assist users in clarifying their information needs, and subsequently help them formulate suitable queries and understand the results (Hearst, 1999; Shneiderman, 1997). More recently, attention has been paid to human-computer interaction in information retrieval and interface design has been driven by the needs of end users, their information-seeking behaviour and psychological aspects of the users (see, e.g. Ingwersen & Järvelin 2005; Marchionini, 1992; Bates, 1989). Contemporary interface design is linked strongly with research in Interactive Information Retrieval (IIR) that provides the necessary theories and frameworks for modelling user behaviour (see, e.g. Robins, 2000; Wilson, 2000). In the current stages of the MultiMatch design, this has been reflected in the expert user studies carried out at partner sites which has informed the current interface design.

This report describes the design and implementation *process* of the initial (or first) prototype. MultiMatch can be seen as one system being developed and deployed in two phases: prototypes 1 and 2. This is important as in itself this first prototype is not necessarily novel or unique in the functionality provided, but should be seen within the wider context of the final system which does offer novelty.

This document describes the design of the first prototype at the first cycle of design (discussed further in section 2) up to the beginning of implementation. Sections 2 and 3 focus on what will be included in the design of prototype 1. Therefore we do not report user testing (which will come after development of the working prototype is completed) and only discuss the implementation strategy decided on for interface development rather than more specific details. Section 4 will address planned testing and evaluation and the overall vision for the proposed final system. This deliverable is the final version of D6.1 (previously released as a preliminary version – D6.1.1 – at 12 months and presented as input for the first year annual review).

The code associated with the components described in this deliverable has been stored on the MultiMatch integration server hosted by OCLC PICA.

1.1 Relation with the MultiMatch architecture

The architecture consists of four sub-systems: (1) Information Indexing and Extraction, (2) Searchable Metadata Repository, (3) Enhanced Information Retrieval and (4) User Interface. This latter sub-system concerns the work described in this report and provides the MultiMatch user with an advanced interface that facilitates both simple and advanced multilingual searching of the MultiMatch content. This provides access using both MultiMatch and native metadata schemas, browsing of the content, as well as stored personalisation, search history, and preferences features. For more information of the overall MultiMatch architecture consult D3.2 (Detailed Specification of the First Prototype), an internal project deliverable. Details of the user interface sub-system are provided in section 3.1.1 of this report.

2. Design Methodology

This section describes the approach taken in *designing* an interactive interface for the first and final MultiMatch prototypes. A user-centred design approach has been advocated in MultiMatch and details for this are found in sections 2.1 and 2.2. The current design has evolved from a number of external inputs including existing literature, a review of current search engine interfaces, an initial task analysis with expert users and internal discussions, and these are described in section 2.3. In section 2.4 we describe previous designs which led to the final interface as described in detail in section 2.5. Finally in section 2.6 we discuss the next stages involved in the design.

2.1 User-centred design approaches

In the generic design of Interactive Information Retrieval (IIR) systems, Hearst (1999) suggests the interface design should follow these principles: (1) offer informative feedback, (2) reduce working memory load, (3) provide alternative interfaces for novice and expert users and (4) permit easy reversal of actions. We will meet these through an interactive design process which will focus first on what people need to do, not what the system needs to do. This will include the formulation of typical scenarios of use (in conjunction with work from WP1) that take into account both cognitive constraints and organisational/social constraints. Therefore the end-users will be involved throughout the development of the interface. The design process (typical of most lifecycles for interface design) is shown in Fig. 1.

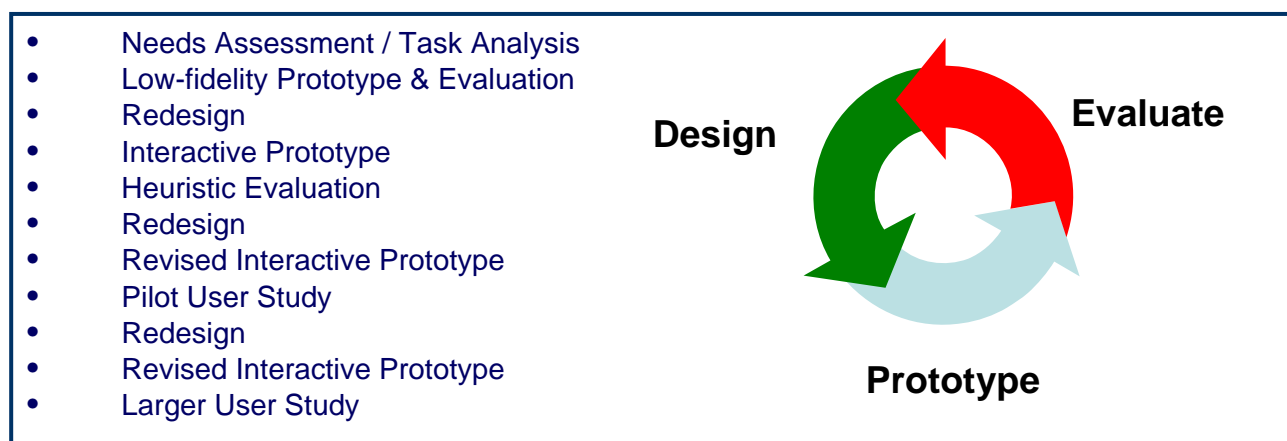


Fig. 1 An iterative design process for user interface design (Hearst’s lecture notes on UI design)

The process (Fig. 1) follows the iterative cycle of: design, prototype and evaluate and includes the main following steps: needs assessment / task analysis, creation of low-fidelity prototype (e.g. paper-based mock-ups) and evaluation, creation of interactive prototype, heuristic evaluation and pilot user study. Further user-centred design methodologies include the following (Rubin, 1994; Faulkner, 2000; Preece et al., 2002). The approach described by Rubin (1994) considers the product development lifecycle as consisting six stages: (1) user and usage needs analysis, (2) specification of requirements, (3) preliminary design, (4) detailed design, (5) product build and (6) product release. Faulkner outlines the process of usability engineering in a similar way, including the information produced at each stage:

Task	Information produced
1. Know the user	User characteristics/background
2. Know the task	User’s current task
3. Capture user requirements	User’s requirements
4. Set usability goals	Usability specification
5. Design process	Design
6. Apply guidelines, heuristics	Feedback for design
7. Evaluation with users	Feedback for redesign
8. Redesign and evaluate with users	Finished product
9. Evaluate with users	Feedback on product for future

The development model described by Preece et al. (2002) is similar to those already discussed. This is shown in Fig. 2 and clearly highlights the iterative nature of the design process. As before, this process typically starts with identifying and refining user needs/requirements, and then undertaking an iterative approach to

design involving evaluation and redesign to produce a final version. The user gets involved at various stages in the lifecycle and exactly how and when is based on the individual application.

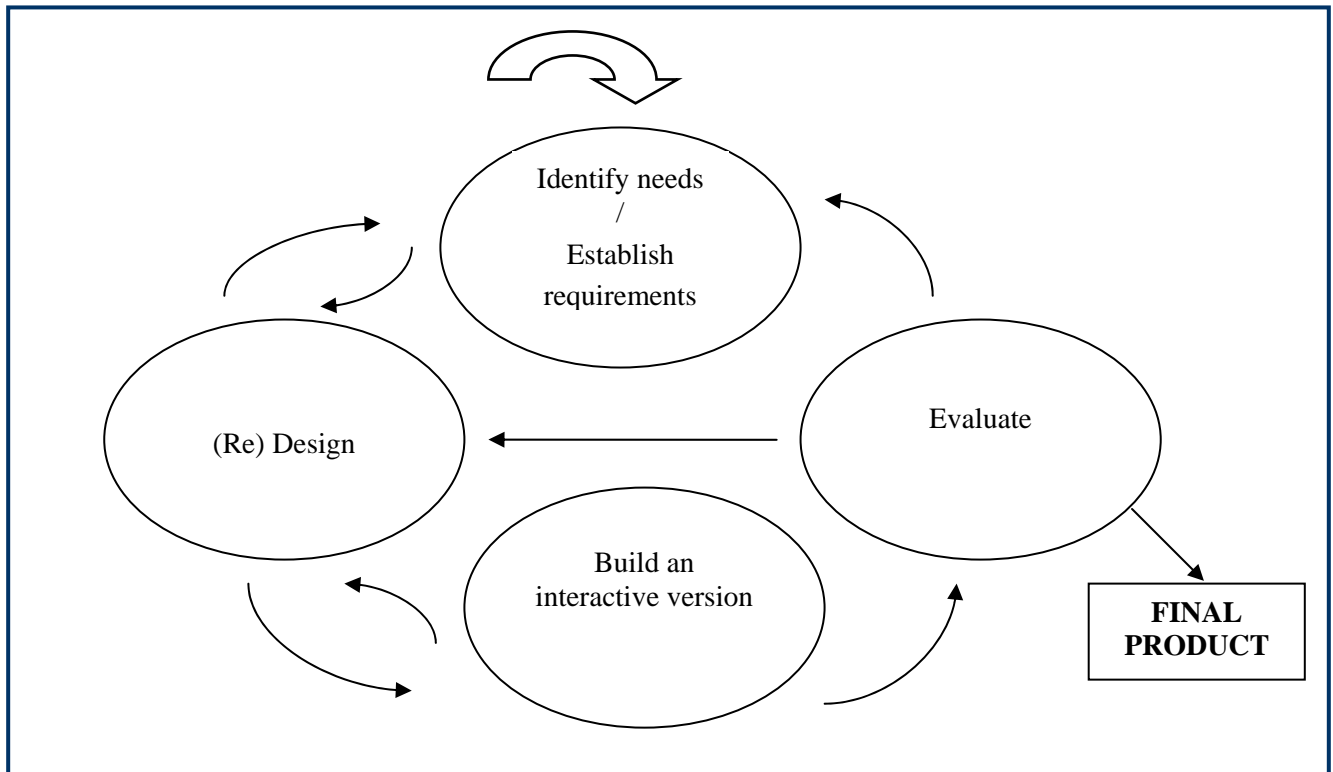


Fig. 2 Another iterative design process for user interface design (Preece et al., 2002)

2.2 The MultiMatch design process

An iterative lifecycle as illustrated by Figs. 1 and 2 has been adopted to develop the user interface for MultiMatch. This has been undertaken alongside WP1 as requirements are gathered, specified and refined. This design process involves using evidence from other sources to inform the interface design including materials such as related research projects and existing systems which exhibit similar functionality to MultiMatch, existing literature and the current functional specification. The user-centred design involves consultation with representatives from user groups (e.g. educational, tourism and cultural heritage) to develop and evaluate a series of prototypes.

Development is iterative and includes two main cycles for prototypes 1 and 2. The development cycle includes the following stages: (1) needs assessment and task analysis; (2) preliminary design using low-fidelity prototypes; (3) design and development of interactive prototype; (4) heuristic evaluation and redesign; and (5) user evaluation. Steps 1-5 will be followed for the first prototype. The second prototype will follow a similar cycle, but starting from stage (3); that is, the needs assessment will not be carried out a second time.

Overall, the design is “task-driven” and developed to meet the needs of users carrying out prototypical tasks in the cultural heritage domain (called scenarios). This defines the more general design process of the interface and does not aim to be prescriptive in a way that limits individuality from groups working on the interface. The five stages for overall design of the interface, as they relate to the steps outlined in Figure 1, are as follows:

Stage 1: Needs assessment and task analysis

The first stage in the lifecycle aims to elicit user requirements and has been (mainly) performed as part of WP1. This helps to identify the needs of both the professional/expert and the general user (the former user is

categorised into the following types: users from cultural heritage, education and tourism). Data gathering techniques being used for this stage have included questionnaires, focus groups and one-to-one interviews.

The first stage also includes a task analysis which attempts to understand what people currently do and why (Ingwersen and Järvelin, 2005; Hackos and Redish, 1998; Preece et al., 2002). Part of the task analysis involves the creation (and refining) of scenarios: an “informal narrative description” of human activities or tasks in a story or workflow (Preece et al., 2002:222-234). This is a natural way for people to describe their tasks and typically does not include information about particular systems or technologies to support the task. These scenarios can then be used during requirements analysis and to derive an understanding of the domain. In the design of the interface, the scenarios will assist with deciding functionality and evaluating the interface in later stages of the development.

Stage 2: Preliminary design of low-fidelity prototype and evaluation

The next stage takes the output from stage 1, together with additional information, and uses this to develop initial design mock-ups. Low-fidelity prototypes of MultiMatch were created which included paper-based sketches and simple low-fidelity designs based on using Powerpoint slides and HTML. Some animated video segments (implemented using Flash) were also created at this stage. The design includes additional evidence gained from past theory on interaction design, a review of literature for systems which exhibit the type of functionality expected from MultiMatch, and using informal feature analysis (or competitor analysis) to review and compare the functionality offered by existing systems and interfaces for multilingual/multimedia retrieval and cultural heritage (Goto and Cotler, 2004). Interviews (e.g. one-to-one and focus groups) and questionnaires with end-users identified by WP1 have also been used to evaluate existing systems identified to represent certain interactive aspects of the MultiMatch system, e.g. multimedia retrieval and multilingual retrieval.

Stage 3: Redesign and development of interactive prototype

Feedback gained from evaluating the initial prototypes informs the design or re-design of the interface and leads to an interactive prototype. The prototype can be used to test breadth or depth; that is used to conceptualise the wider range of functionality to be offered by the system, or focus on a particular path through the system (e.g. the result of performing a search for images). The availability (or not) of a MultiMatch collection will provide the data behind the prototype and add realism. Other sources of data can be used to simulate the type of responses expected from the system (e.g. based on the Flickr¹ or Yahoo!² Application Programming Interfaces or APIs). It is expected that the interactive prototype will illustrate how MultiMatch is able to address the needs of the MultiMatch users as defined by the scenarios gathered earlier in the project. Recommendations from usability guidelines and results from past Interactive IR (IIR) research are also used to help inform the design of the interactive prototype. For example, Shneiderman (1998) suggests the following eight design rules: Consistency, Shortcuts (for experts), Feedback, Closure, Error prevention, Easy reversal of actions, User control and Low memory burden.

Stage 4: Heuristic evaluation and redesign

The use of a heuristic study involves consulting usability experts (or advice from experts) to assess the prototypes and prompt re-design where necessary. This process iterates from the ninth point in Fig. 1 (re-design) for the first and second prototypes as outlined in the Definition of Work (DoW). The first prototype focuses on aspects of the interface which will not depend on the semantic markup and classification provided by WP4. As this technology develops and more functionality is provided, the iterative process of design, prototype and evaluate will be followed to develop the second prototype.

¹ <http://www.flickr.com> (site accessed 18/04/07).

² <http://developer.yahoo.com/> (site accessed 18/04/07).

Stage 5: User evaluation

The interactive prototype will be evaluated by representatives from the MultiMatch user groups based on completing tasks they would typically perform on the proposed system (together with using the scenarios). This evaluation stage will involve user studies and provide useful feedback to either complete a final version of the first MultiMatch prototype, or provide input to the second prototype.

Current status of design

The stages of the process completed up to the design of the first interface have thus far concentrated on gathering user requirements, conducting task analyses and on designing and re-designing low fidelity mock-ups. Currently, the design of the prototype is at the end of stage 2 and we are now undertaking the necessary groundwork to start implementing an interactive prototype. The design procedure followed has involved:

1. Gathering initial information about user requirements using the following sources:
 - Past research (Discussed in Section 2.3.1)
 - An informal evaluation of existing “competitor” websites (Section 2.3.2)
 - User questionnaires and scenarios as described in D1.2 and D1.3 (Sections 2.3.3 and 2.3.4)
2. Designing the first low-fidelity mock-ups. (Section 2.4)
3. Observing and conducting further interviews with cultural heritage professionals to enhance the understanding of their activities, needs, and desires, and to validate scenarios drafted in D1.3 (Section 2.3.5)
4. Updating designs and scenarios based on information/feedback gathered and observations made at interviews (Section 2.5)

A more detailed description of each of these stages is provided in the following sections. After the initial interactive prototype is completed, the cycle will continue with a larger-scale and more formal evaluation of the system that will then feed into the next iteration of design, moving towards the final prototype.

2.3 Design inputs

This section describes information that has influenced/guided the design of MultiMatch prototypes 1 and 2.

2.3.1 Existing literature on interactive information retrieval design

A variety of past research has been undertaken relating to general best practices for the design of interactive information retrieval systems and this has helped to influence the general structure of the MultiMatch prototype. A more comprehensive review of the existing literature for WP6 can be found in the state of the art document Deliverable 1.1.

With regards to search engine interfaces, it has been said that “Nearly every Web search engine offers users the identical search experience, regardless of the task they are trying to accomplish” (Rose, 2006: 797). In order to create a more tailored and flexible search experience, users’ needs and goals should be taken into consideration, in order to determine not only what users are searching for, but also why they are searching (Rose & Levinson, 2004).

Analysing the behaviour of users as they search for information provides informative and valuable insight into user interface design. For example, Gremett (2006) showed how an analysis of users shopping on Amazon.com revealed that in practice users would commonly mix searching and browsing while buying

online products. Marchionini (1995) calls this a *mixed behaviour* strategy of information seeking in which a user searches for information by both navigational browsing and searching a site via some explicit search tool such as a search box.

In modern IR research, more emphasis is being placed on constructing models of the search process which go beyond a simplistic view of search as a one-shot matching function between the user's query and collection of documents. Bates (1989) describes search as an interactive process that evolves in response to the information found: results from a search are not just documents, but also the knowledge accumulated along the way. Bates identifies different strategies that people follow during search including following relationships between documents (e.g. hyperlinks) or browsing over the structure of a collection. She suggests that IR interfaces would be more useful if these search strategies were supported at a higher level. Therefore, both search and browse functionalities should be present and tightly integrated, in order not to interrupt a user's exploration (Beale, 2006; Hearst et al., 2002).

Rose (2006) suggests there are three general areas in which knowledge of information seeking behaviour could inform the design of the user interface for Web search: (1) the goal of the user when conducting a search, (2) the cultural and situational relevance, and (3) the iterative nature of the search task itself. Recognising that users perform different tasks and understanding the user's goals would enable appropriate support mechanisms to be included in the interface design.

Bates (1989) suggests that search is best modelled as an iterative process and that retrieval forms part of a dialogue between the user and system to gradually refine the results. Interface support for iteration could include relevance feedback in image retrieval, or lists of related query terms for query expansion or reformulation in text searching. Rose (2006) summarises by suggesting that user interfaces should provide different interfaces or forms of interaction to meet users' search goals, allow the user to select appropriate contexts for the search (e.g. language, search options, preferences), and support the iterative nature of the search task by inviting iteration and exploration.

Hearst (1999) notes that often when searching or browsing, individuals may become distracted and temporarily follow alternate paths. For this reason, it is recommended to provide ways of recording past queries and offering a means of storing intermediate results throughout the search. This also helps to reduce short-term memory load (Shneiderman et al., 1997, in Hearst et al., 2002).

White et al. (2006) also advocate the development of systems to support users who are engaged in exploratory search activities (i.e. those without a pre-defined or specific search task). Henninger and Belkin (1996) review current systems in terms of the key interface and interaction techniques such as querying, browsing and relevance feedback (to support the iterative refinement of the user's information need). They also advocate the use of task modelling and interaction modelling as key strategies to improve the design of retrieval systems.

Regarding principles for future design interfaces, Rose (2006) advocates making different interfaces available to match different search goals. Another area to investigate is how to improve the browsing process, particularly because the common practice of displaying category lists takes up large amounts of space and often requires a user to guess which category heading will contain the related information of interest (Hearst, 1999).

In summary, the emphasis on modern search engine interface design is on understanding and modelling the user's needs, identifying functionalities to support those needs and implementing systems which support the dynamic nature of the user's tasks and searching activities. These issues have been taken into consideration whilst designing the interfaces for MultiMatch.

2.3.2 Competitor analysis

When initially considering the design of the MultiMatch interface, it was important to survey pre-existing and related sites, in order to get a feel for what sort of features are commonly offered and which functionalities are useful. This can also help MultiMatch to determine how it can differentiate itself and provide added value. Therefore, a brief "competitor analysis" of existing cultural heritage and multimedia search sites was carried out.

The aim of competitor analysis is to compare industries and features (Goto and Cotler, 2004: pp 260-276). It provides a snapshot of a marketplace from a *customer's viewpoint* including the services and features offered by companies. For example, when redesigning websites, competitor analysis is often performed as part of the web design process. The goal of the analysis is to evaluate the features, technology, content, usability and overall effectiveness of services available to customers or users within a domain. The process includes compiling a list of features in a matrix format to establish detailed site offerings and simple comparison methods. The output is a report of what works and doesn't work within each site, individual site analysis and comments, and final recommendations for possible inclusion in MultiMatch.

The approach used was an informal feature analysis (due to time, resources and budget available) which focused on identifying features and examining the user's experience. This was conducted on sites which were related to MultiMatch in some aspect, of which 56 were initially examined (located using an internet search and based on discussions with cultural heritage partners). They fell roughly into two categories:

- Cultural heritage related sites. These included, but were not limited to: official home pages of major European art museums; image and multimedia databases relating to art, architecture, or history; and portals containing links to cultural heritage information and resources.
- Multimedia search engines. These included sites on which it was possible to search for material besides webpage texts (e.g., images, video, and/or audio.)

A detailed list of the sites examined can be found in Appendix A.

Cultural heritage related sites

Currently, most cultural heritage institutions have some sort of online presence in the form of a website. Museums and art galleries have homepages and sometimes specific archives or collections that are part of a larger body have web portals of their own. These websites often provide some degree of access to the associated institution's collection in a digitised format. The degree of material that is available and the sophistication of exploration of this content vary from site to site, depending on the resources available to the cultural heritage institution in question.

However, overall, a majority of these sites do have common features which include both search and browse functionalities at the very minimum. A summary of the relative proportions of functionalities taken from a sample of 56 cultural heritage sites is presented in Table 1.

Overall, most of the sites surveyed offered basic, expected, useful ways of searching and browsing their collections, but were not very interactive or advanced. As technological capabilities have improved, there has been an increasing realisation that the current functionalities for accessing cultural heritage information online can be enhanced and upgraded, particularly in the area of browsing and exploring relationships. This is one area in which MultiMatch can provide some added value.

Table 1: A summary of the functionality of selected cultural heritage search engines

Functionality	Percent	Example
Free text search	91 %	
Browse by category	71 %	www.archinform.net
Advanced search	70 %	
News/Calendar	61 %	www.tate.org.uk
Registration/login	45 %	
Multilingual	34 %	www.louvre.fr
Geographical search / Map	29 %	http://whc.unesco.org/en/map
Shopping	29 %	
Search within results / See “more like this”	29 %	www.fotolia.com
Ability to segregate multimedia results by type (if applicable)	29 %	www.archive.org
Feedback section	23 %	
Timeline / Search by time (12 sites total; 25% of these offer search by time only, 75% have a timeline (2 of the 8 were interactive))	21 %	www.birth-of-tv.org
View results in popup window	21 %	
Change results layout (order by..)	21 %	www.artandarchitecture.co.uk
Hierarchical browse	20 %	http://www.staffspasttrack.org.uk/
Sitemap	20 %	
Controlled vocabulary	9 %	www.tate.org.uk
Colour/layout search	7 %	www.hermitagemuseum.org
Query translation	5 %	www.fotolia.com
Multimedia results arranged by type	5 %	http://ec.europa.eu/avservices/home/index_en.cfm
Faceted browse	3%	http://orange.sims.berkeley.edu/cgi-bin/flamenco.cgi/famuseum/Flamenco
Allow user annotation	2%	BRICKS workspace

Kravchyna (2004) surveyed five categories of users to assess their information needs when using museum websites. The categories included were (i) museum professionals, (ii) scholars/art historians, (iii) the general public, (iv) university students, and (v) high school teachers. Across all groups, primary purposes for using museum sites were to determine the main exhibits and activities of interest, to gain knowledge about museum collections, and to learn of any upcoming activities by consulting any available event calendars. Additional priorities that were unique to the scholar group were related to gathering information for research (i.e. looking for specific images or looking for textual information on a museum object). Therefore, while some needs crossed group boundaries, there were also group-specific requirements.

There are certain functionalities which most, if not all, of the sites surveyed had in common. These include features such as offering both a basic and advanced search, as well as some way to browse by categories. The most popular/frequently observed functionalities of the 56 sites included in this initial survey are shown in Table 1.

Multimedia search engines

Multimedia search engines can offer a variety of possible media formats to be searched. Based on a sample of 16 online multimedia search systems, Table 2 shows a breakdown of the number of combinations for each type (image, audio and video).

Table 2: Search Category Combinations Supported by Popular Internet IR Sites

Media types	Number of sites	Examples
Images Only	9	www.live.com www.clusty.com http://www.google.co.uk/imghp?hl=en&tab=wi&q=
Images, Audio, Video	4	www.alltheweb.com
Video Only	2	www.youtube.com
Audio & Video	1	www.singingfish.com

Table 3: Example online multimedia retrieval systems

Collection holdings	Percentage	Example
Images	86 %	See above
Audio	36 %	
Video	50 %	
Tabs for different media	60 % (3 of 5)	www.altavista.com
Searching functionalities		
Free text search	100 %	
Advanced search	53 %	
Search all types of media at once	20 % (1 of 5)	www.singingfish.com
Browsing functionalities		
Category list	14 %	www.youtube.com
Hierarchical browsing	0 %	
Tag cloud	7 %	www.youtube.com
Results		
Displayed in grid / rows	100 %	
Other display	7 %	www.live.com (infinite scroll bar)
Ability to refine search / change result layout	57 %	www.creative.gettyimages.com http://www.google.co.uk/imghp?hl=en&tab=wi&q=
Multimedia results segregated by type	40 % (2 of 5)	www.altavista.com
Recommendations / "more like this"	14 %	www.youtube.com
Clustering of results	6 %	www.clusty.com

The main points to note are:

- Of the six sites that had content in more than one medium, only one of them (Singingfish) offered the possibility of searching several media types at once. For the rest, search had to be limited to a specific type (i.e. image OR audio OR video, but not a combination.) The results of the Singingfish search, however, are not separated by type.
- Free text was the predominant means of searching. Only one site (YouTube) had the possibility of browsing by category.
- Most of the sites followed a similar layout and respected similar conventions. They were simple and based on the Google interface model.
- In terms of results presentation, again, clear conventions prevail. Image results are displayed in a grid and Audio/Video results are shown as a list, often with a thumbnail and a brief description.
- The one unique presentation feature is Microsoft’s “infinite scrolling bar”...other sites are limited to displaying between 10-60 results per page.
- About half of the sites allowed refinement of the search in some way
- Only 2 sites provided recommendations or “more like this...”
- There is a large degree of homogeneity among these sites, which suggests that there are certain conventions with regards to design and presentation that should be followed.

Browsing

Next, 19 different cultural heritage sites were selected from the larger set of 56 and analysed with respect to browsing functionality. The 19 specific sites were selected based on our decision about the kind of browsing they offered and their relatedness to MultiMatch. The sample cultural heritage sites revealed the different categories provided to allow users to browse a large collection. The relative frequencies of categories found on 4 or more sites are summarised in Table 4.

Table 4: Browsing categories offered by various cultural heritage sites

Category	Number	Percentage
Artists/Creators	14	73.7
Period & Style; Genre	11	57.9
Places	10	52.6
Media/Category (painting, etc.)	10	52.6
Year/Time	10	52.6
Keyword (Subject)	10	52.6
People	5	26.3
Country	4	21.0
Materials	4	21.0
Concepts/Themes	4	21.0
Museum Name	4	21.0
Title	4	21.0

Multilingual access

The Minerva survey (2006) examined the types of monolingual search functionalities provided by 671 European cultural and museum websites. Overall, it was reported that 51% of sites used no search tool at all, 24% offered free text indexing and 14% provided controlled vocabularies (some sites offered both). However, it is unclear how many of these search tools were available in more than one language. Marlow (2006) reviewed the functionality of a number of online museums and art galleries (shown in Table 5).

Table 5: Functionality offered by various online museums and art galleries (Marlow, 2006)

	Welcome pgs for f.l. (if more than one page available)	Multilingual search of site (can locate material written in other languages)	CLIR (query translation)	Controlled vocabulary	Free-text Search	Easy to switch languages	Easy to return to original language
Tate		●	○	⊙	⊙	○	○
Online							
British Museum	●	●	○	⊙	⊙	●	○
National Gallery		○	○	⊙	⊙	○	●
V&A Museum		●	○	○	⊙	●	●
Natl. Portrait Gallery		●	○	⊙	⊙	○	●
Louvre	●	●	⊙ Lafayette database ○ Atlas database	● - Kaleidoscope	●	●	●
Guggenheim Bilbao	●	○	○	○	○	○	○
van Gogh Museum	●	●	○	●	○	⊙	●
Rijks- museum		○	⊙	●	●	○	○
Centre Pompidou			●	⊙	⊙	●	
MoMA		●	○	○	⊙		
Met New York		●	○	○	⊙	●	●
Guggenheim New York		○	○	⊙	⊙		
24 Hr Museum	⊙	⊙	○	○	●	○	●
Easyart.com	●	○	●	●	●	○	

● - multilingual offering ⊙ - only in main language ○ - not offered

It is noteworthy that very few of the sites listed in Table 5 actually offer cross-language search functionality to users. This is typical of what generally obtains for most Internet search engines which tend to lack

multilingual search facilities. The majority of cross-language research remains in the theoretical domain and has not often been implemented or made accessible to the end user (as noticed by Peters and Sheridan (2001) and unfortunately still true today). Perhaps this is surprising given the motivation for multilingual search in (Oard, 1997), but Evans (2006) indicates that factors such as the limited effectiveness of translation, the lack of real-world user need for this kind of functionality, the complexity in effectively providing multilingual interaction and the additional cognitive burden pressed upon the user are all limiting factors.

Cross-language information retrieval systems

Several experimental cross-language information retrieval systems have been built; as mentioned in the state of the art document (Deliverable 1.1), these include MIRACLE (Dorr et al., 2003; He et al., 2003), Keizai³ (Ogden et al., 1999), MULINEX⁴ (Capstick et al., 2000), ARCTOS⁵ (Ogden & Davis, 2000), and CLARITY (Petrelli et al, 2006). However, most of these (with the exception of CLARITY) do not have demos that are publicly accessible via the web.

As a result, it is difficult to gain inspiration from previously-used approaches to providing and displaying feedback on query translation. There are a few sites which utilise automatic query and/or results translation but do not provide the user with any feedback regarding how the system translated the query or enable any changes to be made. These sites include:

- International Architecture Database (<http://www.archinform.net>)
- Gaumont Pathé video archive (<http://www.gaumontpathearchives.com>)
- Roger-Viollet photo archive (<http://www.roger-viollet.fr>)

Two sites were identified which do provide the user with feedback on their query translation:

- Cultural Heritage Language Technologies project (<http://www.chlt.org/mlir/mlir1.php>)
- Turing Center Cross Language image search (<http://knowitall-3.cs.washington.edu/panimages/>)

However, these sites deal only with one type of media (either text or images) and thus do not provide an example of dealing with query translation in a multimedia retrieval setting.

Good examples

The following sites were highlighted as good models to follow and provided inspiration for some aspects of the MultiMatch design, either through a specific feature they offered or their general package of functionalities. Screenshots illustrating aspects of the starred sites can be seen in Appendix B.

