METADATA-BASED SEARCH FOR SPANNING DIGITAL SPACES

by

KONSTANTINOS C. MICHAILIDIS

B.Eng., Applied Informatics and Multimedia, Technological Educational Institute of Crete, 2013

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

DEPARTMENT OF INFORMATICS
ENGINEERING

SCHOOL OF APPLIED TECHNOLOGY

TECHNOLOGICAL EDUCATIONAL INSTITUTE OF CRETE

2015

Approved by:

Major Professor
Demosthenes Akoumianakis

Abstract

Metadata refers to data about digital resources such as photos, documents, and articles. In web-based systems, the importance of metadata is increasing as such data, not only qualify a range of user activities on resources such as photo, URLs and blog entries, but also facilitate alternative indexing, classification and exploratory search. The proposed thesis aims to investigate the use of metadata, particularly tags, to allow users to compile collections of digital resources retained by different social media sites.

One key advantage of tags is that they introduce an emergent shared taxonomy of keywords or classifiers, which aid users in browsing by providing meta-data about the contents of uploaded materials. Based on the notion of tags, collaborative tagging systems allow users to share resources in the web and to annotate them with freely chosen tags. Nonetheless, in most cases, tags, referent resources and the users' digital representations become intertwined and embedded in a single (bounded) digital space. As a result tags cannot transcend boundaries of digital spaces, and most importantly, they do not afford cross-space referencing of resources (i.e., using a single tag to qualify different resources in different services). On the other hand, the capability for cross-space referencing of digital resources fosters synergistic appropriation by virtual groups and emergent compilations of digital material distributed across digital spaces.

One approach to address this problem is through the use of public APIs to establish a form of information connectivity that is emergent and spans the boundaries inscribed into technologies. To this end, the proposed work sets out to explore how such connectivity can be facilitated for resources distributed across Flickr, YouTube, the knowledge repository Freebase and domain-specific repositories. In this manner, it seeks to examine collective and collaborative tagging as a tactic for multi-party collaboration and virtual group work in distributed settings.

Table of Contents

List of Figures

Acknowledgements

This thesis has been developed and written using resources of the department of Informatics Engineering of the Technological Educational Institute of Crete (TEI – Crete) and its research unit iSTLab (Interactive Software Technologies & System Engineering Laboratory). Additionally, this research has been co-financed by the EU European Social Fund (ESF) and Greek national funds through the Operational Program "Competitiveness & Entrepreneurship" (EPAN II) of the National Strategic Reference Framework - Research Funding Program: "COOPERATION 2011 - Partnerships between businesses and research bodies in specific research and technological sectors".

Dedication

I would like to thank everyone who their different type of help contributed to get me at this point. First of all, I would like to thank from my heart my family that without their support I would not be at this point now. Afterwards, Professor Demosthenes Akoumianakis, my scientific responsible, who was an ideal supervisor and a true mentor, motivating, encouraging, reading, reflecting and most of all gave his guidance and patience throughout the entire process. Also, Professor Nikolas Vidakis, as well with the whole iSTLab team for their support and excellent cooperation.

Chapter 1 - Introduction

During the age we traverse information needs are becoming bigger in every moment that elapses. These needs for information are widespread reached with the use of the Internet and technologies that enable search mechanisms to allow users find the desirable result in the best way. Web 2.0 applications include web-based software and services that enable individuals to create, share, communicate and collaborate on the web, regardless of geographical, temporal or technological skill constraints [1].

One basic issue that Web 2.0 elaborates is the notion of metadata. Metadata refers to data about digital resources such as photos, documents, and articles. In web-based systems, the importance of metadata is increasing as such data qualify a range of user activities on resources such as photos, URLs and blog entries. In this particular context the operation of adding metadata has been dubbed as social annotations or tagging. Tagging may be conceived as just one element of the new linguistic vocabulary introduced by web 2.0, with catalytic consequences on searching, adding value to and making sense of digital content as well as more effective knowledge management. Moreover, one emerging issue is the way tagging mechanisms affect and/or improve information retrieval and searching the web.

The crucial role of search engines in locating meaningful information on the web is justified from the fact that currently, three out of the top-five sites on the web, according to Alexa [2], are search engines. The amount of information on the Web is growing rapidly, as well as the number of new users inexperienced in the art of exploring the web. Over the past decade, web search engines have grown in their sophistication and usefulness and consequently their users have increasingly come to trust and rely upon them. Yet, web search is not perfect and finding useful information on the Internet is still challenging.

However, despite their popularity, such services do not always succeed in retrieving the right results for their users. Sometimes, queries suffer from intrinsic features of natural languages such as ambiguity and synonymy of words. According to [3], web search engines have to overcome many problems to succeed in providing searchers with the right information. Thus, a major issue they have to deal with is the fact that average searchers do not experience a standard behavior during their information seeking process.

1.1 Research goals and questions

Web search that utilizes social tagging data suffers from an extreme example of the vocabulary mismatch problem encountered in traditional information retrieval. This is due to the personalized, unrestricted vocabulary that users choose to describe and tag each resource. Previous research has proposed the utilization of query expansion to deal with search in this rather complicated space. However, non-personalized approaches based on relevance feedback and personalized approaches based on co-occurrence statistics only showed limited improvements [4].

One particular movement that tries to provide answers to such queries is the Semantic Web. The Semantic Web aims at adding logic to the World Wide Web. The idea behind this is that the Web becomes better readable for machines. This way, machines would be able to get a better understanding of how pages are related to each