- Courtauld Institute of Art – Overall structure, browsing, folder options
- *Archinform – Overall structure, browsing, aggregation of information
- Beeld & Geluid website – Results refinement, keyframe display
- *Flamenco faceted browsing – Model for more advanced browsing features
- *eCHASE demo – Overall structure, results presentation
- *Alinari site – Detailed image views, browsing
- Fotolia – Main page, random overview, results refinement
- *CLARITY demo – Cross-lingual interaction
- *Grokker search engine – Results presentation and filtering
- *A9 search engine – Results presentation (in columns) and filtering

³ <http://kythera.nmsu.edu:8099> (Site Accessed: 3/10/06).

⁴ <http://mulinex.dfki.de/demo.html> (Site Accessed: 04/10/06).

⁵ <http://crl.nmsu.edu/~ogden/i-clir/cltr-interactive/arctos/page1.html> (Site Accessed: 04/10/06).



- *MIND project / Creative Commons search – Overall multimedia results layout including use of tabs
- *Podzinger/Video Now (Nexidia) – For audio and video search

As a next step, (currently in the process of being planned), we will show a selection of pre-existing sites such as some of those mentioned above to a group of general users. The objective will be to elicit their feedback about which types of sites (and functionalities thereof) people prefer to use.

2.3.3 User requirements and scenarios (D1.2)

Deliverable 1.2 focused on gathering user requirements for the MultiMatch target audience. One hundred person-to-person interviews were conducted with domain experts (educational, tourism, cultural heritage professionals) in order to collect their opinions and needs. The interviews were conducted mainly in face-to-face mode using a questionnaire, and backed-up by a set of scenarios and a vision document in order to give the respondents an idea of the proposed system functionality. In addition to the interviews, there was some analysis of logs from the WIND and Alinari portals and examination of the results of previous user studies in the cultural heritage domain. A large number of potential requirements were identified and analysed.

The principal findings of this analysis can be summarised thus:

- CH professionals do use the internet widely and as part of their daily work routine but they currently depend largely on generic search engines to find the information they need
- they want to query using natural language and familiar Boolean operators
- they would like full capabilities for multimedia retrieval (i.e. images and video as well as text) but, in most cases, are only accustomed to executing text searches
- their main focus appears to be on works of art and their creators, with all associated information, such as critical reviews, information on exhibitions, different versions of same document
- they tend to be frustrated by the volumes of information available on the same subject and would find information filtering, clustering and aggregation functionalities very useful
- they demand high precision of results and need to know the source and level of authority
- they need to be able to save both queries and results for future processing and reuse
- they tend to restrict their searches to their own language plus English, thus missing information only available in other languages
- if multilingual search was available, they would like to have the results associated with descriptive snippets in their own language (preferably) or English (optionally).

Additionally, several scenarios of use were created. Scenarios make it possible for users to envisage the potential of the system, and thus identify and formulate requirements that are not based only on current practice.

The scenarios cover the three user groups being targeted and have been used to communicate the MultiMatch goals and proposed functionality with respect to these user groups. They also help to clarify within the project the most desirable system features and design options. Some of these scenarios were chosen for validation with cultural heritage experts at a later stage in order to affirm their authenticity. These four scenarios are as follows:

Scenario 2: Juan is looking for flowers

Reference	Description
Scenario	<p>Juan is preparing an essay about the use of flowers in art. To do this he needs to mine the web searching for web pages which talk about authors that use flowers in their artworks and all the artworks representing any kind of flowers. Initially Juan decides to use Google and Yahoo! to perform this task but he quickly realizes that, although these search engines perform a good retrieval on the query "flowers", he has to do extra mining over the search results to filter and classify them. Moreover, he has to extract the critical information (i.e. names and titles) manually from the final selected relevant pages. He is upset because he has to finish the work for next Monday and he thinks that it is going to be impossible. Fortunately, he remembers a comment about MultiMatch made to him by his girlfriend Ruth, so he decides to test the search engine to see if it could help him to accelerate his work. Typing the query "flowers" in MultiMatch he realizes that MultiMatch not only retrieves web pages for him (i.e. like Google or Yahoo! do) but also a complete list of authors and artworks related with flowers and the complete list of cultural heritage sites where these artworks can be visited. He is very happy because he has obtained the complete list of items he was looking for with only one query.</p>

Scenario 8: Giovanna publishes her book

Reference	Description
Scenario	<p>Giovanna is a lecturer at Florence University. She is collecting images for a book on Italian art. She has made iconographic searches in many Stock Images and needs to produce a dummy with images and text to show the publisher.</p> <p>Giovanna searches for relevant images and texts on the MultiMatch site. MultiMatch can:</p> <ul style="list-style-type: none"> • Associate images with related text • Order images and text with respect to relevance • Provide annotations relative to the digital quality of the images, colour and orientation • Give information on IPR (copyright of images, copyright author, restrictions) and the relative reproduction fees <p>Giovanna prints the dummy produced with MultiMatch and shows it to the publisher</p>

Scenario 10: Daniela is preparing an exhibition

Reference	Description
Scenario	<p>Daniela, who is preparing an exhibition, needs a list of low resolution files on Jean Michel Basquiat's works of art. She searches for them using MultiMatch: the results should be classified by country and by place; and must include only events between 1971 and 1984;</p> <p>She also needs descriptions of each image in two different languages as she is preparing the exhibition in Italy and will organize it at the Spanish embassy there. She wants to know the technical costs, the right-of-use costs and the mode of payment to use the MultiMatch page as a brochure to be distributed at the event.</p> <p>Some paintings will be sold during the exhibition and Daniela has the possibility to input her comments and costs on the page generated by the MultiMatch</p>

	<p>interface.</p> <p>Daniela found an interesting documentary and video interview with Madonna, the pop singer, who had been Basquiat’s girlfriend for three months on MultiMatch. She would like to show it during her presentation and contacts the author. Finally, she notices that one of Basquiat’s masterpieces is missing from the list. She knows that the painting reproduced the Mona Lisa in some way. So she retrieves a picture of the Mona Lisa and by similarity search retrieves Basquiat’s Mona Lisa 1983. MultiMatch also provides enough information for her to contact the owner of the painting so that she can request permission to exhibit it..</p> <p>Daniela finally wonders if she can advertise the exhibition in some way on MultiMatch to her address list and to a wider group of users. MultiMatch allows her to share a page that she has designed with the exhibition brochure (this page will be indexed for those who will search events in the fixed event period).</p>
--	---

Scenario 12: Leonardo is a content broker at BigSearchImages

Reference	Description
Scenario	<p>Leonardo is a content broker at BigSearchImages Ltd. He helps clients to find special content in BigSearchImages’ image base. In the past Leonardo has used a personal set of archives (Getty Hulton, Alinari, Corbis, Bridgeman, etc.) to find the requested contents. He had to search the same contents in many different vertical repositories; this process was very time-consuming. He now uses MultiMatch to collect contents from different authoritative sources from a single search interface. MultiMatch also stores the queries done by Leonardo in the past.</p> <p>One client, who is writing a book, has just asked for some historical photos of every-day work in early industry, which she needs as illustration for her book. Leonardo can retrieve pictures from the MultiMatch content base using a thesaurus. It contains some keywords that match with the client’s request. Leonardo can browse the resulting subset of images or further limit it by adding query expressions.</p> <p>Once Leonardo has found an image that fits, he asks MultiMatch to search images that are similar to this example. MultiMatch searches by comparing some graphical features of the example image (colours, layout, objects such as human figures) but also looking for semantically related descriptors in the metadata. After a short time, MultiMatch starts to list thumbnail images that are similar. Some are indeed similar to what Leonardo expects, others are not. He picks another two suitable ones and asks MultiMatch to search using the 3 selected examples. Now the images retrieved by MultiMatch fit closely to what Leonardo is looking for.</p> <p>Leonardo’s next client asks for 20 images that show well-known sites (such as Niagara Falls or the New York City skyline) and have a certain atmosphere about them: the client needs cold / calm colours and layout. Leonardo starts his retrieval by thinking of well-known places, browsing photos of these, selecting some that are in landscape format and (to him) radiate calmness. When he has found a few such images showing different places he asks MultiMatch to retrieve images that are visually similar to these examples.</p>

2.3.4 Functional specifications (D1.3)

Arising out of the initial user vision requirements along with the MultiMatch project description and common vision, a set of functional specifications was drafted (Deliverable 1.3). These described what the

first prototype system was expected to do, in general. These were as follows (taken directly from Deliverable 1.3):

Levels of search interaction:

1. **Default level.** MultiMatch will provide users with a default combined search on text, image and audio-video search services. The first prototype will retrieve for a single query all the text documents, still images and audio-video, which will be presented separately to the user. It is not planned to provide search results with full embedded metadata (i.e. which means that search results will not be rendered as cultural objects) but, when possible, the first prototype will make use of CIDOC manually annotated markup to improve the information stored in search results. Issues related with metadata-based search (such as creators and creations) will be addressed in the second prototype.
2. **Specialized level.** As the default level will make use of a simple approach to standalone text, image and audio-video retrieval, it is also planned to provide MultiMatch with these search facilities as standalone search services. Image and video retrieval will also provide the user with text query based and image query based retrieval facilities, although the last one will only be provided for contents indexed within the MultiMatch collection. Finally, the initial manually annotated categories will be used by the first prototype to provide a simple browsing functionality of MultiMatch contents.

The first prototype will provide the following search features for both searching levels:

- Relevance feedback using search results.
- Image relevance feedback for still image and keyframe search results.
- Boolean search. Typical of these boolean operators would be “AND”/ “OR” searches.
- Visual search using low-level features (e.g. colour histograms). The first prototype will only apply this search functionality within the indexed document collection, not for external images.

Search modes:

1. **Default mode.** MultiMatch will implement a default search mode that will provide the user with the capacity to enter a free text query (as a basic search function). MultiMatch will perform monolingual retrieval using the user's native language as a default mode.
2. **Advanced mode.** MultiMatch will allow the user to customize his searches to retrieve information in different languages. At this point, it is planned that the first prototype will perform translanguagual searches in the four selected project languages. The results in different languages will be presented to the user as separate document sets. Filters depend on the information extracted from data indexed; for this first system it is planned to include filtering only for file type and for file size.

Multilingualism will be enabled at this first stage by translating the query from one language to another using a simple combination of machine translation and commercial multilingual/bilingual dictionaries.

Retrieval Functionalities

The first MultiMatch prototype will perform the retrieval process using the following document sources:

- **For default search:** MultiMatch will retrieve information from the following sources using its corresponding search services:
 1. Cultural heritage web sites indexed (text search).
 2. Wikipedia items indexed (text search).
 3. Biblioteca Virtual Miguel de Cervantes contents indexed (text search).
 4. Still images indexed from web pages and/or Wikipedia (image search)
 5. Alinari still images repository (image search)
 6. Sound and Vision videos (video and audio search)
- **For specialized search:** MultiMatch will retrieve information from the following sources:



1. **For image search:** MultiMatch will retrieve images from the Alinari still images repository, images indexed from web pages and/or Wikipedia.
2. **For audio-video search:** MultiMatch will retrieve videos from the Sound and Vision repository.

Also, the first prototype will implement the following features for default and specialized searches:

- **For default search:** MultiMatch will provide the user with the following lists of items:
 1. A list of documents, ranked according to some relevant criteria, and retrieved from the set of text sources indexed in MultiMatch. When possible, the retrieved information will be combined with metadata to provide the user with more comprehensive information about the documents. Text documents will be summarized using query-biased snippet generation techniques which display the text in its original language or a summary translation at least in English. Search results in different languages will not be combined in the first prototype. They will be retrieved and ranked as separate document sets.
 2. A list of still images, ranked according to some relevant criteria, and retrieved from the set of image sources indexed in MultiMatch. When possible, the retrieved information will be combined with metadata to provide the user with more comprehensive information about the images.
 3. A list of videos, ranked according to some relevant criteria, and retrieved from the set of video sources indexed in MultiMatch. When possible, the retrieved information will be combined with metadata to provide the user with more comprehensive information about the videos.
- **For specialized search:** MultiMatch will allow the user to perform standalone image and video searches providing a list of the following items:
 1. The image search service will provide the user with a image list, ranked according to some relevant criteria, which will give access to image thumbnails, original images, sources of the images (i.e. web pages where they are embedded, video files, image databases).
 2. The video search service will provide the user with a video list, ranked according to some relevant criteria, and retrieved using text retrieval techniques based on spoken audio transcriptions.

Interface Functionalities

The MultiMatch first prototype will provide the following interfaces:

- **Graphical Interface.** MultiMatch will implement an initial web based client interface to support the following activities: a default search, a specialized search for images and video, and a browsing facility.
- **Application Programming Interface.** MultiMatch will provide third-party users with an initial API to access default and specialized search level services. This API will be also used by the MultiMatch web interface to perform user searches and to retrieve results.

The main functionalities for the graphical interfaces will be the following:

- **User access privileges.** The user will use the MultiMatch system as anonymous or registered user. Registration implies:
 1. The user will receive a user account protected by username and password.
 2. The user account will give access to his search history and language preferences.
 3. The use of client stored data will enable the configuration of the MultiMatch interface according to the user's preferred languages.
- **Browsing facilities.** The user will be able to navigate and visualize the following MultiMatch contents:

1. The words most frequently appearing in the collection (e.g. tag cloud facility).
 2. A collection overview (e.g. a display/collage of items randomly chosen from the collection)
 3. The user will be able to browse the collection, when possible, by categories, based on pre-existing metadata information (this facility will be only available for specific contents indexed in the first prototype).
- **Language facilities.** The user will have access to the following language facilities at query time:
 1. The user will be able to retrieve documents in the language of the query as well as others (CLIR).
 2. The user's query will be translated into the other languages of the project and suitable matches will be identified.
 3. The user will be able to view possible translations (in the case of an ambiguous term) and select the preferred version.
 - **Presentation and organization facilities.** The following shows the main presentation functionalities related with the user interface:
 1. Results will be displayed in an arrangement that is deemed to be preferred by users (e.g., in a grid, slideshow format, etc.)
 2. The user will be able to control or re-arrange the results display
 3. The user will access text results as a summary showing the query in context (e.g. the query will be highlighted wherever it appears)
 4. The user will be able to sort or re-organise the results in a simple way (e.g. by size/file type)
 5. The user will be able to make an indication of an item's relevance using positive or negative relevance feedback.
 6. As another form of relevance feedback, the user will be given the option of selecting relevant objects or placing them into a "workspace" to signal relevance feedback.
 7. The user will be able to group or cluster the results based on simple criteria (e.g. colour)
 8. The user will be able to read a translation of foreign language results; this translation will be displayed to the user based on his/her preferred presentation style.
 9. The registered user will be able to perform manual annotation of results.

Overall, the functional specification served as an initial guide for the first prototype design by outlining the various features that needed to be incorporated into the interface.

2.3.5 User groups and general user survey

The identification of users was aimed at defining the main characteristics of the user groups that will influence the way the MultiMatch service can be used and the general requirements of the system. We identified three main MultiMatch user groups: educational, cultural tourism (also known as general users), and cultural heritage professionals. Further details of the characteristics of these groups can be seen in Deliverable 1.2.

As soon as the first MultiMatch prototype has been released, we will enlarge the membership of the user groups in order to validate a full set of user needs for the system. Workshops, newsletters and events will be used as instruments to get in touch with the user groups and interview them. However, thus far, initial input has been gathered from the various user groups via interviews and questionnaires.

Initial (see Section 2.3.3) and subsequent (Section 2.3.6) interviews focused on fixed user groups of expert consumers of cultural heritage information in order to define common sets of approaches for carrying out the user requirement studies and system evaluation. The investigation was also extended to include non-professional users via an online questionnaire.

With regards to this latter study, it was important to get an idea of how the general public uses the internet to meet their multimedia needs, particularly in the cultural heritage domain. To investigate these issues, an online questionnaire was published on the community homepage of Libero, a major Italian internet portal. The questions were available only in Italian and they addressed various issues ranging from demographic

questions (age, language skills) to multimedia use to preferred ways of grouping information. In total, 414 people responded to the survey. The results for selected questions can be found in Appendix C. Overall, they indicate that texts and images are the most widely searched-for type of media and that specialised sites (museums/universities,) specialised search services, and Wikipedia are viewed as useful tools. Responses to the question about possible ways of linking or grouping relationships could be useful in influencing the semantic navigation in the second prototype.

2.3.6 Task analysis for cultural heritage experts

According to Hackos & Redish (1998), conducting a task analysis helps to determine specific features to build into a product by helping understand the type of work that target users do. Therefore, interviews and site visits were carried out in order to supplement the initial requirements gathered in Deliverable 1.2. By talking to and observing users in their work settings, it was possible to gain a deeper understanding of the resources they use and the types of things they do on a regular basis.

Interviews were carried out with fifteen individuals at three different institutions: the University of Sheffield, Alinari, and Beeld en Geluid. Of these, seven were academics (university professors in the fields of arts and heritage management, architecture, archaeology, French literature, and history,) four were image professionals who worked for a major photographic archive, and four were video professionals who regularly searched through various motion picture archives. This selection provided a variety of perspectives relating to the cultural heritage field and expanded upon past studies by addressing a broader range of tasks and topic areas.

The interviews began with a set of questions designed to gather information about work tasks and requirements. This was then followed by an informal evaluation of two websites. The interview questions focused on topics including a general overview of the interviewee's work, tools used, use of the web, use of multimedia material, concrete examples of past search processes, ways of improving the current search process, and language issues relating to search. The websites evaluated were selected as examples of unique and potentially useful ways of organising information: these included the Flamenco system for faceted browsing and Clusty, a web search engine that groups results into related clusters (see Appendix A).

All interviewees were more likely to at least begin with a rather specific, directed task which they would use to initiate a search (rather than beginning with a random browse.) Nonetheless, they responded fairly positively to the idea of faceted browsing and the clustering of results. However, watching them interact with the Flamenco faceted browsing system revealed that such a system is not always immediately intuitive to use and therefore would require explicit explanation or assistance to help the user to begin to use it.

Table 6 - Summary of findings of cultural heritage professional interviews

	All	Academics	Image Profs.	Video profs.
Tasks	Looking for information on websites Looking for background information on a particular topic before investigating it further	Preparing presentations for teaching Research and publication (books, articles)	Searching archives for photos on behalf of clients	Searching archives for videos on behalf of clients
Multimedia types searched for on the internet	Texts – 15 Images – 13 Video – 5 Audio - 1	7 6 (used for teaching, not research) 1 0	4 4 0 0	4 3 4 1
Tools used	Web Contacting colleagues/personal networks Printed material from physical archives			
Websites used	Google/Google Images Specialised sites/databases/archives Museum or other commercial sites			
Languages	11 polyglot 2 monolingual (can only read in one language)	3 2	4 0	4 0
Search categories:	<p><i>Names:</i> e.g. Terragni, Raphael, Jacques Varèse</p> <p><i>Places:</i> e.g. centre of Moscow, Nijmegen, Salamanca</p> <p><i>Dates</i> (usually in combination with another category): 14th century, 1949-51</p> <p><i>Titles:</i> e.g. “The Man Midwife,” “Marriage at Cana,” “Nuit et brouillard”</p> <p><i>Subjects:</i> e.g. illustrated punch bowls, emotions (sadness), people judging things, illustrations to accompany a magazine article on “Man and God.”</p> <p><i>Combination of several categories:</i> e.g. “14th century Humberware drinking jug”; “Henri Matisse’s murals and stained glass of Rosary Chapel of Vence, 1949-1951”; “Images or video of land mine clearing in Holland after WWII (1945-48)”</p>			

For example, many people did not initially realise that clicking on a second facet meant that they were browsing within the context of the first facet. Response to the general idea of faceted browsing was positive, but as one person mentioned, “This isn’t hugely user friendly.” Other criticisms of the Flamenco demo were that it was impossible to enlarge the image thumbnails (thus rendering the system “useless,”) stemming of terms led to inappropriate categorizations (i.e. a picture of Henry Ward Beecher was associated with “hen” and “bee,”) and that the facets weren’t always very helpful.

Attitudes towards the Clusty search engine were more positive: “I like this, it’s much more sensible,” and “Google has too many results. This way of dividing things, it’s really important.” However, once again, some of the clustering categories were confusing: “I don’t quite understand some of the categories,” and “How does it decide what the categories are?” were two comments made.

In general, these opinions indicate that faceted browsing or clustering of results can potentially be very useful functionalities if they are used properly. It will be important to choose useful, clear categories and to provide some sort of support to help a first-time user learn how to navigate the facets in the most appropriate way (implementation planned for the final MultiMatch prototype).

Suggestions for improvement

Some interviewees mentioned desired functionalities to help improve their searching process. The academics’ comments were mainly related to internet search in general:

- providing good navigation, query expansion suggestions, and the ability to enter more technical terms into a search.

One individual was particularly interested in improved image content search that did not depend only on verbal queries based on words present in titles or metadata.

Comments made by the image professionals included a desire for:

- more advanced results filtering options (e.g. clustering photos into categories based on whether they depicted famous or non-famous people.)
- eliminating duplicate results from Google image search
- being able to see brief artist summaries with basic, aggregated information about an artist and his/her works

Video professionals expressed a wish for:

- the ability to jump directly to the place where a given word was spoken in a video – 3 people
- improved methods of quickly browsing a video’s content online (without the delays caused by buffering.) – 2 people
- more detailed content descriptions of each shot (i.e. a 4-second top shot of a red car in a roundabout, although creating such descriptions would be time-consuming and likely need to be done manually – 1 person

Implications for system design

Overall, the interviews highlighted the similarities and differences in user needs across job categories. In summary, improvements could be made to the status quo by addressing people’s needs and wishes. Implications were derived from comments made and descriptions of tasks performed. Many of these areas will not be implemented in the first prototype and therefore relate more to the final system. Overall, these user-inspired ideas support and tie in well to the general common vision of the MultiMatch project. Those which relate to prototype 1 include:

- Aggregation of multimedia material that is currently only found dispersed across the web (“There are no databases that store that material in any comprehensive way, so I find them hidden in various corners, or that’s how it feels to me.”)
- Enabling the filtering of results by source/domain (in order to exclude unrelated or un-authoritative information)
- Providing added value and a time-saving technique by enabling automatic dictionary-based query translation (so individuals don’t need to physically look up the words themselves)
- Offering the possibility of machine translation of results (while it may not be used, there are some cases where it could be viewed as “better than nothing,” if the individual has no understanding of a given foreign language.)

Other comments that can be taken into account for the second prototype include:

- Supporting the common search facets of who, what, when, and where, but also providing summaries via “cultural objects” and contextualizing artists and artworks in relation to other people and things
- Facilitation of content-based search: although not frequently used, this could still be utilised and improved as a means of locating material, particularly when the task is related to finding something to express a feeling or mood, (e.g. within an advertising context.)
- Allowing multimodal querying (an individual could submit a still image as a query to look for associated videos.)

Scenario validation

When and where appropriate, individuals were shown some of the scenarios crafted for D1.3 (outlined above in section 2.2.3) and asked to comment on how realistic they were. Comments made included:

Scenario 2 - Juan is looking for flowers	<ul style="list-style-type: none"> • Too general (a student would probably have a more specific task, i.e. flowers in contemporary art, or some more limiters of some sort.) • Nowadays people would probably use Wikipedia (though teachers feel the need to warn about its authoritativeness.)
Scenario 8 - Giovanna is publishing a book	<ul style="list-style-type: none"> • Isn’t quite what we do. Sounds fairly generic • If I want an image for publication, it’s a unique or very specific image, therefore I want a Humberware drinking jug not a medieval pot. • Order results by relevance, implies there are a lot of results. Could be useful if she’s looking for images of the Holy Madonna and child but only work if you’ve got a large quantity of results • Prices is useful; The cost is important because as authors, we’re paying for this (the copyright)
Scenario 10 – Daniela is preparing an exhibition	<ul style="list-style-type: none"> • I wouldn’t really rely on a search engine to find all the images for me because it’s quality...I think there are other ways to search for images as well; it depends on how they’ve been catalogued. • I doubt if anyone would use it as a solo(?) You’d be much more likely to know who’s got what through your networks, there are other ways of doing this • I would worry about this because a number of these works

	are going to be in private collections, are they gonna be digitized.
Scenario 12 – Leonardo is a content broker	<ul style="list-style-type: none"> • Scenario is similar to the type of work we might have to do. • This sort of “double request” could be good (i.e., using both words and related content) but it could also be bad: for example, if everything is automatically generated, you might lose or miss out on some relevant things.

New scenario generation

Overall, the comments made suggested that the scenarios were not completely realistic and representative of how different classes of users might use a system like MultiMatch. Based on some of the actual tasks observed, the scenarios were revised in an attempt to create four situations covering a range of user classes and tasks and involving different media types and languages. They are also based on the type of content available to the first prototype system, and are shown in Table 7.

Table 7 – Revised scenarios based on comments and observations from interviews

User type	Task	Media and languages involved
CH professional	Searching for video footage on Pier Paolo Pasolini, needs to gather background info on who he was	Text, Images, Video English, Dutch
CH professional	Looking for images of (non famous) people drinking coffee that capture a certain emotion	Images English, Italian
Academic	Preparing a presentation on Don Quixote and how it has influenced the arts	Text, Images, Video, Audio (?) English only
Cultural tourist/General user	Planning a visit to London, wants to know about museums to visit, what can see while there	Text, Images, Audio (podcasts) Spanish, some English

The combination of the several factors mentioned in Table 7 (literature, competitor evaluation, functional specifications, and two phases of user-based requirements gathering) led to a solidification of the features and functionalities to be offered in Prototype 1. These will now be discussed in further detail.

3. User interface designs

As described previously, a series of low fidelity paper and HTML-based mock-ups were constructed throughout the design process in order to provide an illustration of how the various functionalities might be depicted and interact with one another.

The progression of these designs may be seen in Appendix D. Versions 1 and 2 were based on ideas proposed at the MultiMatch kick-off meeting and supplemented by ideas from the literature, past systems, and existing sites. Initially, USFD and UNED created versions independently of one another. This process is known as parallel design and it is often favoured because it is a relatively fast way of generating diverse options and exploring as many ideas as possible before deciding on a course of action (<http://www.usabilitynet.org/tools/parallel.htm>). These two different sets of interfaces were shown to some participants at the expert site visits and opinions were elicited.

Next, elements from the two designs were merged into a common interface (Version 3), based around the functional specifications and user requirements. Additional changes were made as a result of making comparisons with pre-existing sites and based on further reading.

A set of demonstration screenshots linked to the revised scenarios were created and then shown at the MultiMatch general meeting (Version 4.) As these were based around a task progression, they provided a more coherent, unified design that illustrated the various aspects of the system. As a result, these designs elicited further comments and suggestions about the technical feasibility and desirability of what was proposed.

Finally, a design meeting involving UNED and USFD was held, which helped to solidify some outstanding uncertainties about aspects of the interface, most specifically the difference between the default and specialised search result presentation (Version 5.) There are still some minor unresolved issues that will be discussed with relevant partners and clarified through testing; however, Version 5 was generally agreed upon as a suitable template for the implementation of prototype 1.

3.1 Description of initial interface for Prototype 1

The design for the first prototype can be seen as an intermediate step towards the creation of the final system. However, since some of the necessary inputs to various components are not fully developed, the design is constrained by the current technical capabilities. This interface will be used as the basis for the first implementation; however, it will then be tested and subject to re-design for the final version of prototype 1.

Sample screen shots depicting the various aspects discussed below are provided in Appendix E.

3.1.1 Initial entry page

The entry page to MultiMatch will offer a visitor the option of engaging with the content in two different ways: either by entering a query into the search box or by browsing the content. Hearst (2002) and Beale (2006) advocate making both navigation options (search and browse) available to users at all times, in order to allow for flexibility in navigation tactics. For this reason, both search and browse options will be constantly accessible. As part of the browsing (as discussed in the next section,) a selection of images randomly chosen from the collection will be displayed on the entry page. These will refresh each time the page is reloaded, to give the user a different experience each time they come back to the site. The initial entry page will be simple and visually clean; however, it has been suggested that it provide a brief description of what MultiMatch is and does, in order to help inform people who are unfamiliar with the objectives and functionalities of the site.

3.1.2 Browsing support

In the final MultiMatch system, the browsing functionality will be highly developed and based around the MultiMatch ontology. Users will be able to browse using various categories and even to select a combination of these facets simultaneously (faceted browsing.) Semantic links will also play a role in the navigation of the collection's content.

For the initial prototype, however, a more limited browsing facility will be available to users. This will take two forms:

1. A random overview of the collection

2. Basic navigation across all material via common linkages

The first of these will serve as an entry point to exploring the collection for visitors who have no specific search goal in mind, or who are unfamiliar with the type of material contained within the MultiMatch collection. The display of a random selection of items is a very basic type of collection overview that is present on several currently existing sites.

A number of studies have examined how users search for material in different contexts. Choi & Rasmussen (2003) focused on the domain of history and assessed users' image queries when searching a photographic archive. They found that "most image content that the participants sought fitted into general person and thing, and event and condition limited by a geographic location and time period" (Choi & Rasmussen, 2003: 504). This seemed to correspond with Collins' (1998) finding that 86% of historical photo queries were related to subject terms.

Cunningham et al. (2004) analysed visual arts-related queries to Google Answers, arguing that since these questions were directed to real people rather than to an automated system, they therefore "represent user needs unfiltered by expectations of how a retrieval system expects a query to be presented" (Cunningham et al., 2004: 48). By far, the greatest characteristic of queries (80%) was the presence of some sort of "bibliographic metadata" (i.e. artist, date, nationality, title, etc.) Other frequently occurring questions dealt with the contents of a work of art (i.e. what was depicted) (40%), the genre (17.6%), or the place where a work was seen (15.3%).

Chen (2001) looked at how art history students searched for images to include in a paper on medieval art. The top four categories of queries were related to location (because the paper focused on cities,) objects in the painting, art historical information (i.e. relating to the production context: artists, medium, style), and people or people-related attributes.

Although some of these studies focused more on analysing queries, it has been argued that in the area of humanities, a keyword-based search "is not sufficient because one is above all interested in *relations* e.g. between artists, their works, the friends, their studies, who they inspired, etc." (Benjamins et al., 2004: 433.) Therefore, it will be important to both provide support for the general categories people are likely to employ (i.e. relating to who, what, when, where) and to allow the user to investigate inter-relationships between these elements.

Thus, for MultiMatch, common linkages can be subdivided into:

1. Common elements used by all archives (including creators and dates and automatically created values such as archive location, media type) and
2. Relations between elements.

The following metadata elements are common to all three organisations:

- Creation.Title
- Creation.Related Actor
- Creation.Related Actor-Type (= Creation)
- Creation.Related Actor-Date (creation date)
- Creation.Subject
- Creation.Description

One of the key features of the metadata is that it is not merely a flat data model but attempts to represent explicitly the important relationships in the cultural heritage domain. In the first prototype the focus will be on providing information on those relations which are key to fulfilling the main user requirements, namely:

- Relation Web page - Creators. For example: Web page A describes Creator B;
- Relation Web page - Creation. For example: Web page A reviews Creation B;
- Relation Actor - Creations. For example: Actor A has created Creation B; Actor A has contributed to Creation C; Actor A has copied Creation D.
- Relation Actor - Actors. For example: Actor A is teacher of Actor B.
- Digital Representation - Creation. For example: Creation A is depicted in Digital Representation B.

3.1.3 Query formulation & refinement

In describing the functionalities of the interface with regards to conducting a search, it is useful to approach it from the angle of seeing how the interface supports the user at each stage of the process. Shneiderman (1997) breaks down a user interaction with a search interface into the following stages: formulation (of a query), action, review, and refinement.

The search box is also present on the main entry page, prominently located at the top of the screen. It is here that the user can type in a query and select the languages of query translation. Given that the processes going on “behind the scenes” are fairly complex, it is important to provide feedback to the users about this and to allow them to have control over the outcome. Therefore, two aspects of query entry need to be communicated to the user. The first of these is to provide some indication that the query will be translated, and to allow him or her to select the languages for doing so. Thus, a series of checkboxes under the search box fulfil this function by enabling the selection or de-selection of a particular language. Being able to select the languages desired customizes the experience for the user and can also save time on the part of the system by not performing any unnecessary translations. To save time for repeat visitors, it will also be possible to set the language preferences so that this does not need to be repeated every time.

The second element that must be made explicit at search time concerns the query syntax. Given the nature of automatic translation, users must have the ability to:

- Mark two or more words as a phrase to be translated together (e.g., still life)
- Signal a word or words that should not be translated (e.g. for proper names like Francis Bacon)

This can be achieved by setting up a query syntax using operators such as { } or * *. However, some explanation of the elements of query syntax that are unique to MultiMatch need to be made explicit to users at search time in order to help them obtain the best (and most appropriate) results. This can either be done by providing a brief explanation under the search box, or one that appears when the mouse hovers over the box.

Another possibility considered (and incorporated in early designs) was also to provide a second series of checkboxes allowing the user to restrict the search to specific media types (as proposed by Levin, 2002.) This is present on other competitor sites and was thought to be a potentially time-saving provision for the system (so that it did not need to unnecessarily search all media.) However, in the end, these were removed in order to make the interface cleaner and simpler. Also, it was thought that the default results presentation would serve the same purpose of showing the user an overview of everything available, while simultaneously allowing them to narrow in on one particular media type with a simple click.

With regards to refinement, the ability to refine searches was deemed important in three studies (Brown et al., 2006; Dalmau et al., 2005; Frost et al., 2000,) with users in the latter study suggesting the possibility of refinement both with keywords and visual examples (i.e. “show me more like this.”)

3.1.4 Query expansion

Query expansion is provided by the MultiMatch Query Expansion service. This provides three mechanisms for query expansion that can be used to refine the user’s search experience: blind relevance feedback which expands the query without user intervention, relevance feedback using user selected items as seeds for new terms, and a thesaurus service that provides related terms.

3.1.5 Results presentation

After the user submits a query, the results presentation will consist of two levels: a broad, limited-functionality overview of all media results (referred to in past documentation as the “default” search view,) as well as an enhanced level providing the option to view and interact with a more detailed version of the results for a given media type (the “specialised” search view.)

The overview presentation will initially be displayed following a search. At this point, the various media results will appear in columns so that all media types can be seen on the page. A casual user will be able to perform simple actions at this stage: namely, navigating to text links, enlarging and viewing the images, and playing entire video or audio files. This setup follows Shneiderman’s (1998) recommended principle of interface design: that is, to provide “overview first, zoom and details on demand.”

In terms of results presentation, expert users in Frost et al. (2000) were unlikely to make decisions based solely on visual information, and thus expressed a desire for supplementary details beyond thumbnail images (i.e., a summary containing information such as artist, date, title, etc.) In MultiMatch, should an individual wish to find out more details about a result or interact with it in some other way, he or she can do so by clicking on one of the specific media type tabs and entering the “specialized” results view. At this point, the functionalities available increase to include:

- For texts: Metadata descriptors, textual relevance feedback
- For images: Metadata descriptors, visual or textual relevance feedback
- For video: Metadata descriptors, keyframe view, possibility of playing entire video or of jumping in at a certain point
- For audio: Metadata descriptors, possibility of playing audio file or of jumping in at a certain point

Other aspects that will be available in both the default and the specialised results interfaces are the ability to save items to a folder or workspace, and the ability to view the search history. These features are common to many competitor sites and are useful in that they allow users mark items for easy reference and return to them for further examination. This helps to reduce short-term memory load, which is one of Shneiderman et al.’s (1997) 8 design desiderata (as cited in Hearst et al., 2002.)

Specialised text results interface

Upon navigating to this interface, a user will be able to interact more with the text results and to view more details about a given item. These details will include an enhanced description (where available) and other metadata information such as author, date of publication, and so forth. Clicking on any of these metadata links can function as text based relevance feedback, launching a new query to find more results corresponding to that particular category (e.g., more works by the same author.) This is a fairly common way of enabling further focused browsing and is present on many other sites (for example, the Alinari catalogue.)

Specialised image results interface

Mostafa & Dillon (1996) assert that when looking for images, individuals generally prefer to search for known items using verbal terms such as date of creation or the name of an artist. However, they mention that “find something like this” based on visual features can also be useful when the search goal cannot be precisely put into words (so in this sense, visual similarity search is more like a form of browsing.) The MultiMatch system will support both types of behaviours.

In prototype 1, image relevance feedback will be centred around visual attributes such as texture and colour. This was decided based on the capabilities of GIFT, the image indexing and retrieval system offered by the University of Geneva. This system allows users to use the search results as both positive and negative examples for launching a new query. However, users may also wish to search for similar images based on

textual rather than purely visual criteria. For this reason, basic text-based relevance feedback will also be provided, thus enabling a user to find more images that were created by the same author or in the same year.

Specialised video results interface

The specialised video results page will have the same sort of metadata-based detail presentation and relevance feedback functionality as the other specialised interfaces. However, it will also offer features unique to the video medium, such as the ability of obtaining a quick overview of the video's content. Boelkelheide et al. (2006) list several means of accomplishing this type of video abstraction, which range from the display of keyframes in a storyboard to the use of slideshows or collages. They assert that the use of some sort of abstraction is necessary to support efficient browsing and understanding of video content.

The use of storyboards was selected as the abstraction view for prototype 1 because these are a tool which the video professionals currently use to obtain a quick visual overview of a video's content and to decide if said content is relevant and thus merits further investigation. However, the downside of this format (and others) is that it largely neglects the audio aspect of a video. Therefore, the final system will examine more advanced ways of enabling browsing of the spoken as well as visual content.

With regards to playback, the default option of playing the entire video will still be available. However, it will also be possible to view the place(s) in the video's transcript where the query terms appear, so that the user can commence playback by jumping into the part of the video that is likely to be of greatest interest. All of the video professionals surveyed said they currently did not have access to this method of video navigation but that it would be very helpful for them to be able to do so.

However, the decision of whether or not to display the snippets of the transcript where the word appears to the user needs further thought. As these transcripts will be generated using automatic speech recognition, it is possible that they will contain many errors and could potentially require great effort to decipher, to the point where displaying them to the user could be detrimental. Conversely, though, it is precisely the presence of so many errors that could make it necessary to reveal to the user the context in which their query appears, so they can make a preliminary relevance judgement and accuracy check, to avoid later confusion.

Specialised audio results interface

The audio specialised features will be similar to those of the video, with regards to the metadata and the two playback options.

3.1.6 Multilingual support

Underlying the interaction with the system's various interfaces is the need for multilingual support. Given that one of MultiMatch's distinguishing characteristics is its multilingual functionality, it is important to consider how the system will communicate the language-related options to the user. The first step is to provide the user with information about possible courses of action with regards to translating the query (e.g., to make it clear that query translation can occur and to show the language choices.)

It is possible that people may not be inclined to use any or all of the translation options, either because they are not used to doing so or because they feel more comfortable searching in languages they know (Artiles et al., 2006.) Nonetheless, it is still a potentially useful tool for some groups of individuals. These may include polyglots who would otherwise have to conduct separate searches for the same term in each language spoken, people looking to widen the scope of an image search, or individuals looking for a subject that is likely to be highly represented in another language (i.e. information on an obscure Dutch artist is probably most likely to be available in Dutch.)

In the case where the user opts for query translation and results are obtained in more than one language, there are two ways in which the results can be presented: separated by language or merged together in one list. Each approach has positive aspects.

An argument favouring the approach of separating the language results:

- Translation issues. The same word may be translated differently from language to language and could therefore produce different results. For example, the Italian word “signora” can mean both “woman” and “Mrs.” A machine translation system will translate this word into Spanish as “señora,” which also carries both meanings. However, for the English translation, the system must choose one of the two alternatives, and it chooses “Mrs.” (which is typically the incorrect one.) Therefore, in terms of modifying incorrect query results, it could be easier (and less visually chaotic) to segregate by language.

An argument favouring the merging of language results:

- Convenience. It may be annoying for individuals to have to constantly change between language views. Additionally, in the case of images, which are relatively “language neutral,” there is no strong motivation for separating results by language.

Clearly, an individual’s preference for one means of results presentation over the other will depend on his or her language skills and on the type of media he or she is most interested in interacting with. Further inquiry with users will provide some insight into the best way of handling this issue.

With respect to interacting with the results, further support is needed throughout the document examination and refinement process. Regarding the former aspect, He et al. (2003) mention that people may require document or passage translation support to recognise relevant results and to decide which of these warrant closer examination. At which stage in the prototype this is provided (in the default versus the specialised interface) needs further examination and testing. Again, the means by which this translation support is provided or appreciated depends on the user’s language abilities. Those with some basic knowledge of a given language may be happy to read through the results as they are shown, while those with little or no skill may need to rely on an automatic translation system to help them decipher what is written.

As query translation will be provided by a machine translation program, it is highly likely that errors may occur. The subsequent retrieved results could thus either be confusing and unrelated, or if the translation error is strong enough, it could mean that no results are retrieved at all. Qu et al (2000) classified machine translation errors into the following most commonly-occurring categories:

1. wrong sense disambiguation
2. word-by-word translation of multiword, idiomatic terms
3. broken phrases
4. wrong phrase construction

Keeping this in mind, He et al. (2003) explain that “iterative refinement depends on two types of knowledge: an understanding of why the machine produced the results that were obtained, and an understanding of the ways in which the outcome could be altered.”

Given the types of possible errors listed above, there are two ways in which support can help to prevent these from occurring. The first involves the use of a bilingual dictionary to show all of the possible translations of a word (in the case where more than one exists.) The second involves enabling, through query syntax, the possibility of marking all or part of a query to a) not be translated or b) to be translated as a phrase. These can help to prevent the improper translation of a query from adversely affecting the results. However, should this happen, as described above, it is important to give the user explicit feedback about how their query was translated, along with a means of changing or refining it.

With regards to the point at which this occurs, however, there are two options: it can either be done before or after the query has been submitted. Both alternatives have been tested in the past. Zazo Rodriguez et al. (2005) experimented with allowing users to view (and edit) their translated queries before seeing the search results. They found that usually users reformulated (or refined) the query when the answer to the question they sought was not found in the initial search results. Petrelli et al. (2006) tested both approaches and found a paradox: users said they preferred a system in which the query translations were displayed after the search, but they searched more effectively when the translations could be reviewed and altered before the query was

submitted. In the end, it was decided to show the translations along with the results in order to keep the search task as a single action, but also to enable refinement if deemed necessary.

For Prototype 1, this approach will also be followed. Users surveyed about their preferences indicated that they would probably not look at the translation feedback unless the results were not what they were expecting. Therefore, given this propensity, it should be sufficient to leave the search process uninterrupted and to display the translations as a form of query refinement after the initial search has been launched.

3.1.7 Personalisation and authentication

MultiMatch is planned as a web search engine to be used without any log-in requirement. In this way, users can directly connect to MultiMatch and perform their queries as they do with general search engines.

However, in order to provide the user with the possibility of customising the MM graphical interface and filtering out unwanted information, MultiMatch will implement an authentication facility which – via the storage of user-defined data in an online database – will allow the account holder to specify a set of search parameters, such as preferred languages, preferred information sources and preferred file formats. Logging in to MultiMatch will also activate some search specific functionalities such as:

- Search history. This stores previous user searches. It is still under discussion if the system will automatically store queries performed by the users (allowing them to make a post filtering to discard those not relevant) or whether users will manually perform this activity at search time saving only those queries really relevant for them.
- Basket facility. This allows the user to save pointers to relevant images and cultural items retrieved on previous searches for future references.

Not logging in to MultiMatch does not imply a restricted access to its search services (the user will be able to perform all the types of searches described above) but will have a reduced possibility of customising search results.

Proprietary indexed contents will be provided to the users as low quality files (i.e. a low resolution image in the case of images, a short and low resolution video in the case of videos and a fragment of the whole text in the case of texts) with the corresponding URL to the vendor site, allowing them to buy or to access the full contents using the specific selling or accessing policies of each content provider.

A basic user account and log-in facility will be implemented in the MultiMatch first prototype, mainly oriented towards defining the user's language preferences and relating to his/her search history. The second prototype will implement the whole set of features described in the preceding paragraphs.

3.2 Next steps

As discussed in Section 2.2, the design process is currently midway through the first lifecycle for prototype 1, and at the “interactive prototype” stage shown in Figure 1 of Section 2.1. At this point, implementation of the first interactive prototype has begun and will proceed until the first system is ready to be presented for review. While implementation work is occurring, further design work and refinement will be simultaneously carried out. For example, a heuristic evaluation of the interface can be conducted to identify any potential usability problems.

Although it will be difficult to begin any sort of large scale user evaluation until the interactive prototype is completely functional (e.g. in late July or early August,) smaller scale studies and user testing will be carried out in parallel with implementation in order to test various design alternatives. These are discussed further in Section 4.1. However, an informal evaluation of the designs was carried out in order to gain feedback from potential users about certain aspects of the interface presentation and to identify if any major changes needed to be made, pre-implementation. A summary of results follows.

3.2.1 User comments on initial prototype 1 designs

After creating the mockups for the initial prototype 1 design, it was considered useful to show them to a selection of users in order to identify any aspects that were confusing, unclear, or in need of alteration. In

total, 9 users were shown the various aspects of the system (as illustrated in Appendix E) and asked to comment. Input was gathered on elements relating to multilingual support and the overall layout. Various implications gathered from the participants' comments include:

- Language selection: The area under the search box should be explained more (some people were unsure about what ticking the boxes would do: would it translate the results, or just the query?)
- Query syntax: Should be provided somewhere near the search box, but people did not want to see it as text cluttering the page. It was suggested that this information be provided via a link (or popup, or mouseover) located near the search box.
- With regards to the task about Don Quixote, all participants said they were likely to search either just in English, or in English and Spanish. Therefore, based on this, it may not be worthwhile to set "all" as the default option
- This being said, participants did acknowledge that the languages selected "depends on what you want out of it." That is, if the main goal is to find images, then language is not crucial. However, one individual said he would conduct a text search using English and Spanish, then launch a new search for images, this time set at the "All" language setting.
- Separating or merging of language results: Overall sentiment favoured the separating. However, one person mentioned he would like to have the results merged (this person had strong foreign language abilities, and thought it would be "tiresome" to keep switching from one language to another.) One way of catering to people like him could be to add an "All" languages tab in addition to the separate language tabs.
- Automatic translation: Was viewed as "better than nothing," but not extremely useful for serious research (although "OK for holidays.") Automatic translation of snippets was suggested to be presented at the overview stage, although it would then presumably need to be offered in the specialised view as well.
- Query translation feedback box: Most participants agreed that it was best to show this after rather than before the query was executed. Feelings regarding the usefulness of this were mixed. Three individuals said they would not look at how the query was translated unless the results were irrelevant or "complete rubbish." One would not look at the feedback because she said she couldn't understand the other foreign languages and thus saw no point to it. However, one individual said that he would look at the translation no matter what, to see if he could improve his results. It seems logical, for the moment, to leave the presentation of this box as is, where it can be easily ignored by those it does not interest.
- Overall structure: Make it clearer what lays behind the "specialised" results tabs (it wasn't obvious that clicking on Texts, Images, Audio, or Video would provide enhanced ways of interacting with the material.)
- Specialised image view: It is important to show image provenance (where does the image come from, what is the URL of the page?)

These suggestions (as well as others gathered in further studies) will help guide and influence the next iteration of design.

4. Implementation and architecture

This section describes the infrastructure adopted to implement the MultiMatch interface. This is significant as the architecture defines how the interface will be developed and deployed. It is also significant because a component-based approach has been adopted that will allow transition between prototypes through a framework of reusable interface objects. The code associated with the components described in this deliverable has been stored on the MultiMatch integration server hosted by OCLC PICA.

4.1 AJAX coding technology

The MultiMatch interface will adopt an AJAX programming approach to build an interactive and usable web interface able to connect asynchronously to the web services provided by the MultiMatch core.

AJAX, which is shorthand for "Asynchronous JavaScript And XML," is a web development technique for creating interactive web applications. The intent is to make web pages feel more responsive by exchanging small amounts of data with the server behind the scenes, so that the entire web page does not have to be reloaded each time the user requests a change. This is intended to increase the web page's interactivity, speed, and usability.

The primary justification for AJAX style programming is to overcome the page loading requirements of HTML/HTTP-mediated web pages. AJAX creates the necessary initial conditions for the evolution of complex, intuitive, dynamic, data-centric user interfaces in web pages—the realization of that goal is still a work in progress.

Web pages, unlike native applications, are loosely coupled, meaning that the data they display are not tightly bound to data sources and must be first *marshalled* (set out in proper order) into an HTML page format before they can be presented to a user agent on the client machine. For this reason, web pages have to be reloaded each time a user needs to view different datasets. By using the **XmlHttpRequest** object to request and return data without a re-load, a programmer by-passes this requirement and makes the loosely coupled web page behave much like a tightly coupled application, but with a more variable lag time for the data to pass through a longer "wire" to the remote web browser.

For example, in a classic desktop application, a programmer has the choice of populating a tree view control with all the data needed when the form initially loads, or with just the top-most level of data—which would load more quickly, especially when the dataset is very large. In the second case, the application would fetch additional data into the tree control depending on which item the user selects. This functionality is difficult to achieve in a web page without AJAX. To update the tree based on a user's selection would require the entire page to re-load, leading to a very jerky, non-intuitive feel for the web user who is browsing the data in the tree.

The AJAX technique uses a combination of:

- **XHTML** (or **HTML**) and **CSS**, for marking up and styling information.
- The **DOM** accessed with a client-side scripting language, especially **ECMAScript** implementations such as **JavaScript** and **JScript**, to dynamically display and interact with the information presented.
- The **XMLHttpRequest** object is used to exchange data asynchronously with the web server. In some AJAX frameworks and in certain situations, an **IFrame** object is used instead of the **XMLHttpRequest** object to exchange data with the web server, and in other implementations, dynamically added `<script>` tags may be used.
- **XML** is sometimes used as the format for transferring data between the server and client, although any format will work, including preformatted **HTML**, plain text, **JSON** and even **EBML**. These files may be created dynamically by some form of server-side scripting.

Like **DHTML**, **LAMP** and **SPA**, **AJAX** is not a technology in itself, but a term that refers to the use of a group of technologies.

AJAX advantages and disadvantages

The main advantages of using **AJAX** are the following:

- **User interface.** The most obvious reason for using **AJAX** is an improvement to the user experience. Pages using **AJAX** behave more like a standalone application than a typical web page. Clicking on links that cause the entire page to refresh feels like a "heavy" operation. With **AJAX**, the page often can be updated dynamically, allowing a faster response to the user's interaction. While the full potential of **AJAX** has yet to be determined, some believe it will prove to be an important technology, helping make the Web even more interactive and popular than it currently is.

- **Bandwidth usage.** By generating the HTML locally within the browser, and only bringing down JavaScript calls and the actual data, AJAX web pages can appear to load relatively quickly since the payload coming down is much smaller in size. An example of this technique is a large result set where multiple pages of data exist. With AJAX, the HTML of the page, e.g., a table structure with related TD and TR tags can be produced locally in the browser and not brought down with the first page of the document. In addition to "load on demand" of contents, some web based applications load stubs of event handlers and then load the functions on the fly. This technique significantly cuts down the bandwidth consumption for web applications that have complex logic and functionality.
- **Separation of data, format, style, and function.** A less specific benefit of the AJAX approach is that it tends to encourage programmers to clearly separate the methods and formats used for the different aspects of information delivery via the web. Although AJAX can appear to be a jumble of languages and techniques, and programmers are free to adopt and adapt whatever works for them, they are generally propelled by the development motive itself to adopt separation between the following:
 1. Adopt separation between the *raw data or content to be delivered* - which is normally embedded in XML and sometimes derived from a server-side database.
 2. Adopt separation between the *format or structure of the webpage* - which is almost always built in HTML (or better, XHTML) and is then reflected and made available to dynamic manipulation in the DOM.
 3. Adopt separation between the *style elements of the webpage*: everything from fonts to picture placement are derived by reference to embedded or referenced CSS.
 4. Adopt separation between the *functionality of the web page* which is provided by a combination of:
 - JavaScript on the client browser (also called DHTML),
 - Standard HTTP and XMLHttpRequest for client-to-server communication, and
 - Server-side scripting and/or programs using any suitable language preferred by the programmer to receive the client's specific requests and respond appropriately.

On the other side, main disadvantages of using an AJAX approach are the following:

- **Browser integration.** The dynamically created page does not register itself with the browser history engine, so triggering the "Back" function of the users' browser might not bring the desired result. Developers have implemented various solutions to this problem. These solutions can involve using invisible *iframes* to invoke changes that populate the history used by a browser's back button. Google Maps, for example, performs searches in an invisible IFRAME and then pulls results back into an element on the visible web page. The World Wide Web Consortium (W3C) did not include an *iframe* element in its XHTML 1.1 Recommendation; the Consortium recommends the *object* element instead. Another issue is that dynamic web page updates make it difficult for a user to bookmark a particular state of the application. Solutions to this problem exist, many of which use the URL fragment identifier (the portion of a URL after the '#') to keep track of, and allow users to return to, the application in a given state. This is possible because many browsers allow JavaScript to update the fragment identifier of the URL dynamically, so that AJAX applications can maintain it as the user changes the application's state. This solution also improves back-button support. It is, however, not a complete solution.
- **Response-time concerns.** Network latency — or the interval between user request and server response — needs to be considered carefully during AJAX development. Without clear feedback to the user, smart preloading of data and proper handling of the XMLHttpRequest object, users might experience delay in the interface of the web application, something which they might not expect or understand. Additionally, when an entire page is rendered there is a brief moment of re-adjustment for the eye when the content changes. The lack of this re-adjustment with smaller portions of the screen changing makes the latency more apparent. The use of visual feedback (such as throbbers) to

alert the user of background activity and/or preloading of content and data are often suggested solutions to these latency issues.

- **Search engine optimization.** Websites that use AJAX to load data which should be indexed by search engines must be careful to provide equivalent data at a public, linked URL and in a format that the search engine can read, as search engines do not generally execute the JavaScript code required for AJAX functionality. This problem is not specific to AJAX, as the same issue occurs with sites that provide dynamic data as a full-page refresh in response to, say, a form submit (the general problem is sometimes called **the hidden web**).
- **Reliance on JavaScript.** AJAX relies on JavaScript, which may be implemented differently by different browsers or versions of a particular browser. Because of this, sites that use JavaScript may need to be tested in multiple browsers to check for compatibility issues. It's not uncommon to see a JavaScript code written twice, one part for IE, another part for Mozilla compatibles. (see also Cross-platform web design). The level of IDE support for JavaScript is exceptionally poor. An issue also arises if the user has switched off JavaScript support in the browser, thus disabling the functionality of the page. Implementing **Mutual Exclusion** in JavaScript may solve the problem.

AJAX libraries

Programming using an AJAX approach implies the use of specific JavaScript frameworks which facilitate not only the layout of web components within the front-end client application, but also the asynchronous connections with the web services to provide the data. Many of such frameworks exist; however, for MultiMatch we have considered the following:

- **Yahoo! User interface library⁶.** The Yahoo! User Interface (YUI) Library is a set of utilities and controls, written in JavaScript, for building richly interactive web applications using techniques such as DOM scripting, DHTML and AJAX. The YUI Library also includes several core CSS resources. All components in the YUI Library have been released as open source under a BSD license and are free for all uses.
- **Google Web Toolkit⁷.** The Google Web Toolkit (GWT) is an open source Java software development framework that makes writing AJAX applications like Google Maps and Gmail easy for developers who don't speak browser quirks as a second language. Writing dynamic web applications today is a tedious and error-prone process; you spend 90% of your time working around subtle incompatibilities between web browsers and platforms, and JavaScript's lack of modularity makes sharing, testing, and reusing AJAX components difficult and fragile. The GWT lets you avoid many of these problems while offering your users the same dynamic, standards-compliant experience. You write your front end in the Java programming language, and the GWT compiler converts your Java classes to browser-compliant JavaScript and HTML.
- **Script.aculo.us library⁸.** script.aculo.us provides easy-to-use, cross-browser user interface JavaScript libraries to make web sites and web applications. Its libraries provide an animation framework, drag and drop, AJAX controls, DOM utilities, and unit testing.

4.2 MultiMatch user interface technology

After deciding to use an AJAX approach for the MultiMatch user interface implementation, the next step has been to decide about the best combination of technologies to use for rapid development. For MultiMatch, we have chosen the following technologies for each part involved in the user interface implementation:

⁶ <http://developer.yahoo.com/yui/>

⁷ <http://code.google.com/webtoolkit/>

⁸ <http://script.aculo.us/>

- **For the server part of the interface**, which will be responsible for making the connections to the MultiMatch web services server and also for implementing a part of the user interface logic, we will use JAVA as coding language. This language is the same one used to implement the server part of the project.
- **For the client part of the interface**, which will be responsible for the layout of contents, tracking of user's interactions and performance of some user interface logic in the client, we will use the Google Web Toolkit.

Reasons for choosing GWT instead of other possibilities are the following:

1. **Dynamic Reusable UI Components.** GWT user interface classes are similar to those in existing UI frameworks such as Swing and SWT, except that the widgets are rendered using dynamically-created HTML rather than pixel-oriented graphics. While it is possible to manipulate the browser's DOM directly using the DOM interface, it is far easier to use classes from the GWT Widget hierarchy. Using widgets makes it much easier to quickly build interfaces that will work correctly on all browsers.

2. **Easy to Implement Remote Procedure Calls (RPC).** A fundamental difference between GWT applications and traditional HTML web applications is that GWT applications do not need to fetch new HTML pages while they execute. Because GWT-enhanced pages actually run more like applications within the browser, there is no need to request new HTML from the server to make user interface updates. However, like all client/server applications, GWT applications usually *do* need to fetch data from the server as they execute. The mechanism for interacting with a server across a network is called making a remote procedure call (RPC), also sometimes referred to as a *server call*. GWT RPC makes it easy for the client and server to pass Java objects back and forth over HTTP.

When used properly, RPCs give you the opportunity to move all of your UI logic to the client, resulting in greatly improved performance, reduced bandwidth, reduced web server load, and a pleasantly fluid user experience. The server-side code that is invoked from the client is often referred to as a *service*, so the act of making a remote procedure call is sometimes referred to as invoking a service. To be clear, though, the term *service* in this context isn't the same as the more general "web service" concept. In particular, GWT services are not related to the Simple Object Access Protocol (SOAP).

3. **Browser History Management.** As AJAX applications don't keep track of states (although it's explicitly programmed) it's not possible to use the browser's back button. GWT makes AJAX sites more usable by easily adding state to the browser's back button history via specific built in classes.
4. **Real Debugging.** In production, GWT code is compiled to JavaScript, but at development time it runs in the Java virtual machine. That means when the code performs an action like handling a mouse event, it is possible to get full-featured Java debugging, with exceptions and the advanced debugging features of IDEs like Eclipse.
5. **Browser Compatibility.** GWT applications automatically support IE, Firefox, Mozilla, Safari, and Opera with no browser detection or special-casing within the code in most cases.
6. **JUnit Integration.** GWT's direct integration with JUnit allows unit testing both in a debugger and in a browser.
7. **Internationalization.** With GWT it's very easy to create efficient internationalized applications and libraries. It includes a flexible set of tools to help to internationalize developed applications and libraries. It also provides a variety of techniques to internationalize strings, typed values, and classes.



8. **Interoperability and Fine-Grained Control.** The GWT compiler translates Java source into JavaScript. Sometimes it's very useful to mix handwritten JavaScript into your Java source code. For example, the lowest-level functionalities of certain core GWT classes are handwritten in JavaScript. GWT borrows from the Java Native Interface (JNI) concept to implement JavaScript Native Interface (JSNI). Writing JSNI methods is a powerful technique, but should be used sparingly. JSNI code is less portable across browsers, more likely to leak memory, less amenable to Java tools, and harder for the compiler to optimize.
9. **Completely Open Source.** All of the code for GWT is available under the Apache 2.0 license.

Most of the presented GWT features cover the gaps presented in section 6.1.1 as AJAX disadvantages.

4.3 MultiMatch User Interface Architecture

As outlined in section 3.2, the Google Web Toolkit (GWT) has been selected as the main web interface implementation tool, thus all client-side applications – along with their corresponding server-side support functionality – are written using this framework.

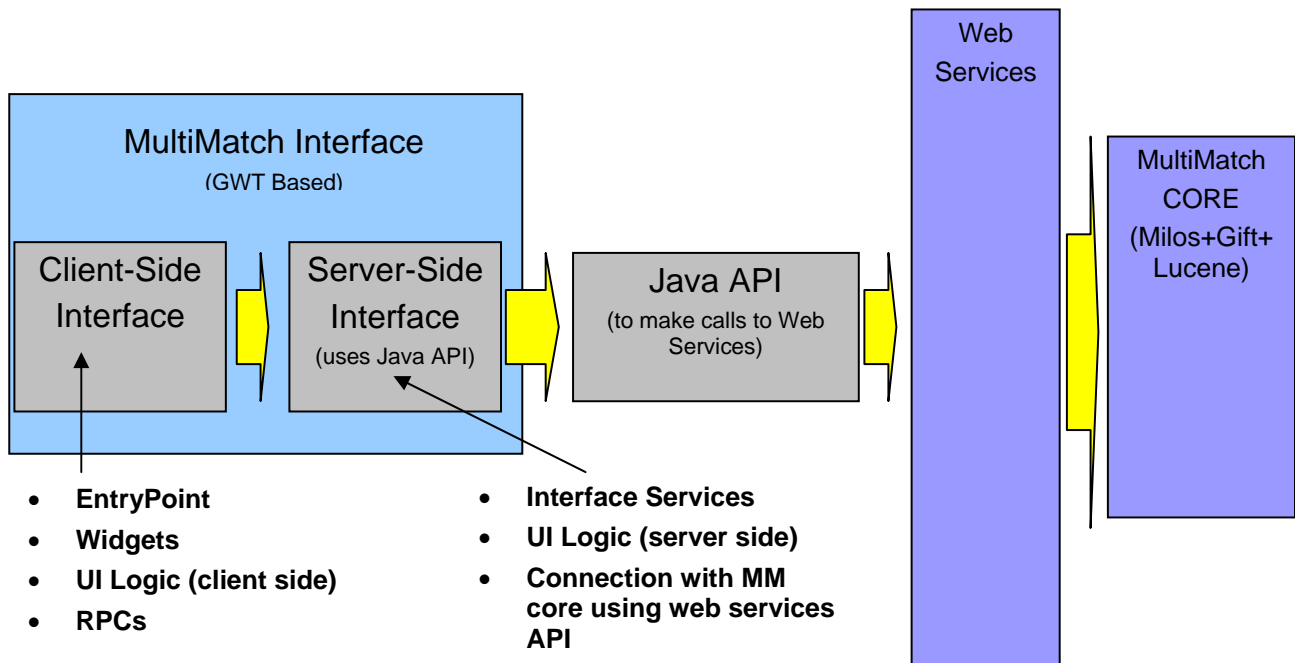


Fig. 3 Main components of the user interface sub-system.

The main user interface architecture can be outlined as follows (Figure 3):

- **Wrapper to call MultiMatch web services.** This component will wrap the MultiMatch web services providing a set of Java classes to be used to call core services. The final goal of this wrapper is to become a standalone API (Application Programming Interface) to be used by the UI as a black box to make calls to the MultiMatch core and to be provided to third party programmers to create new MultiMatch-based applications as well. The benefits of this approach are the following:
 1. Web services calls are encapsulated in a set of Java classes that are easier to call from the UI and completely untied to the UI logic.
 2. Call parameters and return values are Java objects that will be conveniently translated by the API to perform the calls to the web services using their own communication protocols (RESTFUL requests with XML responses).
 3. To have an standalone API to be used by third-party users to build their own MultiMatch based cultural heritage applications.
- **User Interface (server-side).** This component provides a set of services to be called from the client UI part. These services are supposed to enclose complex functionalities (logic) by combining parallel or sequential calls to the API core web services. Also, due to GWT technology, they have to be implemented as asynchronous services to provide GWT compliant Java objects.
- **User Interface (client-side).** This component is responsible for rendering the front end HTML and Javascript web page that will be shown to the client. As this web page is automatically built by GWT in the final compilation, Java classes will be needed to define the set of GWT widgets, components and events needed to lay out the contents and to define interactions between them. Also, RPCs will be triggered from this point in order to perform the interface logic needed to achieve any of the defined UI functionalities.

4.3.1 User Interface Models

The UI server-side functionality is encapsulated in several modules (referred to as models in D3.2), which are detailed below (see Figure 4). All the technical details regarding these models are covered in D3.2.

- **User Interface Search Model.** In the context of MultiMatch web services; a *search* is defined as the specification of parameters to retrieve some desired data set. For instance, a user may want to retrieve the first 50 text documents and videos related to ‘Van Gogh’ written or spoken in English. This means that a search does not only imply the specification of a query (see Query Model explained later), but also incorporates a set of parameters or attributes that may vary depending on the data type to retrieve (text, audio, video, etc.).
- **User Interface Query Model.** While the MultiMatch Search Model’s methods are concerned with setting the search requests to be sent to the MultiMatch core and also to retrieve and to model the search results, the query module is more focused on which types of queries can be sent to the MultiMatch core search service. As MultiMatch is going to store and retrieve information in different media (text, video, images, audio) which will also be enriched with semantic (metadata) information, there is a wide range of possibilities vis-à-vis query operations on the MultiMatch data repositories. We have distinguished four potential types of queries that can be triggered from the user interface: free text queries, metadata queries, metadata-value queries and multimedia queries. Also, simple Boolean operators have also been considered to allow for the combination of query components.
- **User Authentication and Customisation Model.** Essentially, this service is a database facility which stores information allowing the MultiMatch system to “recognise” a user and configure its web interface accordingly. Additionally, this service allows users – via the ‘workspace’ facility – to store information designating objects of interest returned from previous searches. It is to be noted that the workspace does not actually store the objects themselves, only URNs which specify where that object is to be found⁹ would indicate the location of a photograph formatted as a .jpg file named “nice_photo”. When the registered user logs in to the MultiMatch web site and commences a new session, the Authentication/Customisation Model invokes a search method which will return the object and display it within the workspace. Finally, a Search History facility will be also be tied to the user. This feature allows the user to store queries and clicked search results for future reference.
- **User Interface Translation Model.** This service is, fundamentally, a translation engine which allows a user to express his/her request in any (or all) of the four officially supported MultiMatch languages. The Translation Model’s methods are not defined in this document since this service is no more than an interface that builds directly on the services exposed by the MultiMatch translation service.

⁹ For example a URI string such as http://somehost/absolute/URI/with/absolute/somepath/nice_photo.jpg

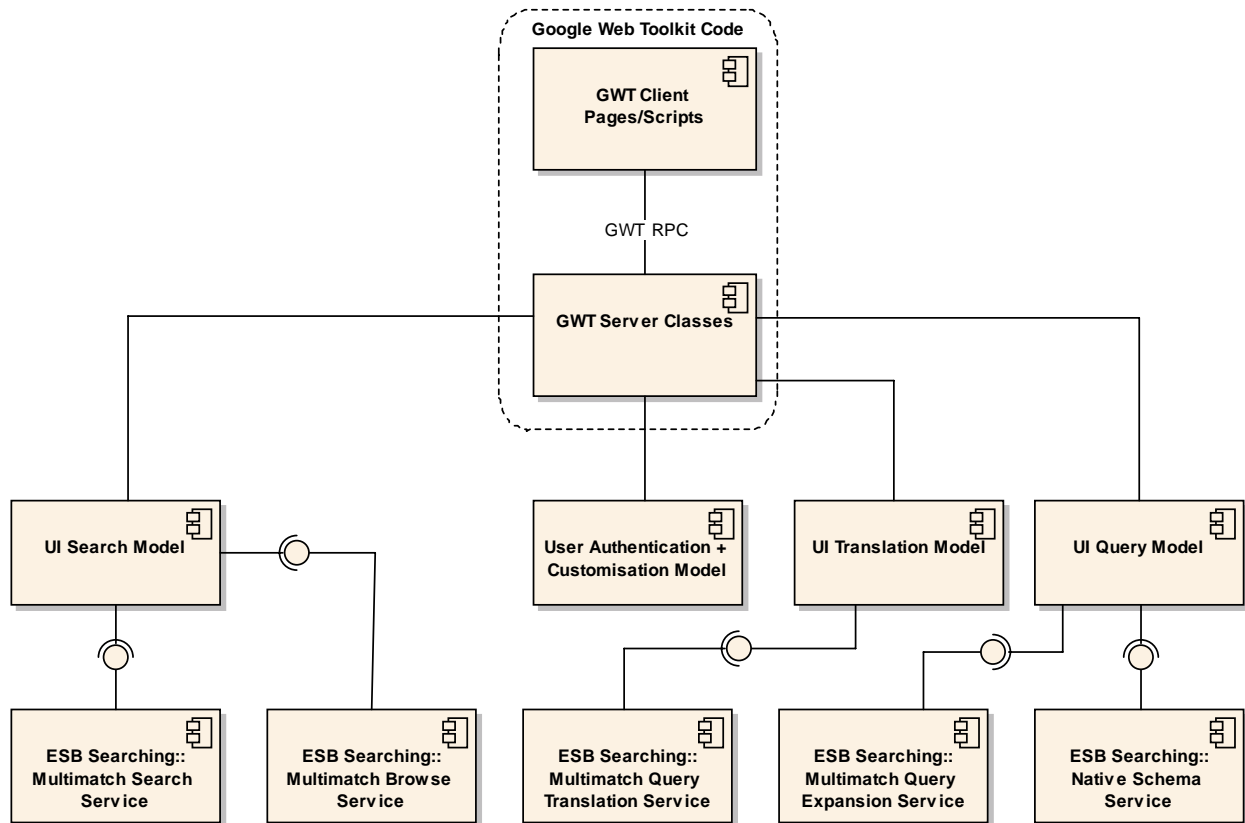


Fig. 4 User interface components (see Deliverable 3.2)

4.3.2 User Interface Widgets and Components

To achieve UI functionalities and also the MultiMatch First prototype goals, the client interface must provide the user with a set of custom widgets and components which laid out on a web page will be able to capture his/her intentions and transform them into web service calls to the MultiMatch core. This set of components has been written down in a tabular format in order to clarify the functionality of each individual component. Tables are shown in Appendix F.

4.3.3 Video and audio clip playback

The MultiMatch audiovisual data collection contains several documentary-length audio and video clips, each of which may contain multiple cultural news items, interviews, and other presentations. It will often be the case, therefore, that only a specific segment of a given video clip will be relevant in relation to any specific query. In order to return only the audiovisual information of interest, a streaming video server has been installed on the MultiMatch integration server. The chosen¹⁰ method for serving audiovisual material is using the embedded Real Networks' RealOne Player. This allows controlled playback of audiovisual material when served from a streaming server, for which we are using RealMedia Helix technology. The full version has been purchased for use in MultiMatch and allows up to 250 simultaneous requests and a suite of conversion tools (e.g. transforming videos in *mpeg* format into *ram/rm*).

In a typical scenario of usage, a segment of a video, from time t to time $t + n$, may fulfil the query parameters of a search. These video clip temporal segmentation markers, commonly referred to as

¹⁰ Methods to server audiovisual material include embedded RealPlayer, Adobe Flash and stand-alone applications such as JFerret (<http://www.idiap.ch/mmm/tools/jferret>). We consulted with Dr. Mike Flynn from IDIAP (mike.flynn@idiap.ch), creator of the JFerret tool.



timestamps, are passed as arguments (along with the name and URL location of the video clip itself) to the MultiMatch specialised video playback interface which will, in turn, send the timestamp information to the Helix streaming server. The server then streams only the relevant section of the video (or audio) back to the end-user's audio-video player which is embedded in the relevant MM web page.

Apart from streaming and playback of the audio-video segment of interest, the MM specialist multimedia browsing facilities are also capable of displaying keyframe and/or key word sequences associated with the returned video segments. These keyframe / key word sequences are also displayed so that the user may then initiate playback of the returned segment at a precise point corresponding to the selected keyframe or key word location.

4.4 MultiMatch development environment

The UI implementation is taking place within the development environment provided by OCLC-PICA, which has been previously tested by the UI team. Now, WP6 partners are working with the MultiMatch tools provided to store, track, compile, integrate and deploy code within the whole implementation project developed by all the MultiMatch coding teams.

To begin with, WP6 partners have been involved in the creation of two demonstration systems to test the feasibility of these tools working with GWT (which had not been previously tested). Results of these tests were successful and the systems were well compiled and deployed. One of them is now running on the MultiMatch server and can be tested at:

<http://bender.lsi.uned.es:8080/SearchServiceViewer/SearchServiceViewer.html>

Secondly, once it was demonstrated that tools were well fitted to the UI requirements, the main UI architecture skeleton was uploaded into the MultiMatch project repository to begin the first prototype implementation. It can be accessed at (this site is password protected and not publicly accessible – contact the project coordinators for access to this site):

http://homer.multimatch.hostedbyfdi.net/svn/repos/trunk/multimatch_gui/

The architecture described earlier is represented in the repository (below the **src/eu/multimatch** folder) by the following folder structure:

1. **Core Folder.** It holds all the classes used to perform calls to the MultiMatch core services. Now it is included within the interface project but in the future will conform to a standalone API that will be included as an external library within the project. The folder is divided into different subfolders which contain each of the models (search, translation, authentication, user) needed to perform the connections to the services.
2. **UI folder.** It contains all the classes used to implement the layout and the services for the GUI. It is divided as follows:
 1. **Server folder.** It will contain the implementation of all the server services classes. These classes must model the corresponding services making the calls to the API and then transforming the returned objects into Java-GWT objects that implement the “IsSerializable” interface. This last step is critical, as the use of GWT only allows those objects implementing this feature to travel from the server to the client. A service class implementation must extend *RemoteServiceServlet* and implement the service class, this last defined in the client folder (see below).
 2. **Client folder.** It will contain the following items:

1. The main entry point to the GWT application. It is a Java class implementing the *EntryPoint* GWT interface. This class is going to be responsible for the layout of components and the communication between them (using a Mediator or an Observable/Observer Design Pattern).
2. The client services implementation. These classes are a complement to the classes implemented in the server folder. They are responsible for making the asynchronous calls to the services and retrieving the results when ready. For each service class in the server folder it is necessary to define two classes in the client folder. First of all, the class for the service which extends the GWT *RemoteService* class must be defined (this class includes everything needed to make the asynchronous call and to retrieve the response) and, secondly, an interface with the methods implemented in the service class (within the server folder) must be defined. In a future version committed to SVN, these classes will be moved to a new *services* folder within the client folder.
3. A **widgets folder** with all the widgets implementations. To define a widget using GWT, it must implement the *Composite* interface. Also, to track events it can implement any of the *Listener* classes defined for GWT.

5. Discussion and summary

This section will discuss the interface design thus far and attempt to show how current work relates to the tasks and work plan described in the DoW. We will also be looking ahead here to give an idea of the functionality which we feel will be available in prototype 2 (i.e. the final prototype), and how this might look on the interface. This is likely to reflect the common vision.

5.1 Planned experiments and user testing

As mentioned previously, various studies will be carried out in parallel with the implementation and will provide further input into the initial prototype design. Since these will precede a working interactive prototype, 2 courses of action exist: to find existing sites with the desired functionalities and use these as examples for testing, or to wait for the relevant MultiMatch services to come online.

While formal user testing will be carried out to evaluate the interactive prototype as a whole, there are some specific areas that could be the subject of particular focus for smaller scale experiments. It has been stated before that the degree and type of multilingual support desired will likely depend on a user's language skills. Therefore, a study to determine the nature of the relationship between language ability and desired functionality would help to design an experience that can be personalised with regards to the multilingual facilities (e.g. translation of queries and examination of results.)

Further investigation of the video and audio visualisation and playback would also be useful. While individuals responded positively to the idea of jumping to particular words being spoken in a video, it is uncertain whether the high level of potential errors inherent in such a system may ultimately do more harm than good, or if the presence of the speech transcript can do anything to help mitigate this. These and other experiments may be carried out both prior to and following the completion of the first prototype.

The user test will be conducted once the final prototype is finished. A controlled set of tasks will be set up in order to evaluate the system. Users will be asked to complete these tasks (based on the processes described in the scenarios.) By observing users during this process and obtaining their comments, it will be possible to identify those aspects of the system which are satisfactory and those which are unclear, confusing, or in need of changing.

5.2 Functionality of final prototype

As stated before, the first prototype is an intermediate step in the creation of the final system and, as such, will not be able to incorporate the full range of functionality to be showcased in the final product. As previously stated in the common vision document, “the key ideas of the proposed search engine are multiplicity and aggregation.” The final system will provide enhanced means of navigating the aggregated content in the ways described below. These are drawn from the ideas put forth in the common vision and WP1 deliverables. They have also been supported and affirmed as useful by some of the users interviewed. Those features which are foreseen for the final system and are currently under development are outlined below, accompanied by illustrative screenshots taken from currently existing sites.

- The creation of a common “MultiMatch” ontology to allow navigation by semantic relationships. This will also incorporate “web mining” as discussed in the common vision. The resulting functionality will be similar to what is shown here:

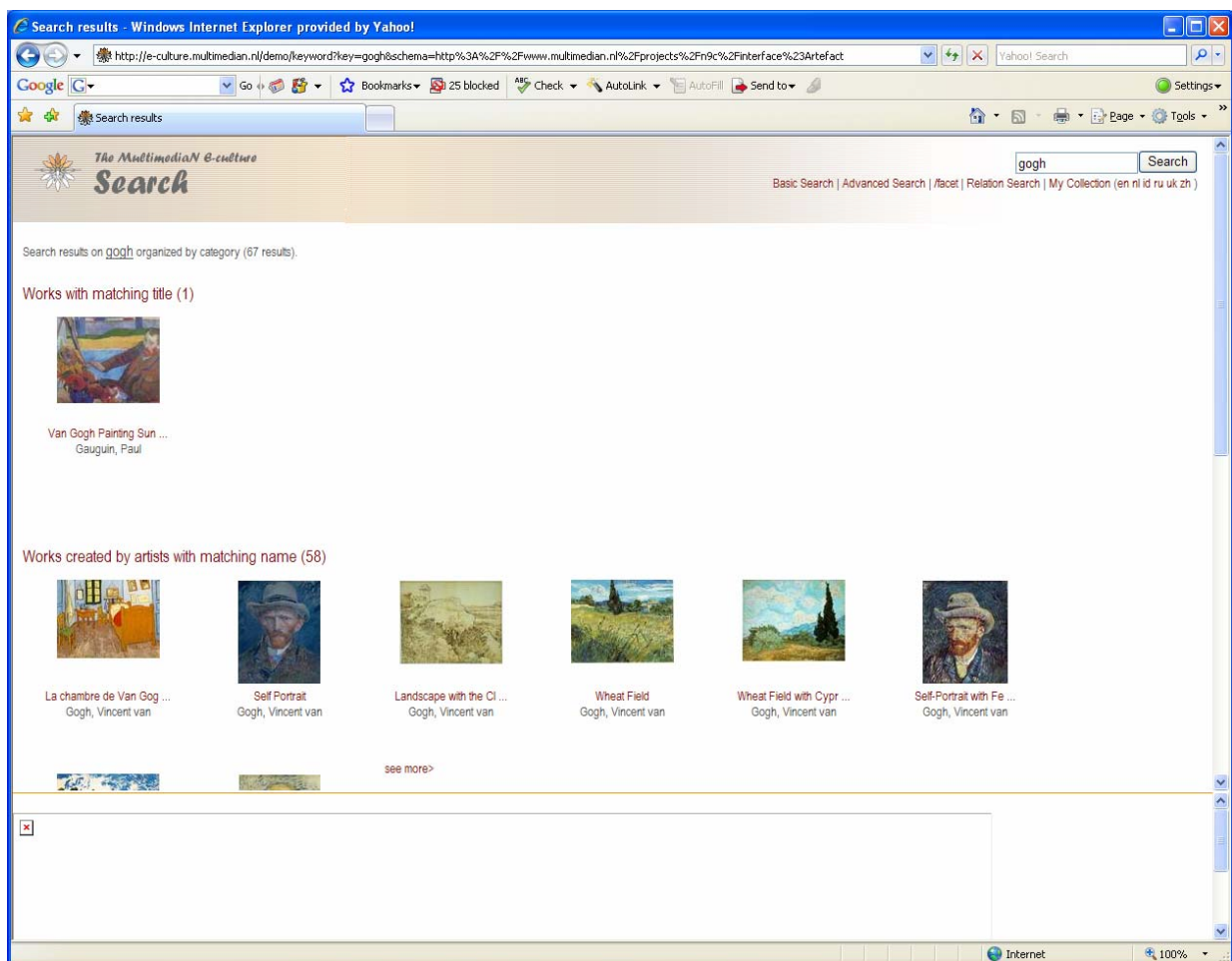


Fig. 5. Taken from demo found at <http://e-culture.multimedial.nl> Other results categories include “works created by a cousin of artists with matching name” and “works created by artists who collaborated with artists with matching name.”

- More advanced browsing capabilities (faceted browsing): As literature and the user interviews indicate, many of the typical information needs in the cultural heritage domain involve a combination of two or more categories (i.e. name, date, and location.) A faceted browsing system would help to support this type of need.

One approach being considered to provide this functionality would be to use the mSpace browser technology (see below for an example.) This enables the navigation of information over a range of categories and is easily personalisable by the user. mSpace software is open source and free to use, so it could be an excellent source to use for the browsing of MultiMatch data.

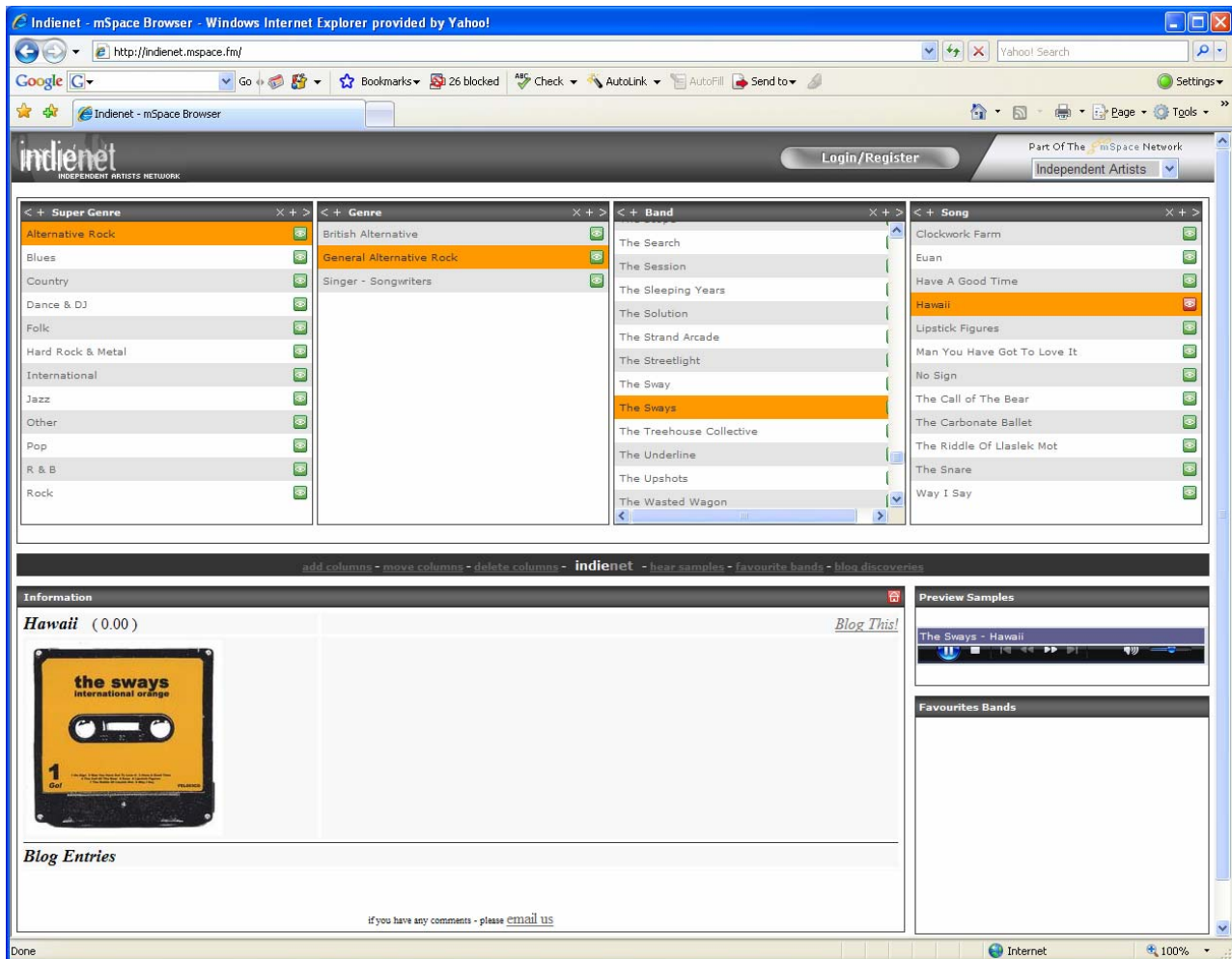


Fig. 6. mSpace demo from <http://indienet.mspace.fm>

- Enhanced and expanded collection overviews
- The following are sample screenshots demonstrating collection overviews that UniGE has explored in the past. These may be used as a starting point for enhancing user understanding of the collection's content.

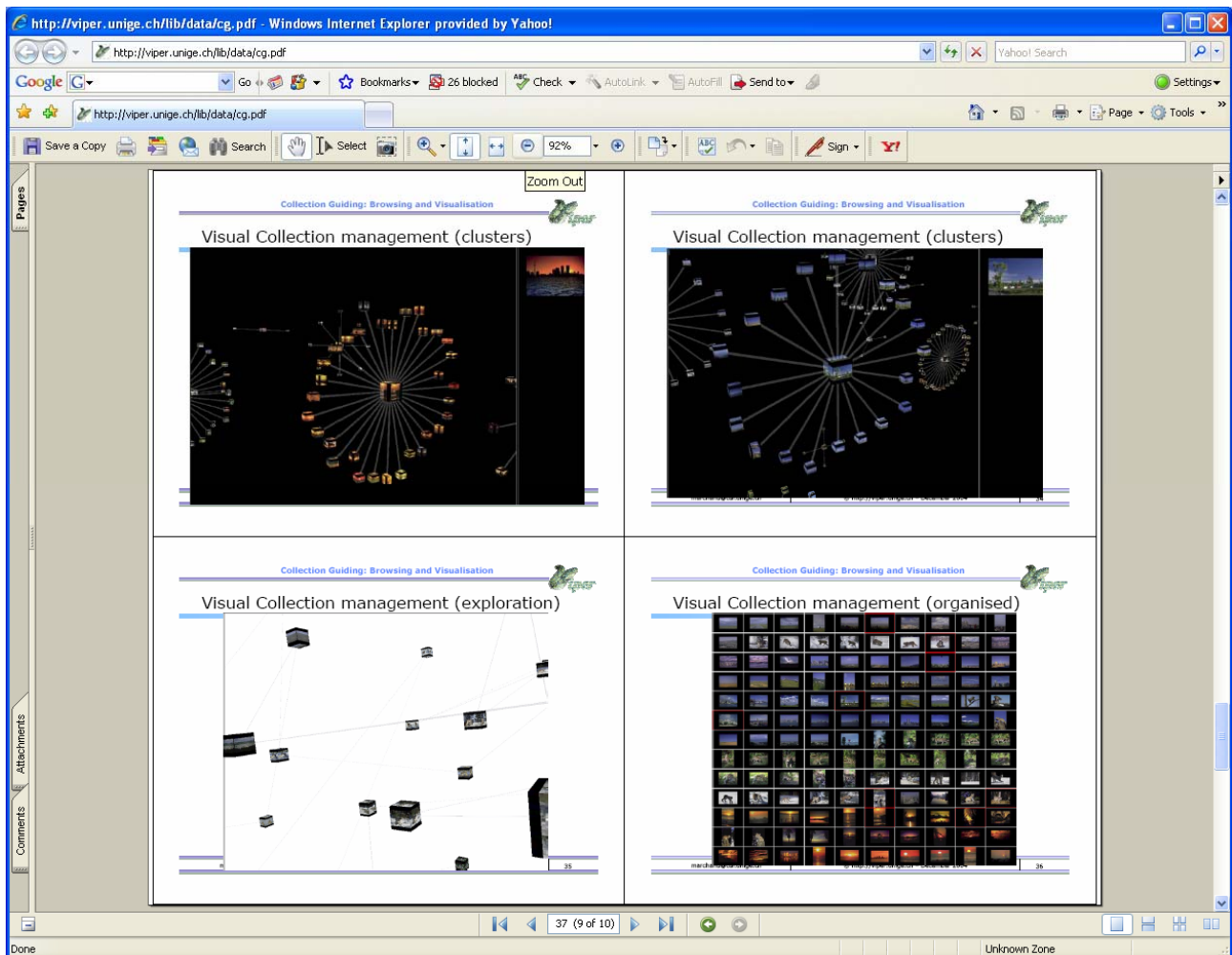


Fig. 7. Collection overviews from <http://viper.unige.ch/lib/data/cg.pdf>

- The creation of cultural objects (i.e. aggregated information units about works, creators, etc.) A cultural object can be defined as an information unit which is built using the information in texts, images and audio/video documents indexed. This piece of information can be displayed in different ways, such as:
 - Typical title plus snippet based description (possibly mixed with images and/or video and/or metadata).
 - More complex structures which, using all indexed materials, are able to model specific information related with a concrete cultural heritage item such as a writer, a painter, an artwork and so on.

The figures below show two different possible conceptualisations of cultural objects in MultiMatch. Figure 6 shows a typical title plus snippet cultural object (with some extra information such as the source category) which links with a web page about *Van Gogh* works (note that in this case the cultural object has been rendered without any image), while Figure 7 shows a more detailed cultural object about *Van Gogh* which has been taken from the MultiMatch ontology.

The Vincent van Gogh Gallery
 The scroll "Birds and Flowers of the Four Seasons" (1808) is one of his most highly regarded works, as is the screen "Stream amid Summer and Autumn Flowers" ...
www.vangoghgallery.com/artistbios/Sakai_Hoitsu.html
 - 12k - Cached - Similar pages - Source Category: Cultural Heritage Institution

Fig. 8. Cultural object created using title plus snippet plus extra information such as the page source (cultural heritage site)

 Name: Vincent Van Gogh
 Birth: 1853
 Death: 1890
 Country: Holland
 Category: Painter

Fig. 9. Cultural object created using extracted metadata

- Means of navigating the material by place and time (using interactive maps and timelines)

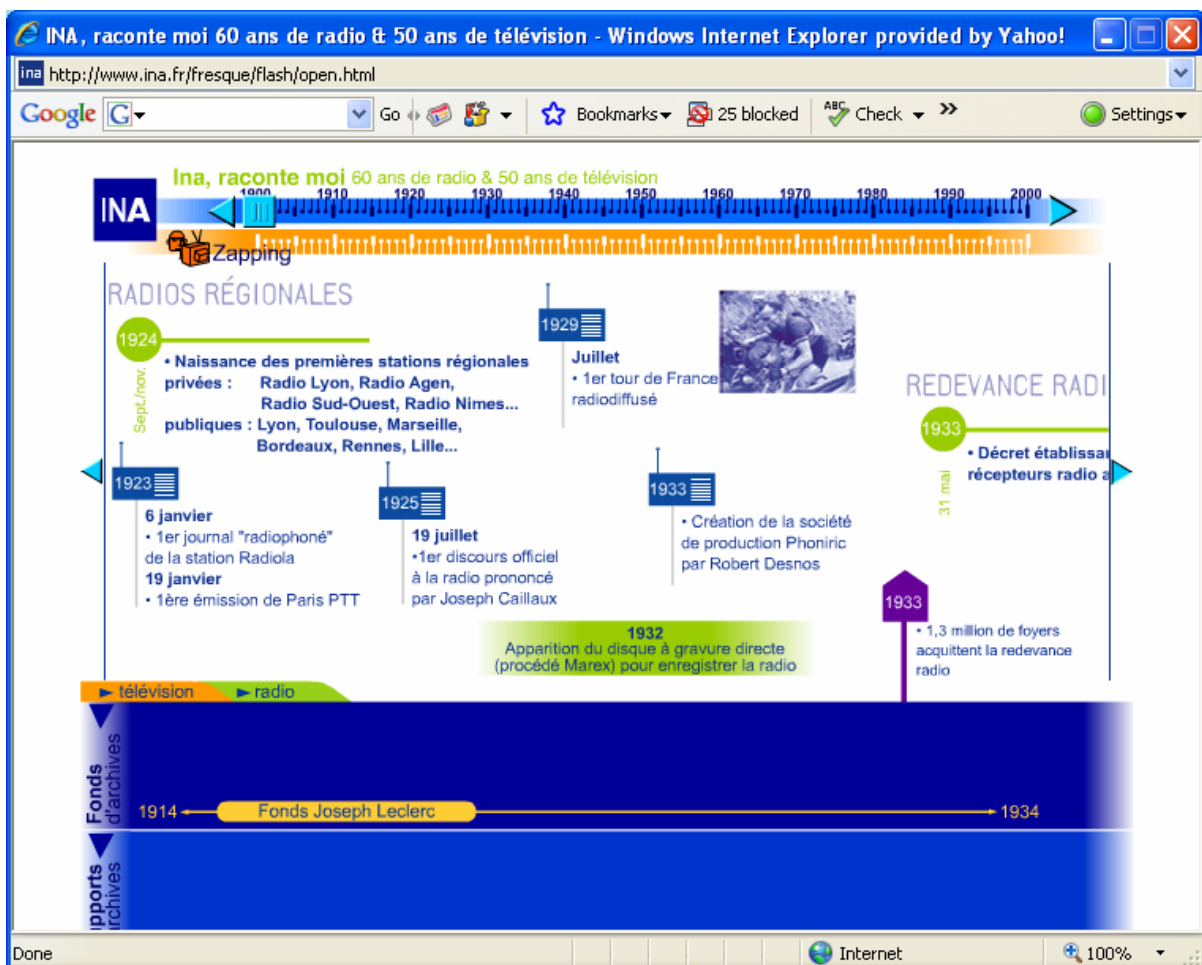


Fig. 10. Interactive timeline from French National Audiovisual Institute (www.ina.fr)

- **Multimodal querying:** This would involve using images to launch a search (in addition to text.) Therefore, it could include Query By Example (finding images visually similar to the submitted image.) It could also be expanded to find multimedia results by scanning a large text document (such as an article or an e-mail) and automatically extracting relevant keywords to use as a search.

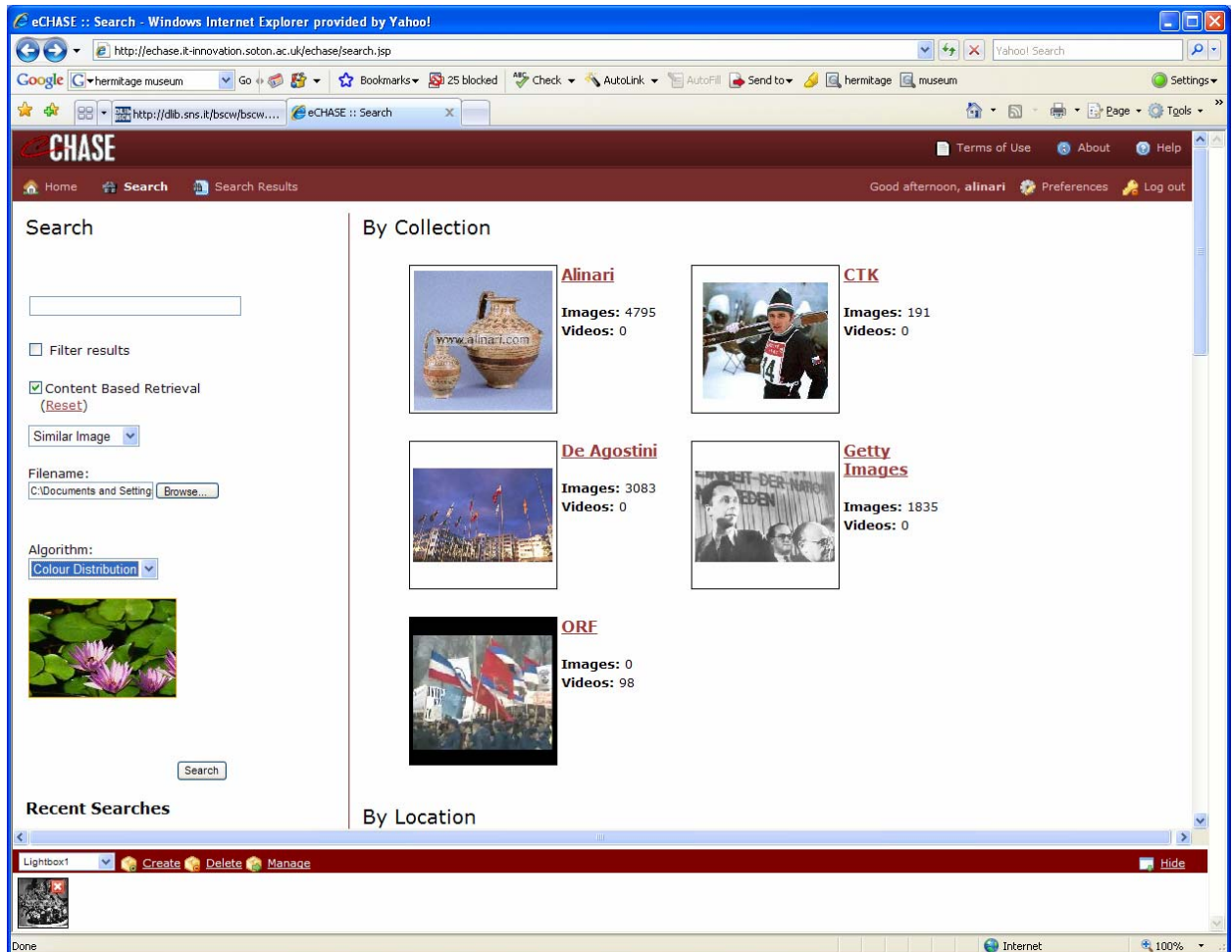


Fig. 11. Taken from eCHASE demo at <http://www.echase.org>

- More advanced means of clustering and refining results: Again, the expanded ontology could also be applied to the search results in order to enable clustering and refining the retrieved items based on a variety of criteria (for example, show only results affiliated with a given person.)

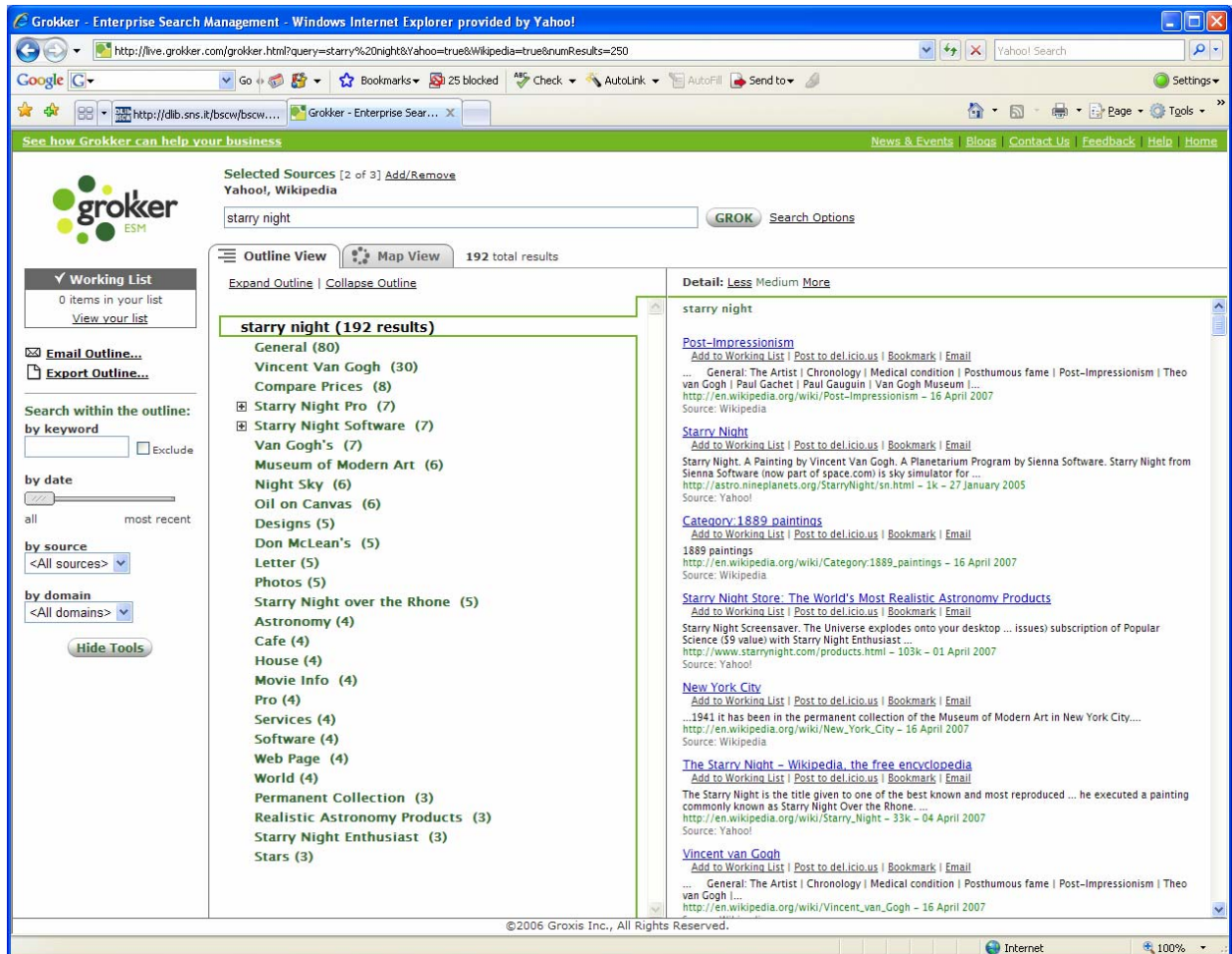


Fig. 12. Results clustering, filtering and detail presentation as shown at www.grokker.com

Below is a simple screenshot illustrating how the final prototype might look and incorporate some of these additional features into the interface:

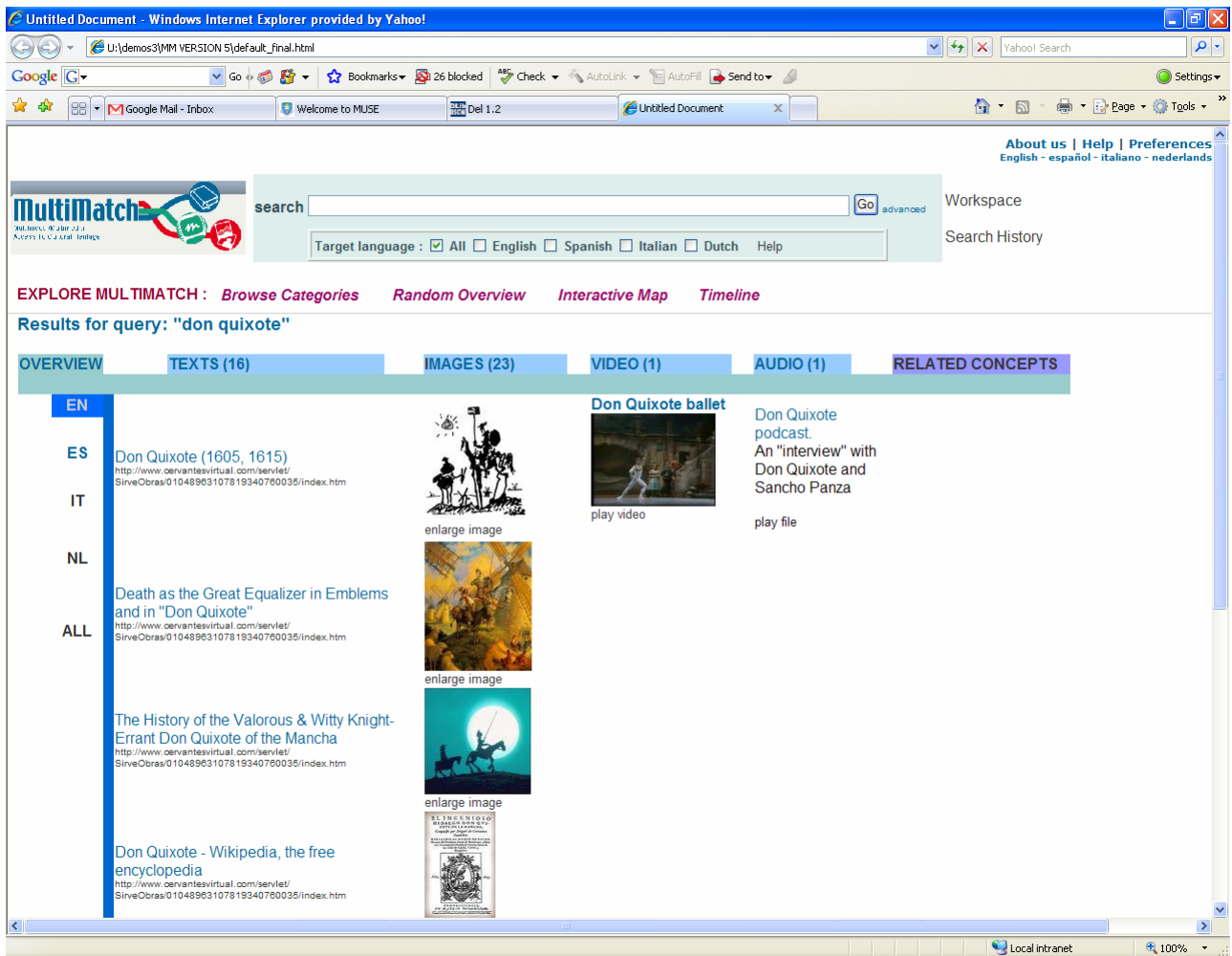


Fig. 13. Sample interface for final prototype including links to expanded functionalities

References

- Artiles, J., Gonzalo, J., Lopez-Ostenero, F., & Peinado, V. (2006). Are users willing to search cross-language? An experiment with the Flickr image sharing repository. <http://nlp.uned.es/pergamus/pubs/uned-iclef06.pdf>.
- Bates, M.J. (1989). The design of browsing and berrypicking techniques for the online search interface. *Online Review*, 13, 407-424.
- Beale, R. (2006). Improving Internet interaction: From theory to practice. *JASIST* 57(6): 829-33.
- Benjamins, V.R., Contreras, J., Blázquez, M., Doderó, J.M., Garcia, A., Navas, E., Hernandez, F., & Wert, C. (2004). Cultural Heritage and the Semantic Web. In *Proceedings of the First European Semantic Web Symposium*, 433-44.
- Boekelheide, K., Brown, A., Fu, X., Marchionini, G., Oh, S., Rogers, G., Saelim, B., Song, Y., Stutzman, F. (2006). Audio surrogation for digital video: A design framework. SILS Technical Report TR-2006-02.
- Brown, S., Ross, R., Gerrard, D., Greengrass, M., & Bryson, J. . (2006). RePAH: Research portals in the Arts and Humanities: A user analysis project. <http://repah.dmu.ac.uk/report>.
- Capstick, J., Diagne, A.K., Erbach, G. Uszkoreit, H., Leisenberg, A., & Leisenberg, M. (2000). "A system for supporting cross-lingual information retrieval." *Information Processing and Management* , 36(2), 275-289
- Chen, H. (2001). An analysis of image queries in the field of art history. *JASIST*, 52(3), 260-273.
- Choi, Y., & Rasmussen, E.M. (2003). Searching for Images: The Analysis of Users' Queries for Image Retrieval in American History. *JASIST*, 54(6), 498-511.
- Collins, K. (1998). Providing subject access to images: A study of user queries. *The American Archivist*, 61, 36-55.
- Cunningham, S.J., Bainbridge, D., & Masoodian, M. (2004). How people describe their image information needs: A grounded theory analysis of visual arts queries. *JCDL*, 47-48.
- Dalmau, M., Floyd, R., Jiao, D., & Riley, J. (2005). Integrating thesaurus relationships into search and browse in an online photograph collection. *Library Hi Tech*, 23(3), 425-452.
- Dorr, B., He, D., Luo, J., & Oard, D. (2003). "iCLEF at Maryland: Translation selection and document selection." In C. Peters (Ed.), *Working Notes for the CLEF 2003 Workshop*.
- Faulkner, X. (2000). *Usability Engineering*. Houndmills: Macmillan.
- Frost, C.O., Taylor, B., Noakes, A., Markel, S., Torres, D., & Drabenstott, K.M. (2000). Browse and Search Patterns in a Digital Image Database. *Information Retrieval*, 1, 287-313.



- Goto, K., & Cotler, E. (2004). *Web ReDesign 2.0: Workflow that Works*. Berkeley: New Riders.
- Gremett, P. (2006). Utilizing a user's context to improve search results. *JASIST* 57(6), 808-812.
- Hackos, J., & Redish, J. (1998). *User and task analysis for interface design*. New York: Wiley Computer Publishing.
- He, D., et al. (2003). Making MIRACLES: Interactive translingual search for Cebuano & Hindi. *ACM Transactions on Asian Language Information Processing*, 2(3), 219-244.
- Hearst, M. (1999). "User Interfaces and Visualization". In: Baeza-Yates, R. & Ribeiro-Neto, B. (eds.), *Modern Information Retrieval*, 257-323. New York: ACM Press.
- Hearst, M. (2000) *User Interfaces & Visualization for Information Access*, Tutorial presented at SIGIR2000 (<http://bailando.sims.berkeley.edu/talks/sigir-viz-tutorial-2000.ppt>).
- Hearst, M., et al. (2002). Finding the flow in web site search. *Communications of the ACM*, 45(9).
- Henninger, S., & Belkin, N. (1996). Interface issues and interaction strategies for information retrieval systems. *Conference companion on human factors in computing systems: Common ground*, 352-353.
- Ingwersen, P. and Järvelin, K. (2005). *The turn: integration of information seeking and retrieval in context*. Dordrecht, The Netherlands: Springer.
- Kravchyna, V. (2004). *Information needs of museum visitors: Real and Virtual*. PhD dissertation, University of North Texas.
- Levin, S. (2002). *Designing a user interface for a web-based multimedia information retrieval system*. MSc dissertation for Information Management, University of Sheffield.
- Marchionini, G. (1992). Interfaces for end-user information seeking. *Journal of the American Society for Information Science*, 43(2):156-163.
- Marchionini, G. (1995) *Information Seeking in Electronic Environments*, Cambridge University Press (May 26 1995), ISBN: 0521443725.
- Marlow, J. (2006). *Designing a localisation strategy for Tate Online: requirements and recommendations*. MSc dissertation for Master of Arts in Multilingual Information Management, University of Sheffield.
- Minerva Project (2006). *Multilingual Access to the digital European cultural heritage*. <http://www.mek.oszk.hu/minerva/survey/delir20060130>. [Accessed 16 August 2006]
- Mostafa, J., & Dillon, A. (1996). Design and evaluation of a user interface supporting multiple image query models. *Proceedings of 59th annual conference of the American Society of Information Science*, Baltimore, October 21-26.



- Ogden, W., Cowie, J., Davis, M., Ludovik, E., Nirenburg, S., Molina-Salgado, H., et al. (1999). "Keizai: An interactive cross-language text retrieval system." Paper presented at the Machine Translation Summit VII, Workshop on Machine Translation for Cross-Language Information Retrieval, Singapore, PRC.
- Ogden, W.C., & Davis, M.W. (2000). "Improving cross-language text retrieval with human interactions." Proceedings of the Hawaii International Conference on System Science (HICSS-33), Vol. 3.
- Peters, C. & Sheridan, P. (2001) "Multilingual information access." In Lectures on information Retrieval, M. Agosti, F. Crestani, and G. Pasi, Eds. Springer Lecture Notes In Computer Science Series, vol. 1980. Springer-Verlag New York, New York, NY, 51-80.
- Petrelli, D., Levin, S., Beaulieu, M., & Sanderson, M. (2006). Which user interaction for cross language information retrieval? Design issues and reflections. *JASIST*, 57(5), 709-722.
- Preece, J., Rogers, Y., and Sharp, H. (2002). Interaction design: Beyond human-computer interaction. New York: Wiley.
- Qu, Y., Eilerman, A.N., Jin, H., & Evans, D.A. (2000). The Effect of Pseudo Relevance Feedback on MT-based CLIR. Proceedings of Content-based Multimedia Information Access: Recherche d'informations assistee par ordinateur.
- Rose, D.E. (2006). Reconciling information-seeking behaviour with search user interfaces for the Web. *JASIST* 57(6): 797-799.
- Rose, D. and Levinson, D. (2004). Understanding User Goals in Web Search. In Proceedings of WWW 2004, New York, USA. ACM.
- Rubin, J. (1994). Handbook of usability testing: how to plan, design, and conduct effective tests. New York: Wiley.
- Shneiderman, B., Byrd, D., & Croft, W.B. (1997). Clarifying Search: A User-interface Framework for Text Searches. *D-Lib Magazine*.
- Shneiderman, B. (1998). Designing the User Interface: Strategies for Effective Human-Computer Interaction. Reading, MA: Addison Wesley.
- White, R. W. and Ruthven, I. (2006). A study of interface support mechanisms for interactive information retrieval. *JASIST* 57(7), 933-948.
- Zazo Rodriguez, A.F., Figuerola, C., Alonso Berrocal, J.L., & Fernandez Marcial, V. (2005). iCLEF 2005 at REINA-USAL: Use of free on-line machine translation programs for interactive cross-language question answering. http://www.clef-campaign.org/2005/working_notes/workingnotes2005/rodriguez05.pdf.



APPENDIX A – Sites used for competitor analysis

Cultural Heritage related sites

Alinari Business site	http://business.alinari.it
American Memory, Library of Congress	http://memory.loc.gov/ammem/index.html
Archaeological Records of Europe	http://ads.ahds.ac.uk/arena/search/
Archaeology Data Service demo	http://www.common-info.org.uk/enhanceddemonstrator.htm
Artcyclopedia	www.artcyclopedia.com
BBC radio and TV catalogue	http://open.bbc.co.uk/catalogue/infax
Beeld en Geluid website	www.beeldengeluid.nl
Birth of TV project	http://www.birth-of-tv.org/birth/
Bricks workspace	http://dev.brickscommunity.org/BRICKS_Workspace
British Pathé (film archive)	http://www.britishpathe.com/
Canadian Broadcasting Archive	http://archives.cbc.ca/index.asp?IDLan=1
Corbis	http://pro.corbis.com/search/searchFrame.aspx
Courtauld Institute of Art	www.artandarchitecture.org.uk
EASE history	http://www.easehistory.org/index2.html
e-culture demonstrator	http://e-culture.multimedien.nl/demo/facet
e-culture search demonstrator	http://e-culture.multimedien.nl/demo/search
English Heritage Viewfinder	http://viewfinder.english-heritage.org.uk
Europa Audiovisual Service	http://ec.europa.eu/avservices/home/index_en.cfm
exalead	http://www.exalead.com/search
Fine art	http://dart.fine-art.com/featured/default.asp
Flamenco faceted browse demo	http://orange.sims.berkeley.edu/cgi-bin/flamenco.cgi/famuseum/Flamenco
Fotolia stock photography archive	www.fotolia.com
France in America	http://international.loc.gov/intldl/fiahtml/fiahome.html
French cultural history portal	http://www.numerique.culture.fr/mpf/pub-fr/index.html
Hermitage Museum	http://www.hermitagemuseum.org/
Historical Voices Portal	http://www.historicalvoices.org/galleries.php
History Channel	www.history.com/media.do
Homework help	http://www.cosmeo.com/
INA (French national audiovisual institute)	http://www.ina.fr/archivespour tous/index.php
INA radio and TV timeline	http://www.ina.fr/fresque/
International Architecture Database	www.archinform.net



Internet Archive	http://www.archive.org
Intute Arts & Humanities Portal	http://intute.ac.uk/artsandhumanities
Library Archives Canada: Art and Photography	http://www.collectionscanada.ca/art-photography/index-e.html
Louvre Museum	www.louvre.fr
Michael UK	www.michael-culture.org.uk
Museo Bagatti	http://www.museobagattivalsecchi.org/defaultall.htm
MuseoSuomi	http://www.museosuomi.fi/
Museum of Modern Art, New York	http://www.moma.org/collection/browse_results.php?object_id=80013
Museumland	www.museumland.net
NASA Multimedia	http://nix.nasa.gov
Numerique.culture.fr	http://numerique.culture.fr/mpf/pub-fr/index.html
Open Video Project	http://www.open-video.org/
Portable antiquities	http://www.finds.org.uk/
Prelinger historical film archive	http://www.archive.org/details/prelinger
Rijksmuseum	http://www.rijksmuseum.nl/index.jsp
Screenonline (British film history)	http://www.screenonline.org.uk/
Screenonline Children's tour	http://www.screenonline.org.uk/tours/childrens/tourchildrens1.html
Staffordshire PastTrack	www.staffspasttrack.org.uk
Tate Online	www.tate.org.uk
Timeline of art history	http://www.metmuseum.org/toah/splash.htm
Timemap (interactive timeline demo)	http://www.timemap.net/
Today in history	http://www.todayinhistory.de/
UIUC Digital Gateway	http://nergal.grainger.uiuc.edu/cgi/b/bib/bib-idx
Virtual silver screen	http://www.collectionscanada.ca/silverscreen/
World Heritage	http://whc.unesco.org/en/map



Multimedia search engines

AllTheWeb	http://www.alltheweb.com
Altavista	http://www.altavista.com
Ask	http://www.ask.com
Clusty	http://clusty.com
Corbis	http://pro.corbis.com
Ditto.com	http://www.ditto.com
Exalead	http://www.exalead.com
Getty	http://creative.gettyimages.com
Google Images	http://www.google.co.uk/imghp?hl=en&tab=wi&q=
Google Video	http://video.google.co.uk/
Lycos	http://www.lycos.com
Microsoft Image Search	http://www.live.com
Picsearch	http://www.picsearch.com
Singingfish	http://www.singingfish.com
Yahoo	http://www.yahoo.com
YouTube	http://www.youtube.com

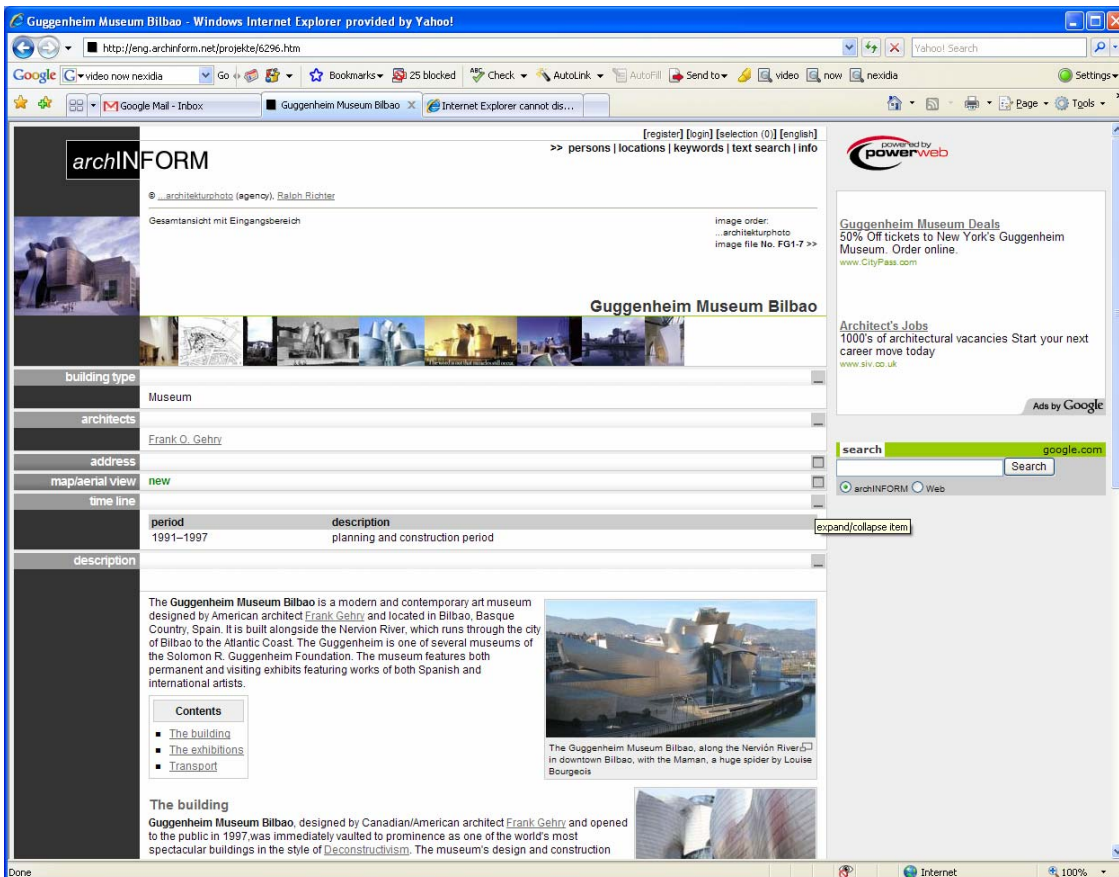
Browsing capabilities

Alinari Business	http://business.alinari.it
American Memory	http://memory.loc.gov/ammem/index.html
Artcyclopedia	http://www.artcyclopedia.com
ARTstor	http://www.artstor.org
Courtauld Institute of Art Cushman Photo Collection	http://webapp1.dlib.indiana.edu/cushman/browse/index.jsp http://orange.sims.berkeley.edu/cgi-bin/flamenco.cgi/spiro/Flamenco
Flamenco Architecture	http://orange.sims.berkeley.edu/cgi-bin/flamenco.cgi/famuseum/Flamenco
Flamenco Fine Arts	http://orange.sims.berkeley.edu/cgi-bin/flamenco.cgi/nobel/Flamenco
Flamenco Nobel Prize winners	http://www.hermitagemuseum.org
Hermitage	http://www.louvre.fr
Louvre	http://www.michael-culture.org.uk
MICHAEL UK	http://www.moma.org
MoMA	http://e-culture.multimedien.nl/demo/facet
MultimediaN	http://www.oceansbridge.com
Oceansbridge	http://www.bbc.co.uk/bbcfour/paintingflowers/
Painting Flowers - BBC	http://www.rijksmuseum.nl
Rijksmuseum	http://www.tate.org.uk
Tate	http://www.wga.hu
Web Gallery of Art	

APPENDIX B – SCREENSHOTS OF GOOD EXAMPLES

The screenshots that follow are taken from various pre-existing websites. Currently, no single site offers the combination of all multilingual, multimedia, layout-related, and interactive functionalities that MultiMatch plans to ultimately incorporate. However, each of the following examples was chosen to demonstrate one (or more) aspects of functionality that MultiMatch might wish to incorporate in the final system.

Example 1 – Aggregated results presentation



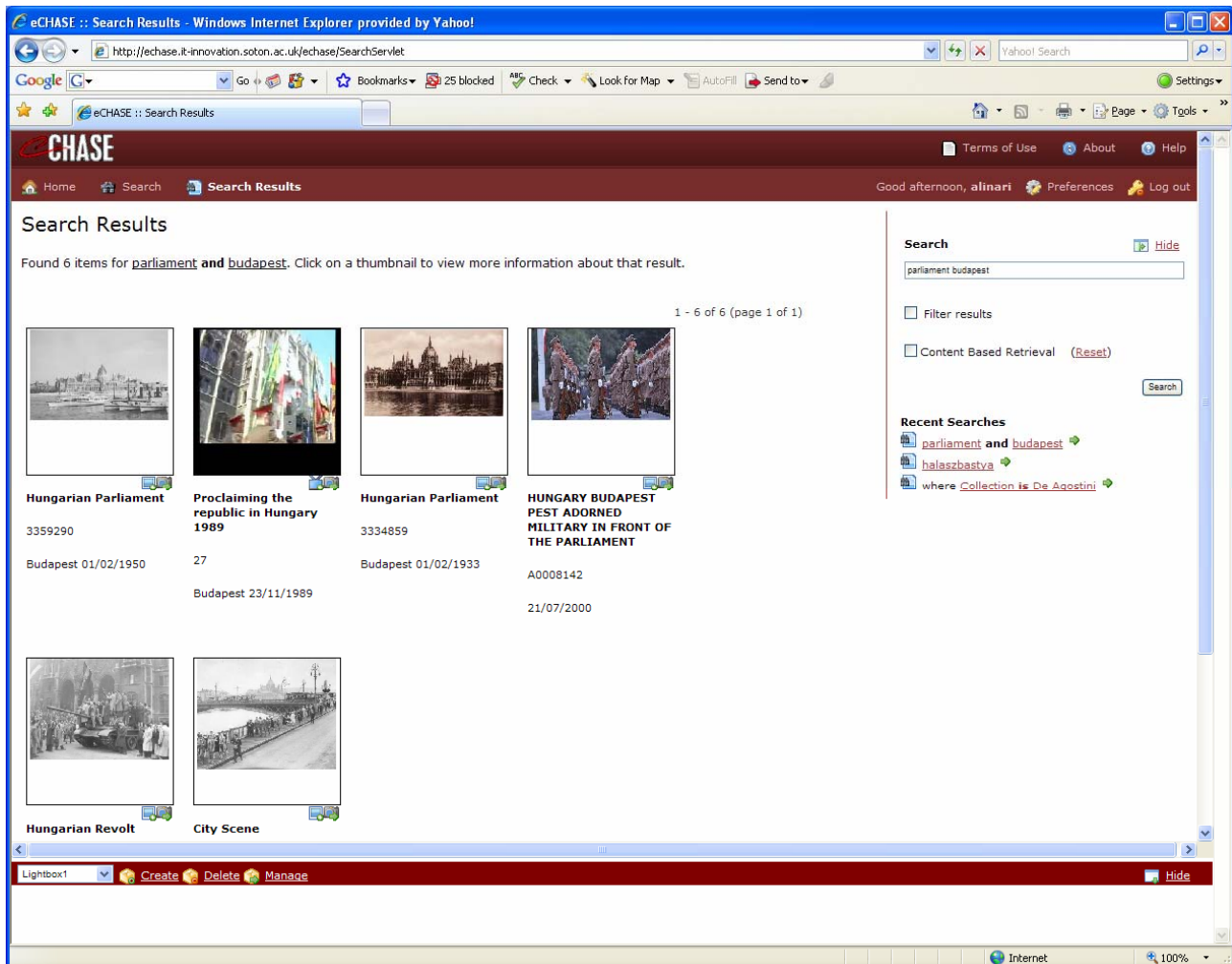
Archinform (International Architecture Database)

<http://www.archinform.net>

This site provides a range of information about various architects and buildings. Browsing by person's name, location, and keyword is possible, along with free text search.

The screenshot above shows a sample results presentation for the Guggenheim Bilbao building. It is possible to access a variety of aggregated information about this building including type, architects, timeline, and description. It is also possible to view the location on Google maps. Any of these categories can be minimized from the overall view. Finally, at the bottom there are external links to related sites.

Example 2 – Search results presentation, filtering, saving (in lightbox), and “recent searches”



eCHASE Project Demo: <http://www.echase.org>

This project provides access to image and video material, predominantly relating to Central and Eastern Europe. This screen shot shows a sample results page view.

Functionalities shown here include a lightbox for saved items (located in a frame at the bottom of the page,) the possibility of viewing the search history, as well as “find more like this” options for each search result (although what exactly this means is not particularly clear.)

More advanced search options include the ability to filter results or to conduct visual similarity search for images. Although the specific ways in which this interface presents various functionalities may or may not be ideal, this system demonstrates one approach to designing a system that is very similar to MultiMatch.

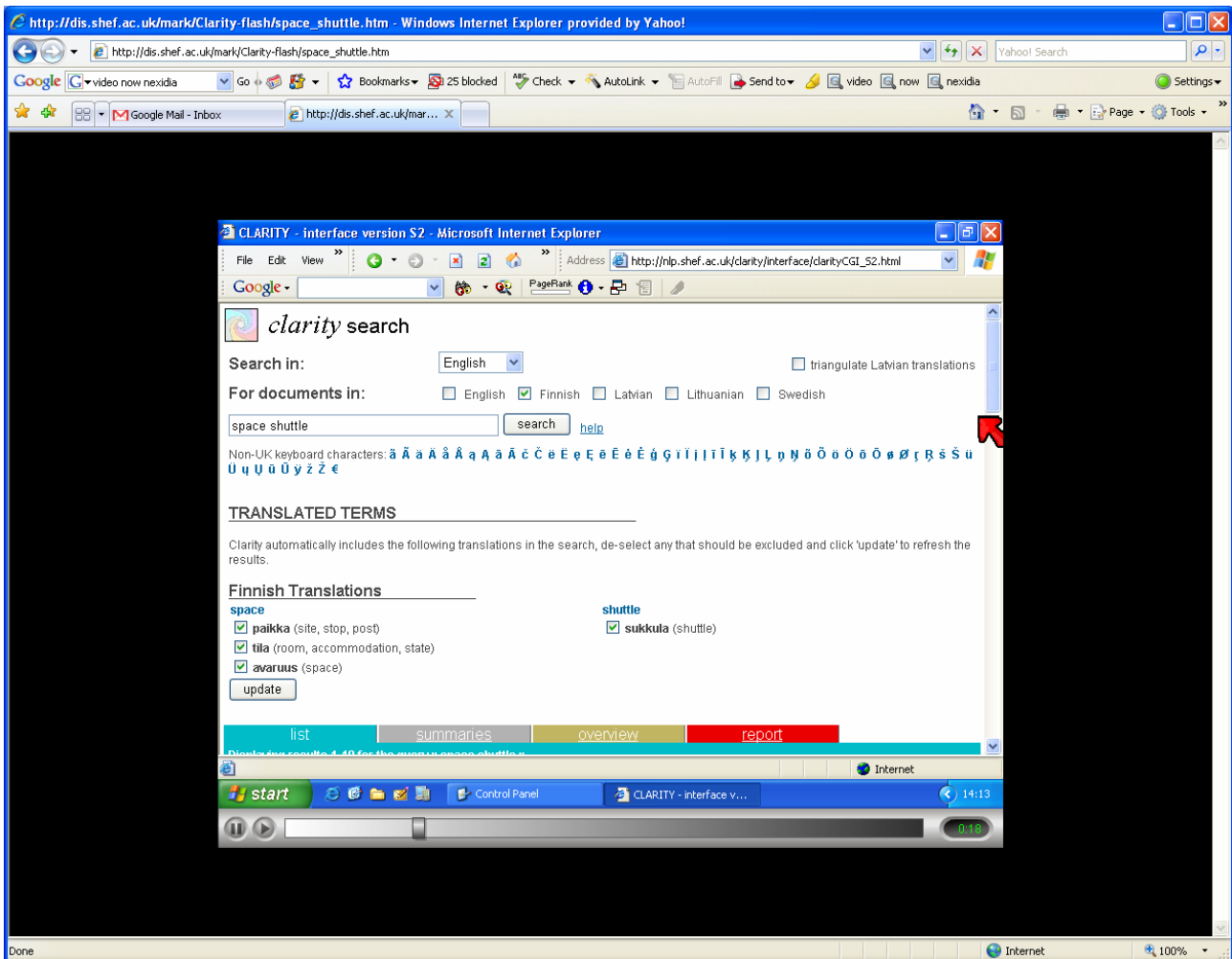
Example 3 – Detailed image results view (including metadata)



Alinari Business site: <http://business.alinari.it>

This screenshot illustrates a detailed results view for an image, providing metadata including title, artist, and keywords. Clicking on any of the links will launch a new query on the associated topic (i.e. clicking on “Giotto di Bondone” will retrieve all works that were created by that artist).

Example 4 – Visualization of interactive cross language search results

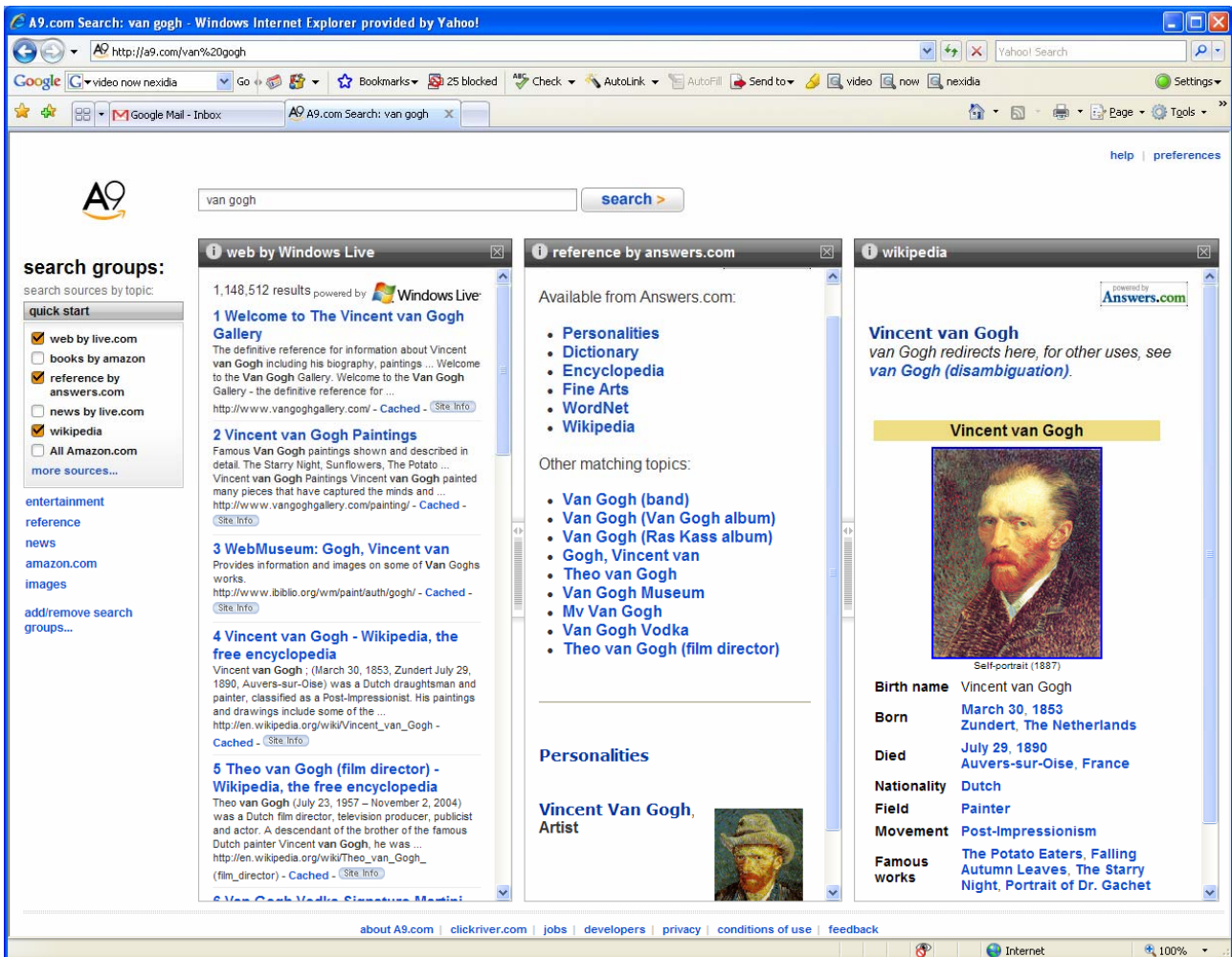


CLARITY cross-language system demo: http://dis.shef.ac.uk/mark/Clarity-flash/space_shuttle.htm

This animated demo illustrates how the CLARITY system enables cross-language searching and translation feedback. Under the heading of “Finnish Translations” it is possible to see all of the possible translations for the query terms (in this case, “space” and “shuttle.”) The word “space” has several alternative meanings, which are displayed to the user with the appropriate back-translations.

If any of these alternatives are not considered relevant, the user can un-check them and then update to refresh the results. This approach was tested with users and found to be satisfactory. However, what the demo does not illustrate is how the interface appears when several languages are chosen for translation, and how it would work should the user wish to update the query in several different languages. It is possible that if all four were selected, the interface could become very cluttered and chaotic, and that the mechanics of updating the search could become complicated.

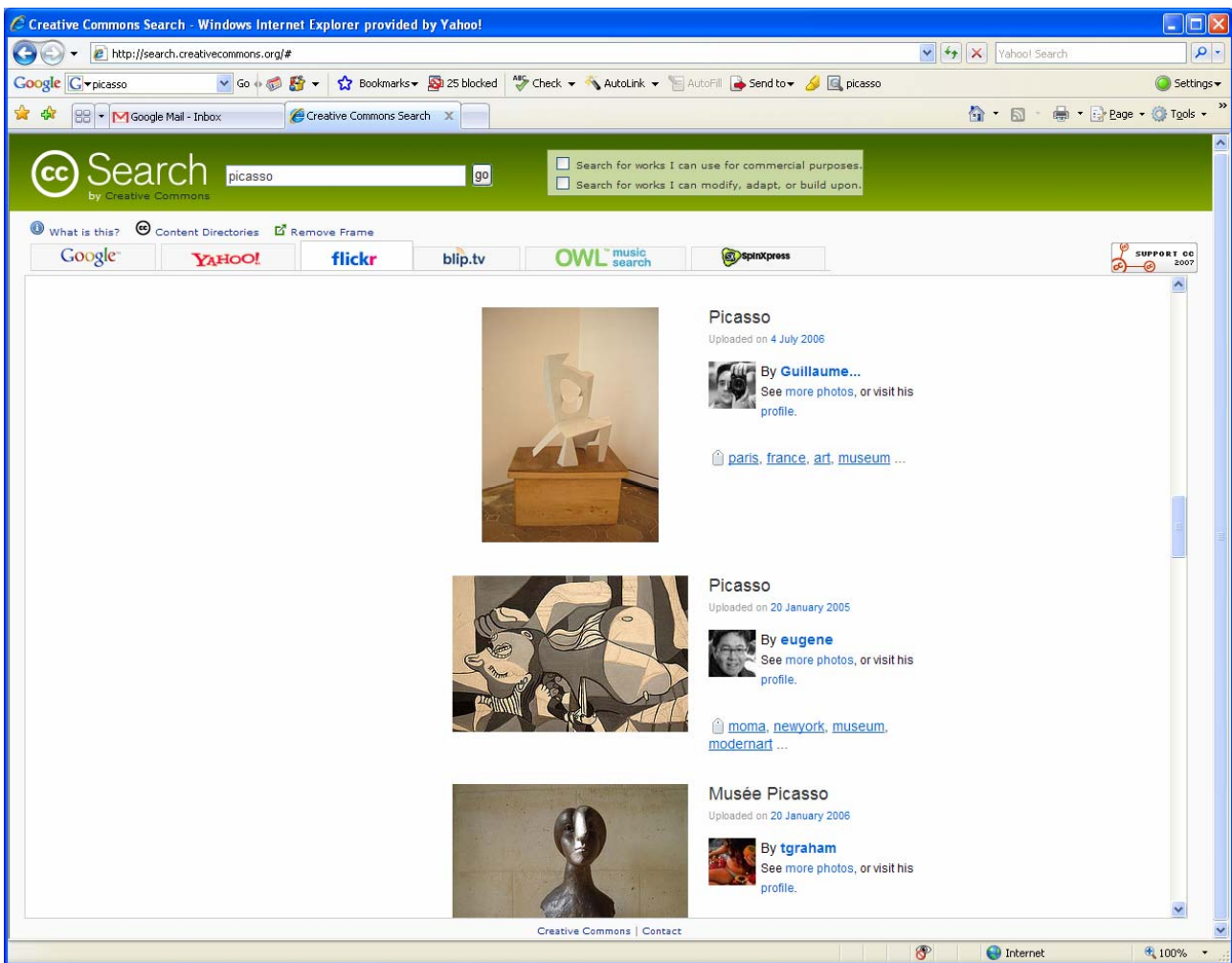
Example 5 – Presentation of results from different sources (columns)



A9 Search Engine: <http://www.a9.com>

This system searches a variety of sources (selectable via the check boxes on the left hand side) and then displays the various results in columns, as shown above. In this way, it is possible to get a quick overview of the various results and to compare them with a brief glance. The columns are re-sizable and the results presentation can be interactively changed and updated (both by selecting or de-selecting sources and by re-arranging the existing columns.) This was used as the inspiration for the default results presentation.

Example 6 – Alternative presentation of results from different sources (tabs)



Creative Commons Search: <http://search.creativecommons.org>

Again, this is a system that searches a variety of sources (including Google, flickr, and blip.tv) to provide a range of multimedia results. This site has chosen a different presentation approach to that of A9, opting to display the various results in tabs rather than in columns. Although it is not possible to see all the results at one time, it is easy to switch between them using the tabs. This model inspired the design of the specialised results display interfaces.

Example 7 – Visualization (and search) of audio material

The screenshot shows a web browser window displaying the PodZinger search results for the query "impressionist painting". The page features a search bar with the query and a dropdown menu set to "All Series". Below the search bar, there are several search results, each with a thumbnail, a title, a date, and a snippet of the transcript. The search results are sorted by date and relevance. A sidebar on the left contains links to various categories like "Local Government" and "Government and Politics". A "Windows Vista" advertisement is visible on the right side of the page.

PodZinger: <http://www.podzinger.com>

This site enables the search of audio (podcast) content based on the automatically generated transcript. It is possible to listen to the entire podcast as well as to jump directly into the part of the file where the search terms appear (highlighted in orange and displayed alongside the timestamp information where the word appears.) A snippet of the transcript is provided to put the word in context and help the user judge whether it is relevant. Due to the fact the transcripts are automatically generated, a high proportion of the results can contain errors or be irrelevant, so this technology is not without its faults.

APPENDIX C – RESPONSES TO WIND SURVEY

As previously mentioned in section 2.3.5, the WIND survey received a total of 414 responses. The survey questions were published in Italian; therefore, the following is a translation of the original version.

How frequently do you search for cultural material on the web?

%

Weekly	33.1
Daily	25.4
Monthly	21.7
Rarely	14.7
Never	1.4

What is the reason/motivation for your search?

It was possible to choose more than one; hence percentages add up to more than 100%

%

Personal	65.2
Tourism	39.6
Studies	31.1
Professional	30.2

How frequently do you look for the following types of information?

Top 3 responses from each category of media were taken.

	Daily or weekly (sum of Daily column+weekly column)	Daily	Weekly	Monthly	Rarely
Images	66.2	30.7	35.5	15.5	
Articles	61.4	33.1	28.3	19.3	
Videos	16.7		16.7	14.0	30.4

What languages do you know?

Blank responses are not included here but presumably those mean the same as “no knowledge.” Top 3 answers for each language are listed here; languages are in order of relative ability.

	Native	Fluent	Read/write	Read but cannot write	No knowledge
Italian	85.3	2.7	2.7		
English		17.9	41.1	22.9	
French			22.7	26.6	14.7
Spanish			7.7	24.7	22.5
German			3.6	8.9	41.3
Dutch					47.6
Portuguese				3.4	45.7
Other					

How would you rate the importance of the following sources of cultural information?

	% responding Essential OR useful (the sum of “essential” and “useful” columns)
Regular visits to specialised sites (museums, universities)	60.1
Specialised search services	59.4
Online dictionaries (Wikipedia)	58.3
Book and publication search engines	50.8
Catalogues and articles	50.0
Image search engines	49.8
Multimedia databases	45.7
Video search engines	37.7
General search services	35.3
Links to sites suggested by experts	33.8
Web directories (dmoz, Yahoo!)	32.8
Image sharing sites (Flickr)	27.5

Subscription to specialised newsletters	26.8
Video sharing sites (YouTube)	25.9
RSS feeds from specialised sites	24.4
Unofficial sources (blogs, personal sites)	20.2

How often do you search for:

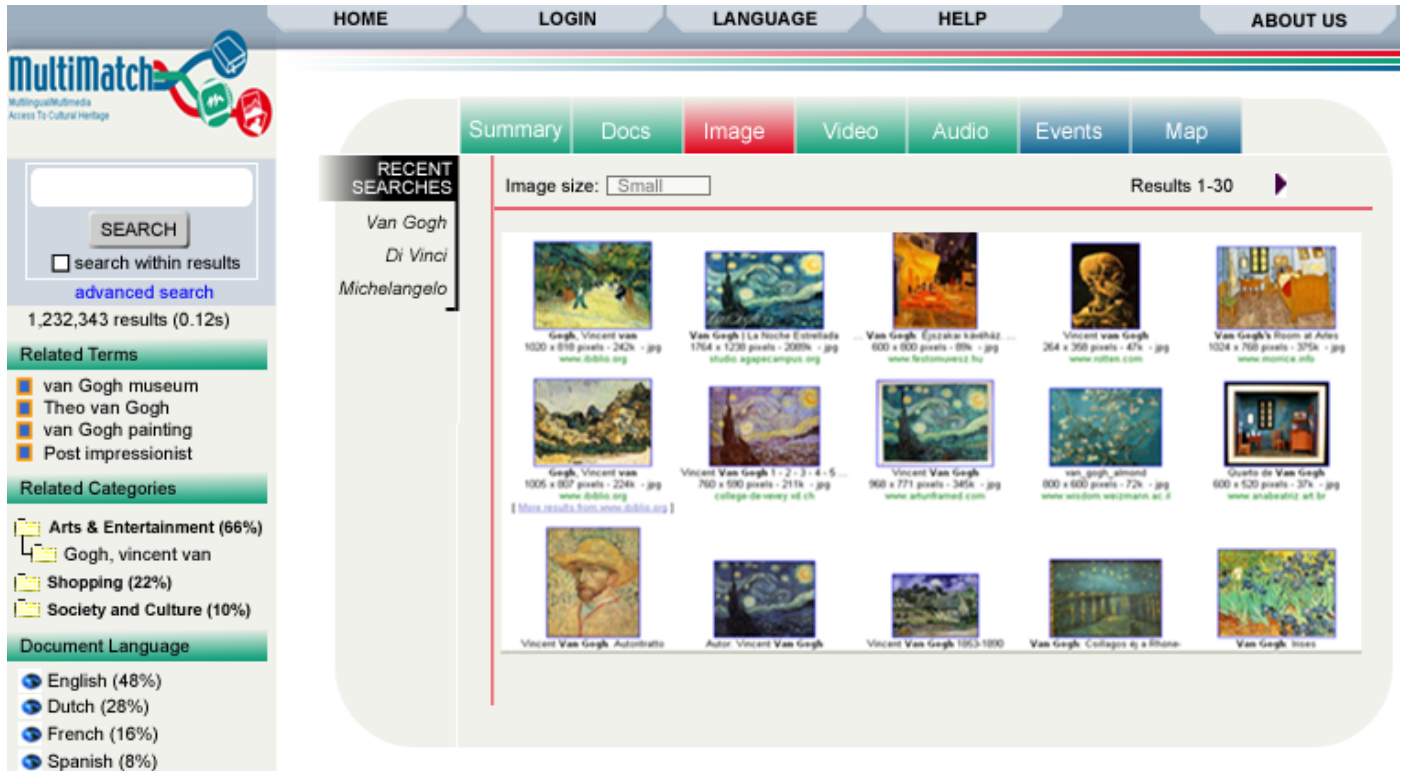
	Daily OR weekly
Particular sites (Uffizi)	30.7
Cultural events (exhibit on Caravaggio)	28.3
Authors (Giacomo Leopardi)	28.0
Events/news (Reopening of Scrovegni Chapel)	23.6
Artworks (Divine Comedy)	20.8
Periods (Romanticism)	19.3
Academic publications (The construction of Brunelleschi's dome)	18.6

If you had a search engine able to correlate/interrelate information, which types of relationships would you find most useful?

Associations between	Essential or useful
Masterpieces AND authors	57.8
Works, authors, or sites AND cities	57.7
Places, people, or things AND authors	52.7
Authors, works, or sites AND keywords	49.5
Authors AND authors/artists	46.9
Things in common between two or more authors	46.2
Relative popularity of authors, works, or sites	39.9
Opinions/feelings/moods AND works/authors/sites	31.4

APPENDIX D – PRELIMINARY DESIGNS

Screen shot from kick-off meeting



UNED Version 1

Macromedia Flash Player 8

File View Control Help

MultiMatch

All Authors Artworks Sites News

flowers

Search












Sign in >>

Images Video Browse MultiMatch Contents

Results for "flowers" : 15 web pages | 6 authors | 5 artworks | 2 news

Categorized Results

Sort by Category Sort by Concept edit

Web Pages	Authors	Artworks	Sites	News
<p>Tate Modern Current Exhibitions Fischli & Weiss</p> <p>Fischli and Weiss: Flowers & Questions. A Retrospective - About Visiting information Book tickets Events & Education Room Guide Films Further Resources ...</p> <p>www.tate.org.uk/modern/exhibitions/fischliandweiss/default.shtm - 20k - Cached - Similar pages - Source Category: Cultural Heritage Institution</p> <p>Daniel Brown - 40 artists, 40 days</p> <p>From the Flower Series 1999-present. In Olympic Colours ... Originally started in 1999, Daniel Brown's Flower Series uses advanced mathematics to generate ...</p> <p>www.tate.org.uk/40artists40days/daniel_brown.html - 6k - Cached - Similar pages - Source Category: Cultural Heritage Institution</p> <p>Iate Collection Roses by Leonard McComb</p> <p>Leonard McComb from Blossoms and Flowers, Roses 1975-9 - © Leonard McComb. from Blossoms and Flowers (P07501-P07506; complete). Roses 1975-9 ...</p> <p>www.tate.org.uk/servlet/ViewWork?cgrouoid=999999961&workid=9069&searchid=10770 - 16k - Cached - Similar pages - Source Category: Cultural Heritage Institution</p> <p>The Vincent van Gogh Gallery</p> <p>The scroll "Birds and Flowers of the Four Seasons" (1808) is one of his most highly regarded works, as is the screen "Stream amid Summer and Autumn Flowers" ...</p> <p>www.vangoghgallery.com/artistbios/Sakai_Hoitsu.html - 12k - Cached - Similar pages - Source Category: Cultural Heritage Institution</p> <p>Anales galdosianos [Publicaciones periódicas], Año XXVI, 1991 ...</p> <p>This manuscript contains more examples of Galdós's art work than any other: drawings of flowers and an elaborate sketch of a human head (f. ...</p> <p>www.cervantesvirtual.com/servlet/SirveObras/12615097557815940765657/p0000003.htm - 59k - Cached - Similar pages - Source Category: Cultural Heritage Institution</p> <p>Result Page: 1 2 3 Next</p>	<p> Vincent Van Gogh (1853-1890) Painter - Holland</p> <p> Leonard McComb (1930) Painter - United Kingdom</p> <p> Daniel A. Brown (1950) Art and technology - USA</p> <p> Benito Perez Galdós (1843-1920) Literature - Spain</p> <p> Peter Fischli (1952) Sculptor - Switzerland</p> <p> David Weiss (1946) Sculptor - Switzerland</p>	<p> Title: Sunflowers Collection: Unknown Vincent Van Gogh</p> <p> Title: Flowers Collection: Unknown Vincent Van Gogh</p> <p> Title: Tulips Collection: From Blossoms and Flowers Leonard McComb</p> <p> Title: Roses Collection: From Blossoms and Flowers Leonard McComb</p> <p> Title: Unknown Collection: Flower series (1999 - present) Daniel Brown</p>	<p>Tate Gallery United Kingdom</p> <p>Prado Museum Spain</p>	<p>Van Gogh dispute is coming to a head Sydney Morning Herald, Australia - Oct 11, 2006 ... of Victoria, Dr Gerard Vaughan, is optimistic that research under way in Amsterdam will confirm the authenticity of the gallery's only Van Gogh painting. Head ...</p> <p>Van Gogh exhibit focuses on wheat, harvest themes KSIWO, OK - Oct 21, 2006 DALLAS A vibrant painting Vincent Van Gogh finished weeks before his death is the centerpiece of a new exhibit at the Dallas Museum of Art. ...</p>

Copyright MultiMatch 2006

This was one of UNED's initial designs which illustrated the idea behind the default search (in which several types of related results are all displayed on the same page.)

USFD Version 1



This design emerged after a USFD brainstorming session and illustrates a sample image results page.

UNED Version 2

Documento sin título - Windows Internet Explorer provided by Yahoo!

U:\demos2\mm VERSION 1\MultiMatch\MultiMatch\index.html

Google

MultiMatch
MultilinguaMultimedia
Access to Cultura - heritage

Websites | Video | Images | Audio

Search

Sign In >>
Preferences

View Results by Original Language

English (50)
Spanish (20)
Italian (35)
Dutch (40)

RESULTS FOR THE QUERY flores in ENGLISH.

Your query was translated as FLOWERS (to flores). Click on a word to select an alternative meaning

TIP: You can click on the star next to each search item to save it on your workspace.

View Results by Media

Webbsites (75)
Images (40)
Video (15)
Audio (5)

Filter Results by MultiMatch Categories

Authors

Van Gogh (12)
El Greco (10)
Velazquez (7)

Works of Art

Sunflowers (10)
Las Meninas (7)

Art Periods

Romanticism (10)

Starred

Description

BBC - Painting Flowers - Van Gogh's Sunflowers

About Vincent van Gogh's Sunflowers, on BBC Four's Painting Flowers site. www.bbc.co.uk/bbcfour/paintingflowers/paintings/sunflowers_van_gogh.shtml - 51k - cache - Similar Pages

JSTOR: Notes on the Early el Greco

There are precisely the same purple tones, both in the flowers and in the Modena triptych-those flowers, which are quite characteristic of El Greco and ... links.jstor.org/sici?sici=0951-0788(193901)74:430%3C28:NOTEEG%3E2.0.CO;2-C - Similar Pages

JSTOR: The Masterpieces by Velazquez in the Imperial Gallery at Vienna

Two of the children's portraits are the master-works of Velazquez in the rendering 1 ... Note the glass jar, the flowers with impression and freshness of ... links.jstor.org/sici?sici=0951-0788(190407)5%3A16%3C338%3ATMBVIT%3E2.0.CO%3B2-P - Similar Pages

Vincent Van Gogh Still live with flowers art print poster ...

Still live with flowers Vincent Van Gogh art prints posters. Buy Vincent Van Gogh Still live with flowers art prints and posters from Easyart.com in the UK.

Images

Starred

Image

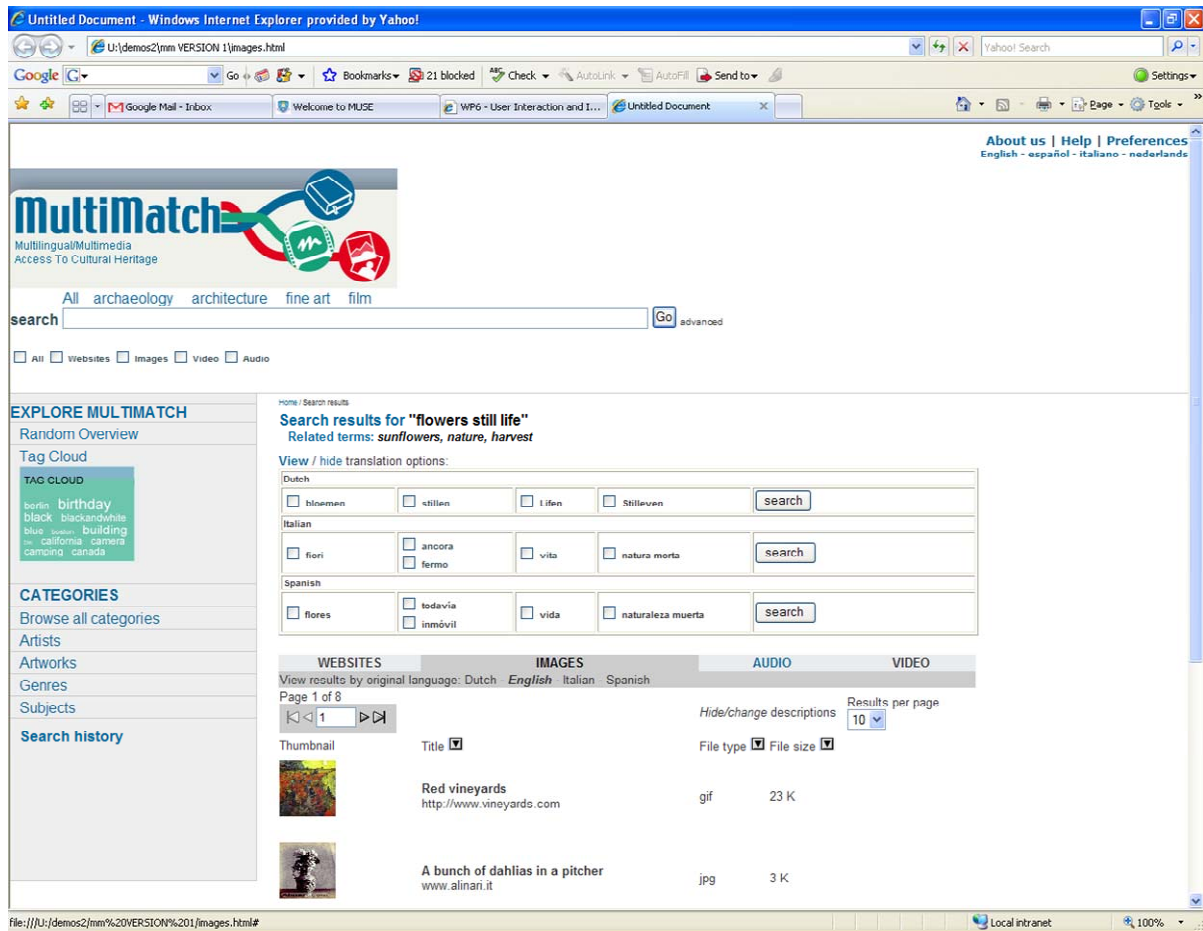
Description

Sunflowers
www.sunflowers.com



Red vineyards
http://www.vineyards.com

In this design, UNED responded to USFD version 1 and added enhanced functionalities such as the ability to “star” relevant items, and to refine the search results by selecting the languages and media types (as well as other categories) of interest.

USFD Version 2

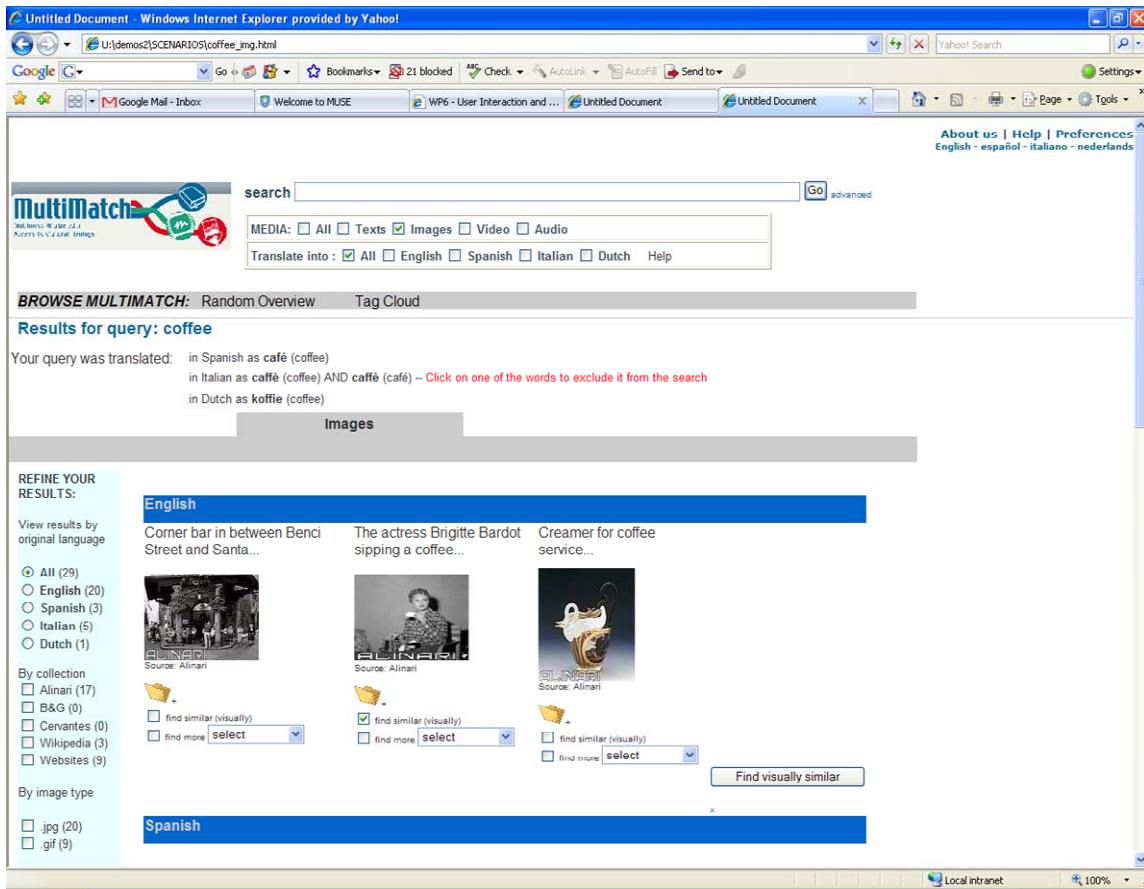


The screenshot displays the MultiMatch web application interface. The main content area shows search results for the query "flowers still life". The results are presented in a table with columns for "Thumbnail", "Title", "File type", and "File size". The first result is "Red vineyards" from "http://www.vineyards.com" with a file type of "gif" and a size of "23 K". The second result is "A bunch of dahlias in a pitcher" from "www.alinari.it" with a file type of "jpg" and a size of "3 K".

Thumbnail	Title	File type	File size
	Red vineyards http://www.vineyards.com	gif	23 K
	A bunch of dahlias in a pitcher www.alinari.it	jpg	3 K

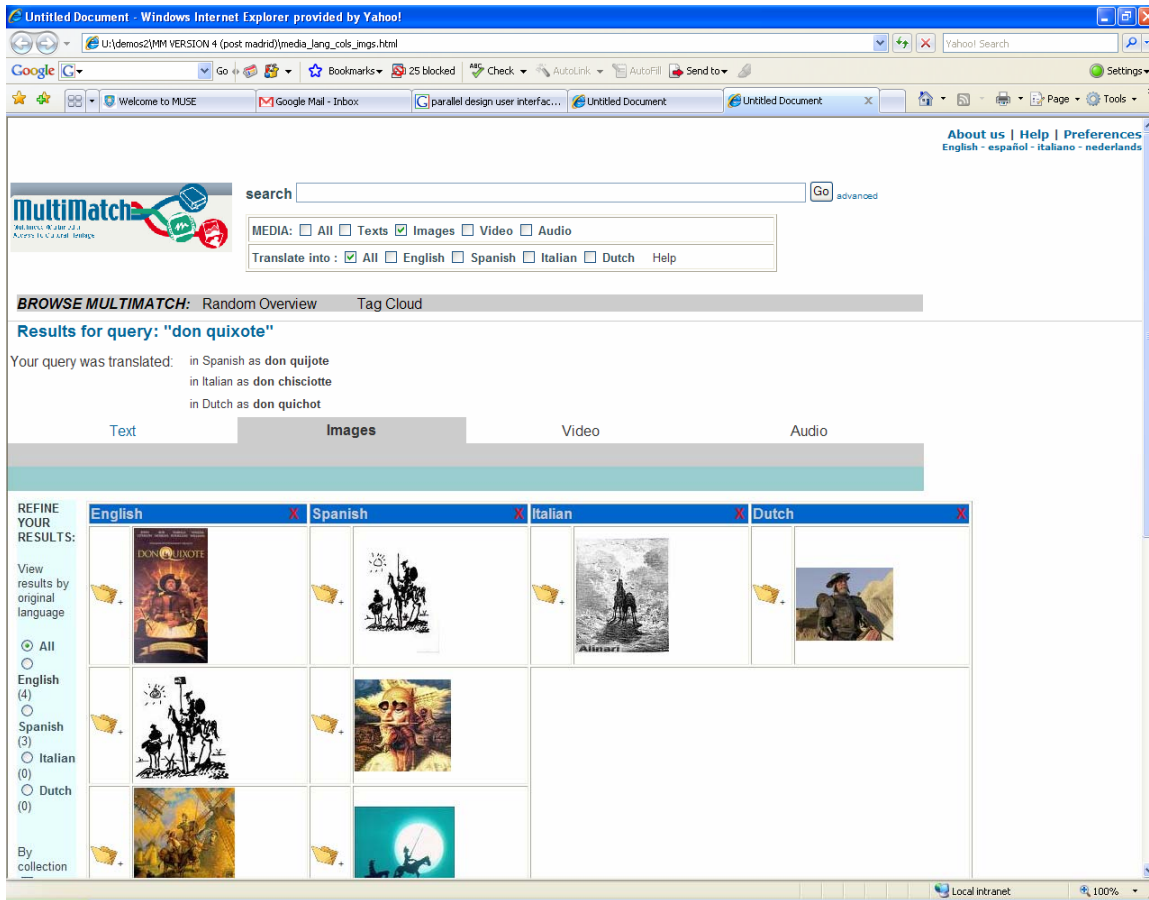
In this example, the cross-language interaction was explored further by introducing some sort of translation feedback mechanism, based on the CLARITY approach.

Version 3



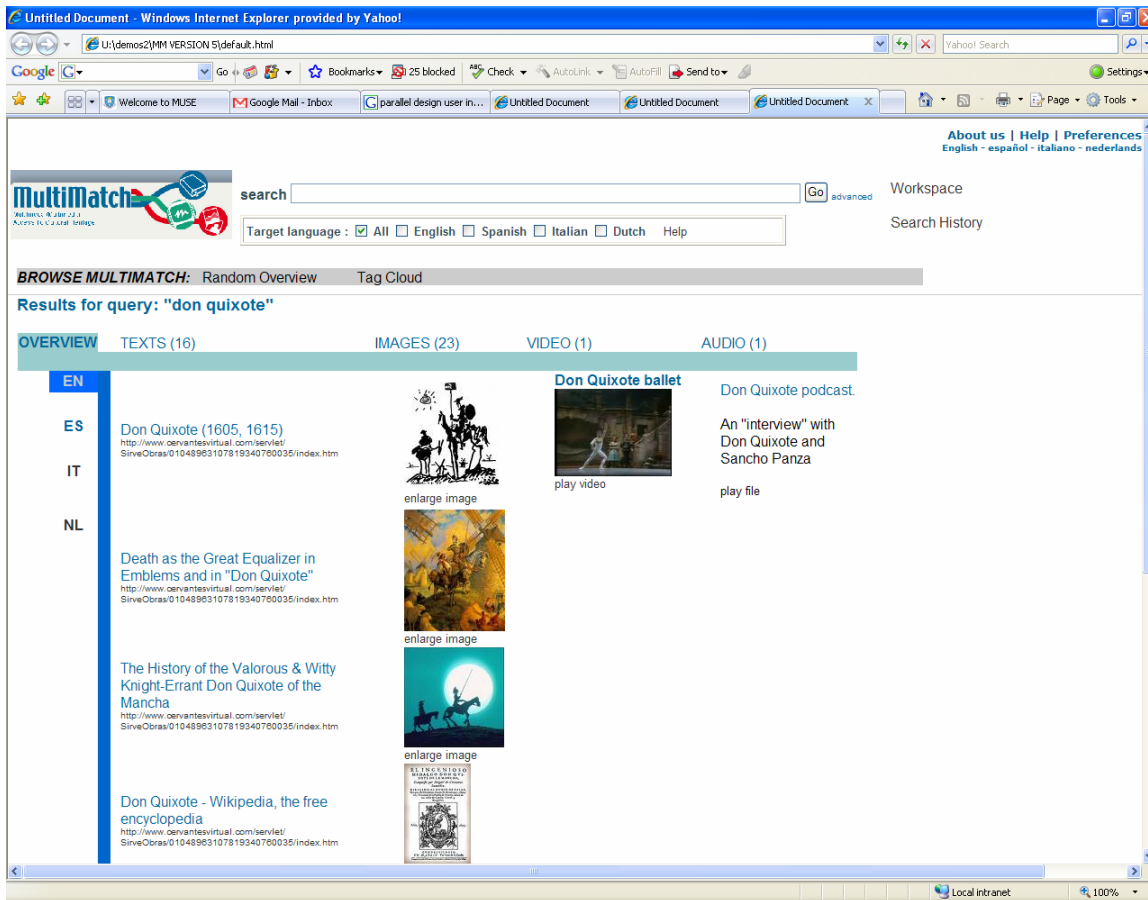
At this point, it was decided to offer the option of selecting both media and language options before launching the search. This image results page corresponds with one of the revised scenarios and shows the various relevance feedback and refinement options. Images are separated by language but all viewable on the same page; while this was viewed as positive because images are language-neutral, it is complicated to split them in this way. The query translation feedback was simplified in order to save space.

Version 4



This version explored the various layouts (columns versus tabs.) There were four different possible designs, based on which factor was seen to be the driving feature (media or language) and which type of presentation occurred for each (tabs-tabs or tabs-columns.) The example illustrated here shows a tabs-columns approach with media as the driving factor and language as the secondary factor.

APPENDIX E: INITIAL INTERFACE FOR PROTOTYPE 1

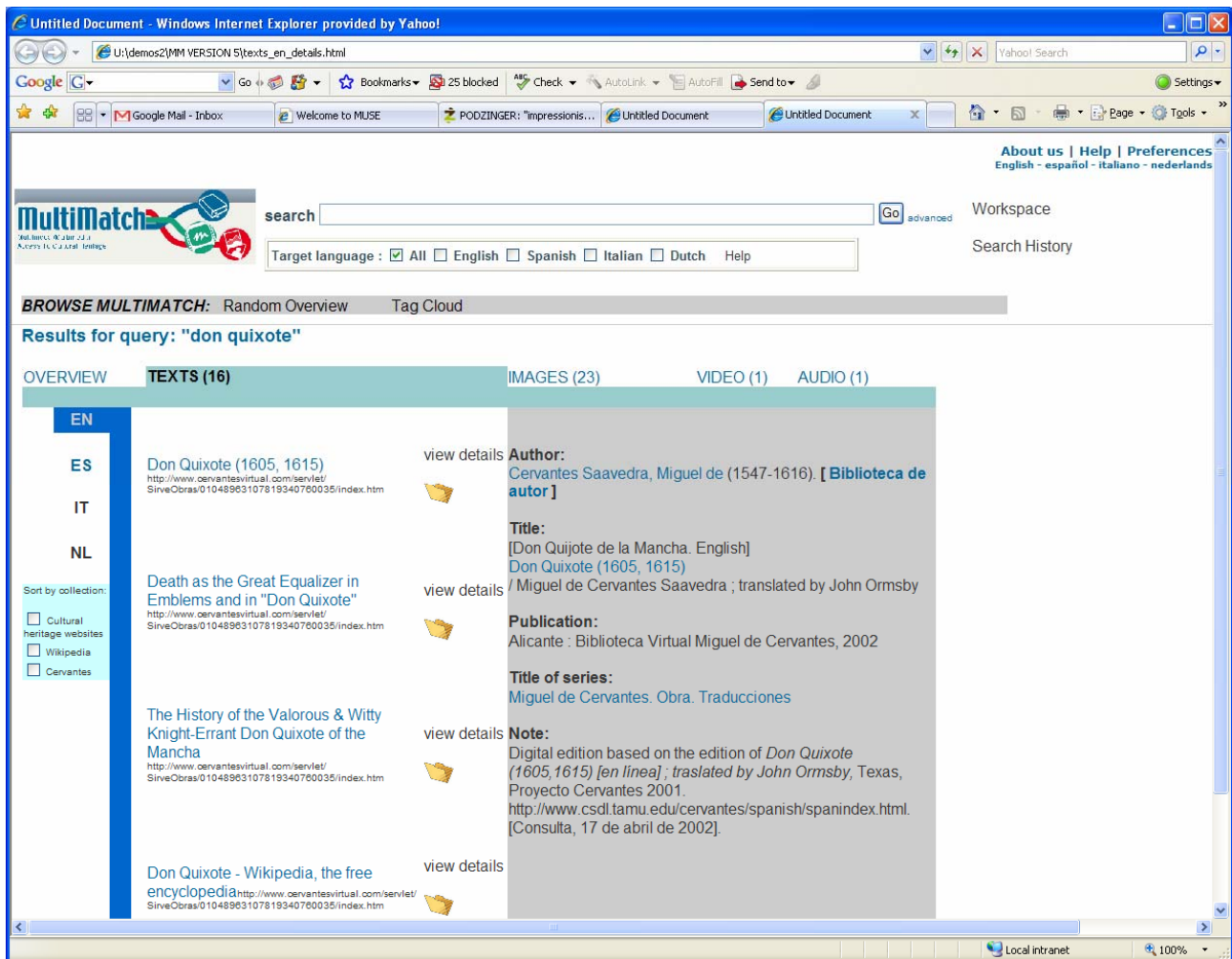


Default results presentation

This will initially appear after the user has submitted a query. The “overview” tab along the top is highlighted in light blue, indicating that this is providing a broad view of the various results retrieved. Down the side are vertical tabs (in dark blue) corresponding to the language results. By clicking on the “ES” tab, the overview would refresh to display the results in Spanish.

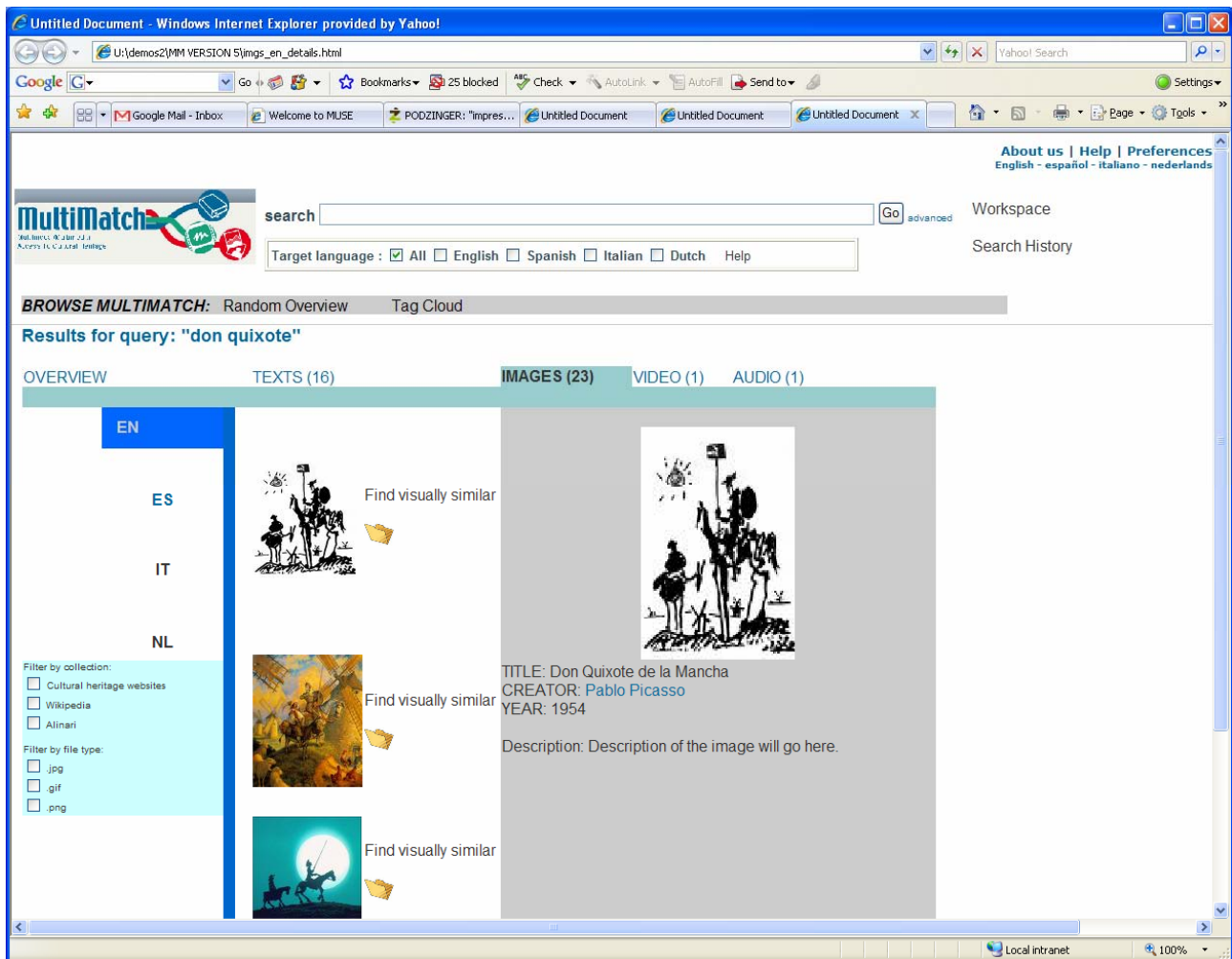
Functionalities available at this stage include navigating to the web links (texts), enlarging the images, and playing the entire video and audio files.

By clicking on any of the other headings at the top (i.e. TEXTS, IMAGES, VIDEO, AUDIO,) the user will then go into the specialised results presentation for that particular media type.



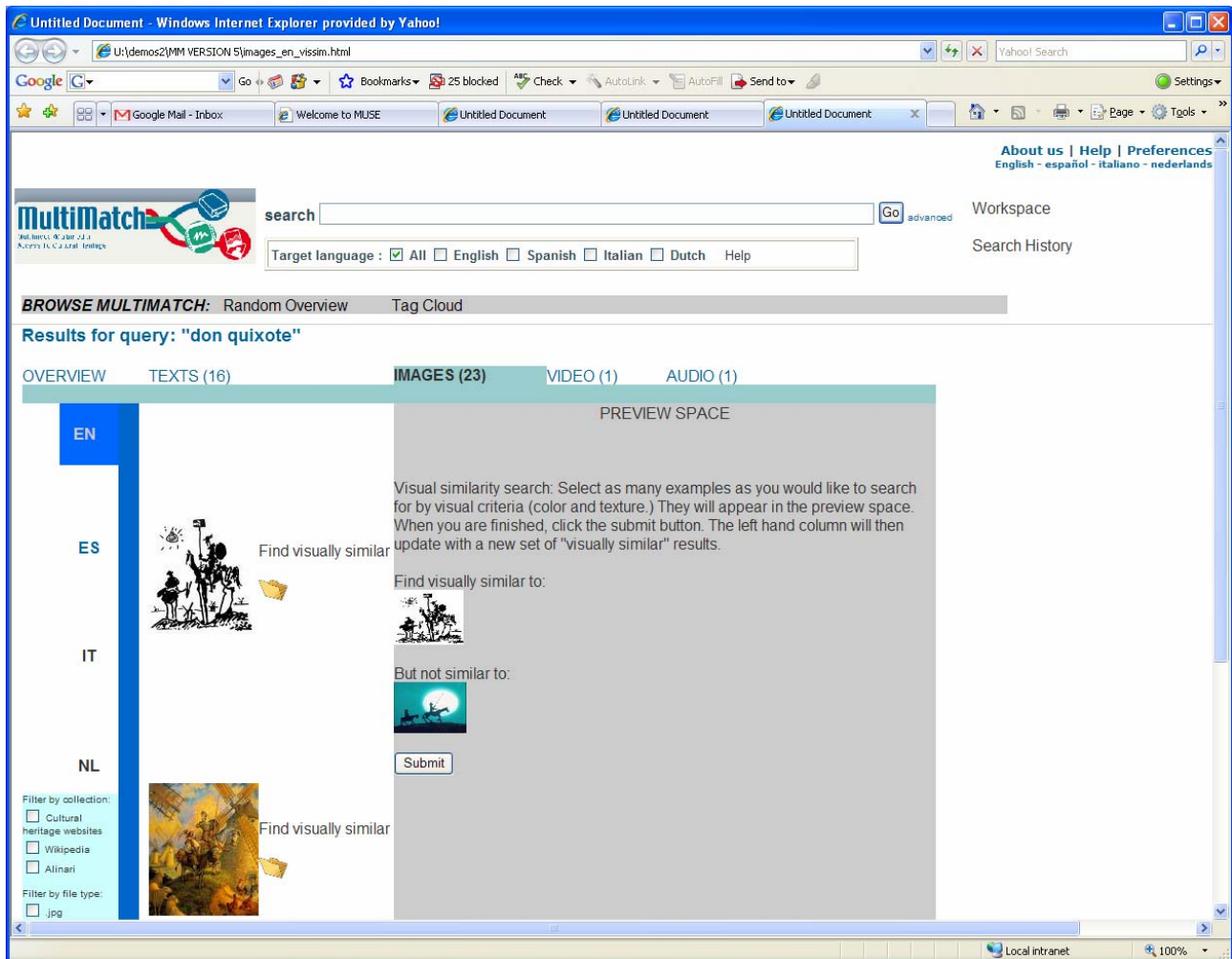
Specialised results presentation – Text

The enhanced functionalities available here are the options of saving a link to the workspace (by clicking on the folder icon) and to filter the results by collection (by manipulating the checkboxes in the turquoise section on the left.) These two features will be consistently available for all media, although the nature of the filtering categories will vary depending on the type. By clicking on “view details” for a given item, further metadata will appear in the gray “summary space.”



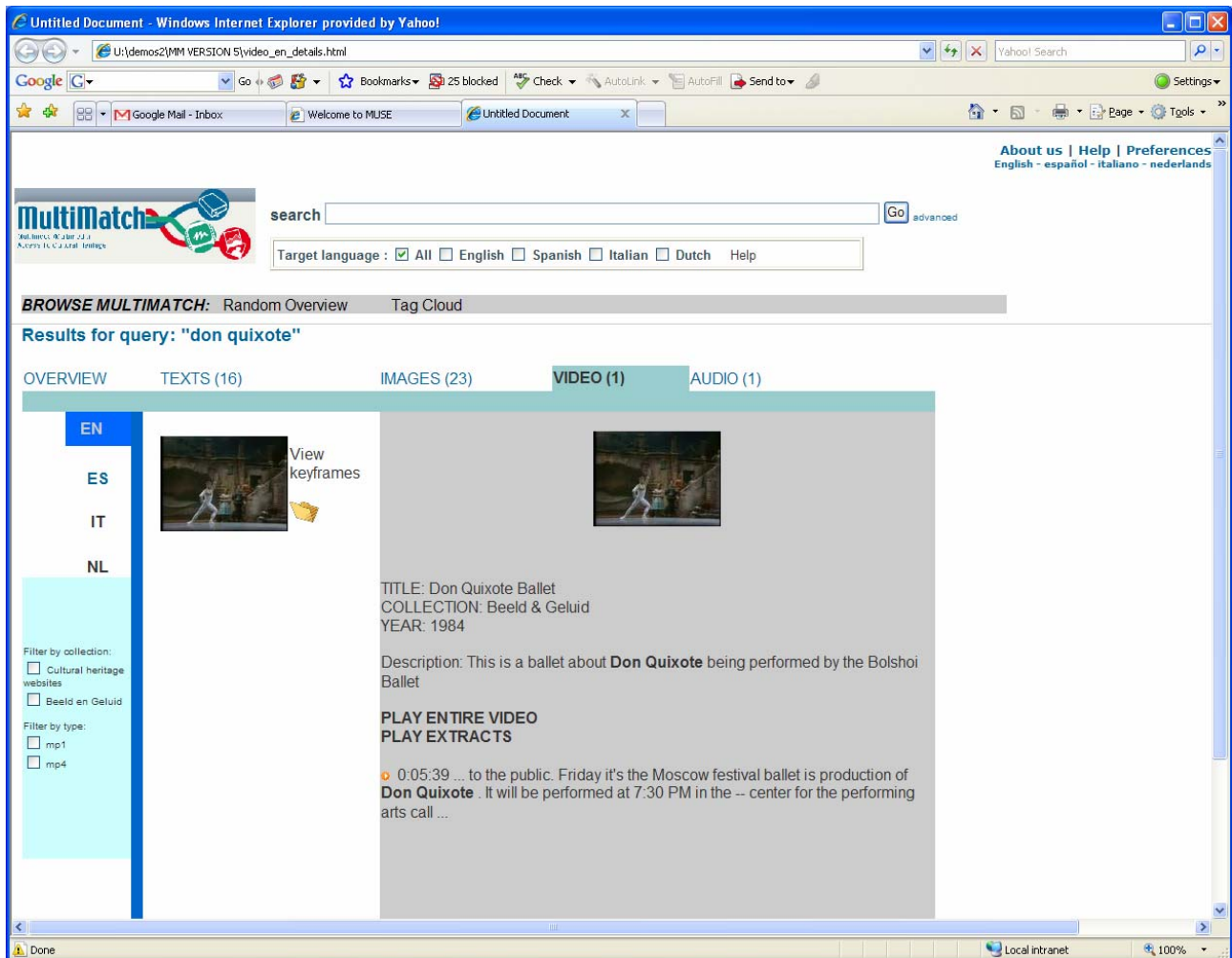
Specialised image results presentation – detailed view

Upon clicking on any of the images, further details will appear in the gray area. All metadata available for the item will be displayed, and where possible, clicking on any of the words (here, “Pablo Picasso”) will then update the images displayed to reflect this new query (i.e. all images created by Picasso would be shown.)



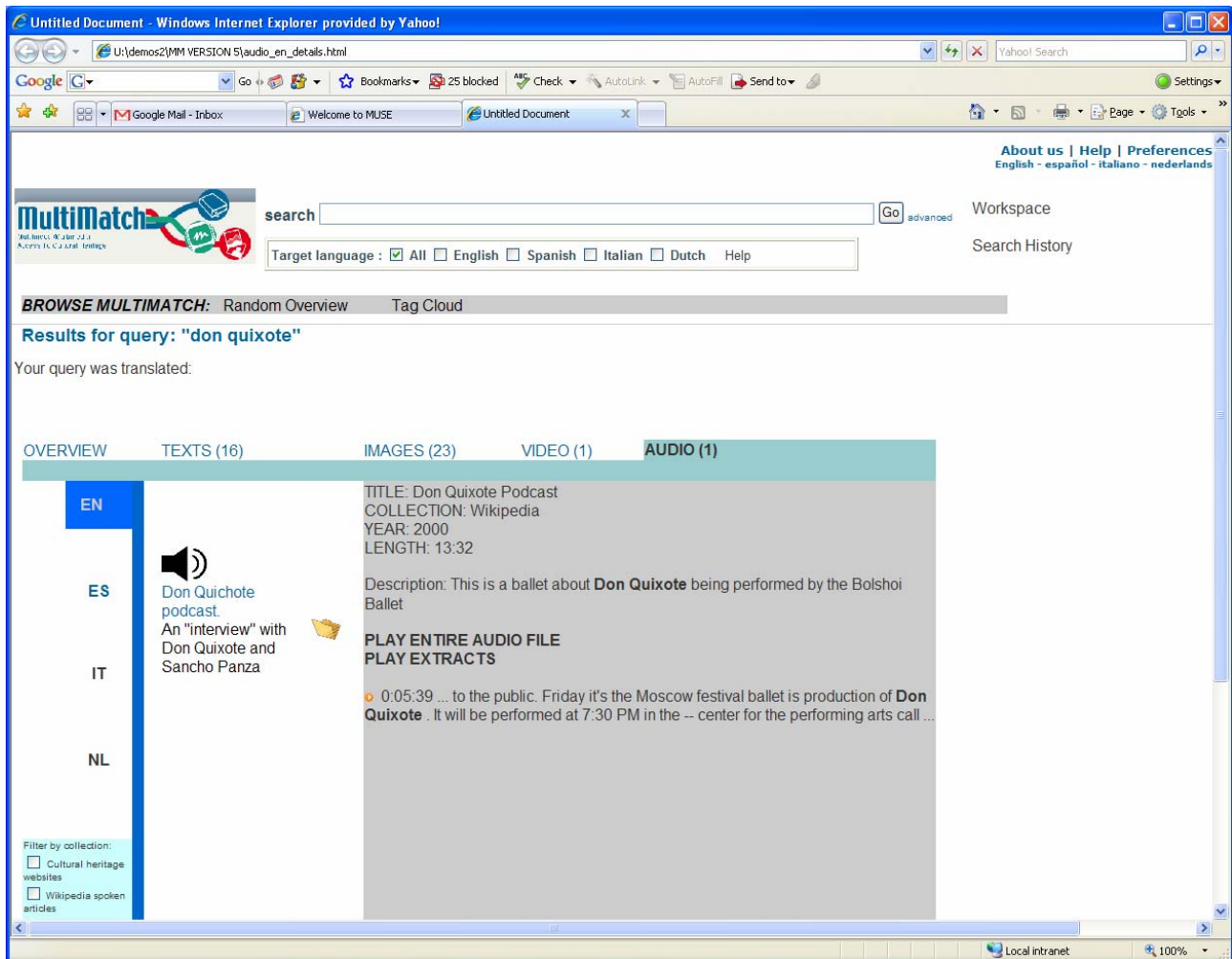
Specialised image results presentation – Visual similarity

By clicking on “find visually similar” next to any of the image thumbnails, the visual similarity search will appear. A brief text will explain how the visual similarity search works and what types of things it will retrieve. Users can specify as many positive and negative examples as they would like before clicking on “submit.”



Specialised video results presentation

Clicking on “view keyframes” will display a storyboard of still image keyframes from the video in the gray section. Clicking on the video itself will show further metadata information (as illustrated above,) and will give the user the possibility of playing the entire video or of playing just an extract (as occurs with PodZinger.)



Untitled Document - Windows Internet Explorer provided by Yahoo!

U:\demos2\MM VERSION 5\audio_en_details.html

Google | search | Go advanced

Target language : All English Spanish Italian Dutch Help


BROWSE MULTIMATCH: Random Overview Tag Cloud

Results for query: "don quixote"

Your query was translated:

OVERVIEW TEXTS (16) IMAGES (23) VIDEO (1) **AUDIO (1)**

EN

ES  Don Quixote podcast. An "interview" with Don Quixote and Sancho Panza

IT

NL

TITLE: Don Quixote Podcast
COLLECTION: Wikipedia
YEAR: 2000
LENGTH: 13:32

Description: This is a ballet about **Don Quixote** being performed by the Bolshoi Ballet

PLAY ENTIRE AUDIO FILE
PLAY EXTRACTS

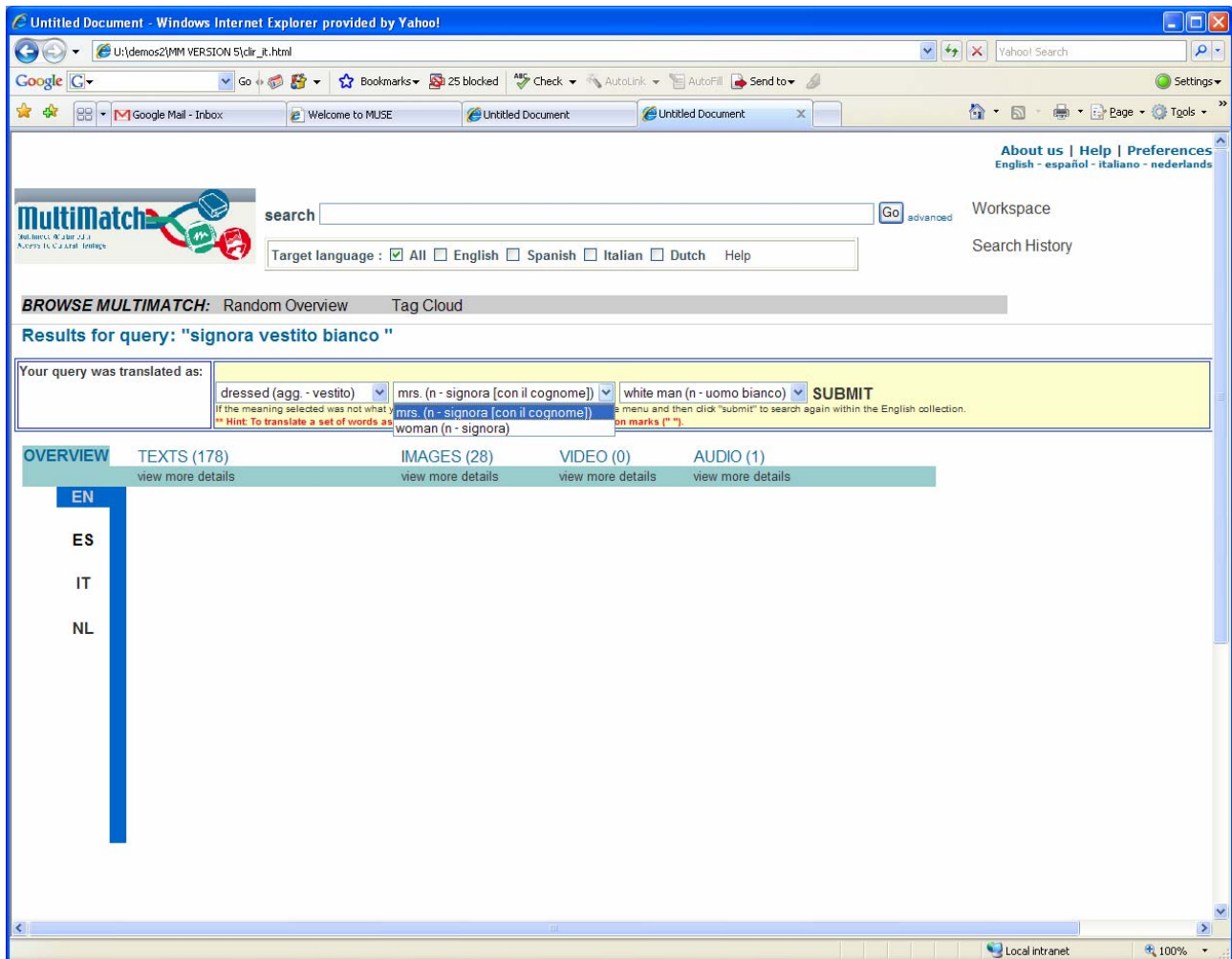
0:05:39 ... to the public. Friday it's the Moscow festival ballet is production of **Don Quixote** . It will be performed at 7:30 PM in the -- center for the performing arts call ...

Filter by collection:
 Cultural heritage websites
 Wikipedia spoken articles

Local intranet 100%

Specialised audio results presentation

Clicking on the loudspeaker icon will once again display metadata information and allow the playing of the entire file or of jumping in at pre-determined points.



Example of query translation feedback

This shows a sample of how query translation information can be displayed and modified. In this case, the query “signora vestito bianco” (meaning woman white dress) was typed in by an Italian speaker. However, no English language results appeared. The yellow box contains information about how each word of the query was translated, with a back-translation into Italian. The query was translated as dressed mrs. white man, due to the fact that each of the Italian words had multiple possible meanings and the machine translation system chose the inappropriate alternatives. Using the drop down menus will display the various alternatives (as shown in the middle of the three boxes above) and allows the user to alter any necessary terms before re-launching the query.

APPENDIX F

USER INTERFACE COMPONENT /WIDGETS LIST

This Appendix describes the initial set of widgets and components identified to be included in the UI implementation for the first prototype. This list must be taken as an initial approach to the UI design and may vary or be extended as implementation progresses.

Headers meaning is the following:

1. **Id:** Is the component identifier (letter+number). The letter can be C if the component can be identified as common for the whole interface or a S if the component is specific for any of the identified search interfaces or functionalities.
2. **Name:** The name of the widget/component.description: A brief description of its functionalities. What does the component do?
3. **Description:** A brief description of functionalities. What does the component do?. If there are more than one, they should be specified in different lines.
4. **Subcomponents:** Components used by this component to be built (if any).
5. **Input:** Data and interactions received by the component to perform a function.
6. **Output:** Data obtained by the component when performing a function. The idea of this field is to write down the expected effect of using the component in terms of the data that will be retrieved from the MM core or from another component.
7. **Relations:** This one is very important. It should describe the component's interaction with any other component within the interface. This information will be used to define the events triggering and listeners' policy to perform communication between components.

The recommendation is to first complete points 1-6 for all identified components, then fill the last one.



Id	Name	Description	Subcomponents	Input	Output	Relations
C-00	Main Entry Point	<p>Allows the user to interact with the MM search engine.</p> <p>The user initially performs a Natural Language (NLQ) to retrieve cultural heritage contents.</p> <p>This entry point allows the user to:</p> <ol style="list-style-type: none"> 1- perform a search 2- explore search results from different views. 3- refine searches via new query or relevance feedback components. 4- log in to the system. 5- customize a user profile (only for previously registered users) 6- perform workspace actions (only for previously registered users) 7- perform search history actions (only for previously registered users) 	<ol style="list-style-type: none"> 1- Search Tab Panel (C-06) 2- Logon Panel (C-07) 3- Workspace Panel (C-08) 4- Search history Panel (C-09) 5- Customization Panel (C-10) 	<p>Any of the actions described in the description field:</p> <ol style="list-style-type: none"> 1- To perform a search 2- To explore search results from different views. 3- To refine searches via new query or relevance feedback components. 4- To log in the system. 5- To customize user profile (only for previously registered users) 6- To perform workspace actions (only for previously registered users) 7- To perform search history actions (only for previously registered users) 	<p>Data provided by each of the functional MM components:</p> <ul style="list-style-type: none"> • Search Components • Log in component • Workspace component • Search history component • Preferences component 	<ol style="list-style-type: none"> 1- Search Tab Panel (C-06) 2- Logon Panel (C-07) 3- Workspace Panel (C-08) 4- Search history Panel (C-09) 5- Customization Panel (C-10)
MAIN PANELS						
S-01	Overview Search Panel	<p>Shows different media search results using separate columns.</p> <p>Its goal is to show the user a brief overview of the various search results retrieved via a natural language query or a relevance feedback action.</p> <p>Note: This panel does not incorporate any functionality permitting complex data manipulation (such as metadata searching).</p>	<ol style="list-style-type: none"> 1- Text Results panel 2- Image Results panel 3- Video Results panel 4- Audio Results panel 5- Language filtering component 6- Collection filtering component 7- File type filtering component 	<ul style="list-style-type: none"> • Search results (text, image, video, audio) retrieved via a natural language query or a relevance feedback action. • The user can browse and explore data using its different subcomponents. 	<p>Data retrieved and/or generated as a result of browsing / exploring actions</p>	<ol style="list-style-type: none"> 1- Main Search Box 2- Text search panel 3- Image search panel 4- Video search panel 5- Audio search panel
S-02	Text Search Panel	<p>Shows image search results.</p> <p>This panel allows complex interactions with retrieved data, such as:</p> <ul style="list-style-type: none"> • Explore metadata. • Enlarge images. 	<ol style="list-style-type: none"> 1- Text Results panel 2- Text preview area 3- Language filtering component 4- Collection filtering component 5- File type filtering component 	<ul style="list-style-type: none"> • Text search results retrieved via a natural language query or a relevance feedback action. • The user can browse and explore data using its different subcomponents. 	<p>Data retrieved and/or generated as a result of browsing / exploring actions</p>	<ol style="list-style-type: none"> 1- Main Search Box



Id	Name	Description	Subcomponents	Input	Output	Relations
S-03	Image Search Panel	Shows image search results. This panel allows complex interactions with data retrieved such as: <ul style="list-style-type: none"> Explore metadata. Enlarge images. Perform image relevance feedback 	<ol style="list-style-type: none"> Image Results panel Image preview space component Language filtering component Collection filtering component File type filtering component 	<ul style="list-style-type: none"> Image search results retrieved via a natural language query or a relevance feedback action. The user can browse and explore data using its different subcomponents. 	Data retrieved and/or generated as a result of browsing / exploring actions	1- Main Search Box
S-04	Video Search Panel	Shows video search results. This panel allows complex interactions with data retrieved such as: <ul style="list-style-type: none"> Explore video metadata. Explore video transcription. Playback video. Explore relevant keyframes. Perform image relevance feedback 	<ol style="list-style-type: none"> Video Results panel Video preview space component Language filtering component Collection filtering component File type filtering component 	<ul style="list-style-type: none"> Video search results retrieved via a natural language query or a relevance feedback action. The user can browse and explore data using its different subcomponents. 	Data retrieved and/or generated as a result of browsing / exploring actions	1- Main Search Box
S-05	Audio Search Panel	Shows audio search results. This panel allows complex interactions with data retrieved such as: <ul style="list-style-type: none"> Explore metadata. Playback audio. 	<ol style="list-style-type: none"> Audio Results panel Audio preview space component Language filtering component Collection filtering component File type filtering component 	<ul style="list-style-type: none"> Audio search results retrieved via a natural language query or a relevance feedback action. The user can browse and explore data using its different subcomponents. 	Data provided by browsing and exploring actions	1- Main Search Box
S-06	Search Tab Panel	Shows to the user all the search views in a tabbed format	<ol style="list-style-type: none"> Overview Search Panel Text Search Panel Image Search Panel Video Search Panel Audio Search Panel 	<ul style="list-style-type: none"> Multimedia search results retrieved via a natural language query or a relevance feedback action. The user can browse and explore data using its different subcomponents. 	Data provided by browsing and exploring actions	1- Main Search Box
S-07	Log-on Panel	Allows the user to register and subsequently log into the MM web site.	<ol style="list-style-type: none"> Login Box Registering Panel (if the user is not previously registered in the system) 	<ul style="list-style-type: none"> alpha-numeric string for username case-sensitive alpha-numeric string for password 	Authentication within MM search system. Authentication flag. User profile (preferences) are retrieved from the user authentication database and used to configure the graphical interface	Potentially any MM visual component which is user-customisable. In the case of the language filter widget (the one where the user specifies both the query and document target languages), the various checkboxes of this widget could be pre-selected based on information from the User-authentication database.



Id	Name	Description	Subcomponents	Input	Output	Relations
S-08	Workspace Panel	Allows the user to interact with his workspace items	1- Workspace tab panel	Workspace data to populate the panel.	Browsing actions to explore the workspace. Some of these actions may involve server calls to retrieve heavy weighted resources such as full size images or streaming audiovisual data.	
S-10	Customization Panel	Allows the user to define his default profile for searches within MultiMatch	1- Options check boxes 2- Options radio buttons	Boolean values indicating customisation parameters (e.g. "Query Languages: Dutch? Yes Spanish? No" This Boolean information would be captured via check-boxes.	The user's profile (i.e. preferences) is stored in the system	
S-11	Registering Panel	Via this panel, users may create personal MM accounts in order to store various details and preferences concerning information presentation options.	1- text boxes to perform registration. With fields such as name, login, password. 2- Customization panel	1 – alpha-numeric string for username 2 – case-sensitive alpha-numeric string for password	Authentication within MM search system. Authentication flag. User authentication data is stored in the system The user's profile (i.e. preferences) is stored in the system	

SEARCH COMPONENTS

C-01	Main Search Box	Perform free text query based search using text, image, video and audio search engines.	1- Text box 2- Button 3- Logo Image (Optional)	1- Natural language query. 2- Target Language box 3- User Preferences 4- Button click	Text, image, video and audio search results. <ul style="list-style-type: none"> If the search action is performed within the overview search panel, then text, image, video and audio results will be retrieved. All search panels will be refreshed. Else, only specific media results will be retrieved and displayed. 	1- Overview search panel 2- Text search panel 3- Image search panel 4- Video search panel 5- Audio search panel Sends the search results to each one.
C-02	Target Language Box	Allows to select the target languages used to perform the search action.	1- Spanish language preference check box 2- English language preference check box 3- Italian language preference check box 4- Dutch language preference check box	1-User selection. Restriction <ul style="list-style-type: none"> For users who are not registered or not logged in, the 'All' check box is selected by default. 	Flags to perform query translation.	1- Main Search box

Id	Name	Description	Subcomponents	Input	Output	Relations
			5- All languages check box	<ul style="list-style-type: none"> For customized users check boxes will be selected by default based on the user's preferences. 		
C-03	Relevance Feedback Component	Tied to a search result (text, image, video, audio). It allows to directly perform text relevance feedback based on the search result textual information and the initial query performed by the user.	None	Text of the search result to which it's tied.	Text, image, video and audio search results. <ul style="list-style-type: none"> If the search action is performed within the overview search panel, then text, image, video and audio results will be retrieved. Else, only specific media results will be retrieved. 	<ol style="list-style-type: none"> Overview search panel Text search panel Image search panel Video search panel Audio search panel <p>Sends the search results to each one.</p>
S-12	Image Relevance Feedback Component	Associated with an image result (still image or video keyframe). It enables image relevance feedback based on the low level features.	None	Image low level features of the search result with which it is tied.	Image or video search results. <ul style="list-style-type: none"> If the action is performed within the image search panel, then image results will be retrieved. If the action is performed within the video search panel, then video search results will be retrieved. 	
RESULTS COMPONENTS						
C-04	Text Result Component	Shows information of a retrieved text resource. Info showed: <ul style="list-style-type: none"> Resource title Resource snippet Resource url File format File size 	<ol style="list-style-type: none"> Text labels for each information field. Links to the resource itself. <ul style="list-style-type: none"> Title label links to the resource. Starred item component . Relevance Feedback component. 	<ol style="list-style-type: none"> The component is automatically populated using the retrieved information. User clicks on browsing items User clicks on the starred component User clicks on relevance feedback component. 	<ol style="list-style-type: none"> Browse using provided links. Storage of results within the workspace. Perform a new search using query+search result information (text relevance feedback) 	
C-05	Text preview space component	Shows more detailed information about the text resource. Info showed: <ul style="list-style-type: none"> Default information showed by text result component Metadata information available. 	<ol style="list-style-type: none"> Text labels for each information field Links to the resource itself. <ul style="list-style-type: none"> Title label links to the resource. 	<ol style="list-style-type: none"> The component is automatically populated using the retrieved information. User clicks on browsing items User clicks on the starred 	<ol style="list-style-type: none"> Browse using links provided. Storage of results within the workspace. Perform a new search using query+search result information (text relevance feedback) Perform metadata based 	

Id	Name	Description	Subcomponents	Input	Output	Relations
			3- Starred item component . 4- Metadata labels allows to perform metadata values based queries.	component. 4- User clicks on relevance feedback component. 5- User clicks on metadata vaules to perform metadata based searches.	searches using metadata-value pairs.	
C-06	Image Result	Shows information of a retrieved image resource. Info showed: <ul style="list-style-type: none"> • Resource title • Resource thumbnail • Image url (for a full view) • Container url (e.g. if the resource is embedded in a web page) • Resource snippet (if available) • File format • File Size 	1- Links to the resource itself should be tied to some of the info fields showed: <ul style="list-style-type: none"> • Resource title links to the full resource. (This link will guide the user to the specialized interface) • Container URL (if available) links to the container web page. 2- Starred item component . Tied to the result to store it in the workspace. 3- Relevance feedback component.	1- The component is automatically populated using the retrieved information. 2- User clicks on browsing items 3- User clicks on the starred component. 4- User clicks on relevance feedback component	1- Browse using provided links. 2- Storage of results within the workspace. 3- Perform a new search using query+search result information (text relevance feedback)	
C-07	Image preview space component	Shows more detailed information about the image resource. Info showed: <ul style="list-style-type: none"> • Default information showed by image result component. • Metadata information available. • Full size image. Incorporates image relevance feedback functionality.	1- Text labels for each information field 2- Full size image view 3- Links to the resource itself. Suggested: <ul style="list-style-type: none"> • Container URL (if available) links to the container web page. 4- Starred item component . 5- Metadata labels allows to perform metadata values based queries. 6- Image relevance feedback component	1- User clicks on relevance feedback component. 2- The component is automatically populated using the retrieved information. 3- User clicks on browsing items 4- User clicks on image relevance feedback component. 5- User clicks on metadata values to perform metadata based searches. 6- User clicks on the starred component.	1- Browse using links provided. 2- Storage of results within the workspace. 3- Perform a new search using query+search result information (text relevance feedback) 4- Perform a new search using image low level features (image relevance feedback) 5- Perform metadata based searches using metadata-value pairs.	
C-08	Video Result	Shows information of a retrieved video resource. Info showed: <ul style="list-style-type: none"> • Resource title • Resource thumbnail (e.g. relevant video keyframe) • Relevant TimeLine/s 	1- Links to the resource itself should be tied to some of the info fields showed: <ul style="list-style-type: none"> • Resource title will link to the full resource (This link will guide the user to the specialized interface) 	1- The component is automatically populated using the retrieved information. 2- User clicks on browsing items 3- User clicks on the starred component.	1- Browse using provided links. 2- Storage of results within the workspace. 3- Perform a new search using query+search result information (text relevance feedback)	

Id	Name	Description	Subcomponents	Input	Output	Relations
		<ul style="list-style-type: none"> video url (for a full view via streaming) Container url (e.g. if the resource is embedded in a web page) File format File Size 	<ul style="list-style-type: none"> Container URL (if available) will link to the container web page. 2- Starred item component . Tied to the result to store it in the workspace 3- Relevance feedback component.	4- User clicks on relevance feedback component		
S-13	Video Preview Space Component	Shows more detailed information about the video resource. Info showed: <ul style="list-style-type: none"> Default information showed by video result component. Matadata information available. Full size image. It also allows to perform image relevance feedback based on video keyframes	1- Text labels for each information field 2- Video playback (embedded player) 3- Set of relevant keyframes extracted from the video. 4- Links to the resource itself. Suggested: <ul style="list-style-type: none"> Container URL (if available) links to the container web page. 5- Starred item component . 6- Metadata labels allows to perform metadata values based queries. 7- Image relevance feedback component.	1. User clicks on relevance feedback component. 2. The component is automatically populated using the retrieved information. 3. User clicks on browsing items 4. User clicks on image relevance feedback component (previously he has to select a keyframe to feed the system) 5. User clicks on metadata vaules to perform metadata based searches. 6. User clicks on the starred component. 7. User plays the video	1- Browse using links provided. 2- Storage of results within the workspace. 3- Perform a new search using query+search result information (text relevance feedback) 4- Perform a new search using image low level features (image relevance feedback) 5- Perform metadata based searches using metadata-value pairs. 6- Play video.	
S-14	Audio Result	Shows information of a retrieved audio resource. Info showed: <ul style="list-style-type: none"> Resource title Relevant TimeLine/s audio url (to listen to via streaming) Container url (e.g. if the resource is embedded in a web page) File format File Size 	1- Links to the resource itself should be tied to some of the info fields showed: <ul style="list-style-type: none"> Resource title will link to the resource (full audio). (This link will guide the user to the specialized interface) Container URL (if available) will link to the container web page. 2- Starred item component . Tied to the result to store it in the workspace 3- Relevance feedback component.	1- The component is automatically populated using the retrieved information. 2- User clicks on browsing items 3- User clicks on the starred component. 4- User clicks on relevance feedback component.	1- Browse using provided links. 2- Storage of results within the workspace. 3- Perform a new search using query+search result information (text relevance feedback)	
S-15	Audio Preview Space Component	Shows more detailed information about the audio resource. Info showed:	1- Text labels for each information field 2- Audio playback (player embedded)	1. The component is automatically populated using the retrieved	1- Browse using links provided. 2- Storage of results within the workspace.	



Id	Name	Description	Subcomponents	Input	Output	Relations
		<ul style="list-style-type: none"> Default information showed by audio result component. Metadata information available. Full size image. <p>It also allows to perform image relevance feedback based on video keyframes</p>	3- Links to the resource itself. Suggested: <ul style="list-style-type: none"> Container URL (if available) links to the container web page. 4- Starred item component . 5- Metadata labels allows to perform metadata values based queries.	information. 2. User clicks on browsing items 3. User clicks on relevance feedback component 4. User clicks on metadata values to perform metadata based searches. 5. User clicks on the starred component. 6. User plays the audio	3- Perform a new search using query+search result information (text relevance feedback) 4- Perform metadata based searches using metadata-value pairs. 5- Play audio.	
C-09	Text Results Panel	A panel that holds the set of text results retrieved	1- A set of text results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous arrow button component (to navigate forward and backward)	1- Directly populated with search results info. 2- The user navigate the list using the next and previous component	1- Access to next and previous list of search results.	
C-10	Image Results Panel	A panel that holds the set of image results retrieved.	1- A set of image results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous component (to navigate forward and backward)	1- Directly populated with search results info. 2- The user navigate the list using the next and previous component	1- Access to next and previous list of search results.	
C-11	Video Results Panel	A panel that displays the set of video results retrieved.	1- A set of video results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous component (to navigate forward and backward)	1- Directly populated with search results info. 2- The user navigate the list using the next and previous component	1- Access to next and previous list of search results.	
C-12	Audio Results Panel	A panel that displays the set of audio results retrieved.	1- A set of audio results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous component (to navigate forward and backward)	1- Directly populated with search results info. 2- The user navigate the list using the next and previous component	1- Access to next and previous list of search results.	
FILTERING COMPONENTS						
C-13	Language filtering Component	Enables filtration of search results by language. By default, the user's language drives which language tab to select by default. The user can click on any of the tabs to turn the view only to the selected language.	<ul style="list-style-type: none"> Spanish language radio button [check box] English language radio button [check box] Italian language radio button [check box] Dutch language radio button [check box] 	Target language box	Inclusion / Exclusion of data depending on language preferences as specified by the selected filters	

Id	Name	Description	Subcomponents	Input	Output	Relations
		Only one language is showed at the same time. Language radio buttons showed are those in which search action have been performed.				
C-15	Collection filtering component	Enables filtration of search results by collection provider (e.g. Alinari digital photo collection)	<ul style="list-style-type: none"> • Alinari check box • Sound and vision check box • Biblioteca virtual check box • Wikipedia check box • All check box 	User selection. By default, the “all” check box will be selected.	Inclusion / Exclusion of data depending on collection provider preferences as specified by the selected filters	
C-16	File type filtering component	Enables filtration of search results by file type	<ul style="list-style-type: none"> • Text file filters (check boxes) [PDF, web ...] • Image file filters (check boxes) [jpg, png, gif, ...] • Video file filters (check boxes) [avi, rm, mov, ...] • Audio file filters (check boxes) [mp3, wav, ...] 	User selection. By default “all” check box will be selected.	Inclusion / Exclusion of data depending on search criteria as specified by the selected filters	
MULTIMEDIA COMPONENTS						
S-16	Video Player	Enables audio audio playback via a web page embedded player object, such as a RealPlayer panel. Allows typical operations with video such as play, stop, fastforward, rewind, and so on.	None	Video streaming	Video playback	
S-17	Audio Player	Enables audio audio playback via a web page embedded player. Allows typical operations with audio such as play, stop, fastforward, rewind, and so on.	None	Audio streaming	Audio playback	
WORKSPACE COMPONENTS						
S-18	Starred item component	Enables the storage and/or removal of a retrieved item in the workspace.	Every dynamically retrieve search result workspace item will be assigned an individual starred item component which performs the operations described in the adjacent column.	The user selects the item to store it in the workspace or, if the item is already in the workspace, to delete it there from.	<ol style="list-style-type: none"> 1- The item is stored in the workspace 2- The item is deleted form the workspace 	
S-19	Workspace text items panel	This component extends the text results panel component, ordering text items by the date they were initially placed in the workspace.	<ol style="list-style-type: none"> 1- A set of text results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous arrow button 	<ul style="list-style-type: none"> • Text search results stored by user in the workspace. • The user can browse and 	Access to next and previous list of search results.	



Id	Name	Description	Subcomponents	Input	Output	Relations
			components (to navigate forward and backward)	explore data using its different subcomponents		
S-20	Workspace image items panel	This component extends the image results panel component, ordering image items by the date they were initially placed in the workspace.	<ol style="list-style-type: none"> 1- A set of text results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous component (to navigate forward and backward) 	<ul style="list-style-type: none"> • Image search results stored by user in the workspace. • The user can browse and explore data using its different subcomponents 	Access to next and previous list of search results.	
S-21	Workspace video items panel	This component extends the video results panel component, ordering video items by the date they were initially placed in the workspace.	<ol style="list-style-type: none"> 1- A set of text results (to be determined by user. By default, 10 items will be displayed per page) 2- Next and previous component (to navigate forward and backward) 	<ul style="list-style-type: none"> • Video search results stored by user in the workspace. • The user can browse and explore data using its different subcomponents 	Access to next and previous list of search results.	
S-22	Workspace audio items panel	This component extends the audio results panel component, ordering audio items by the date they were initially placed in the workspace.	<ol style="list-style-type: none"> 1- A set of text results (to be determined by user. MM default settings will display a maximum of 10 items at a time) 2- Next and previous component (to navigate forward and backward) 	<ul style="list-style-type: none"> • Audio search results stored by user in the workspace. • The user can browse and explore data using its different subcomponents 	Access to next and previous list of search results.	
S-23	Workspace tab panel	This component displays all the workspace items in a tabbed format	<ol style="list-style-type: none"> 1- Workspace text items panel 2- Workspace image items panel 3- Workspace video panel 4- Workspace audio panel 	<ul style="list-style-type: none"> • Multimedia search results which have been manually placed in the workspace by the user. • The user can browse and explore data using its different subcomponents 	Data provided by browsing and exploring actions	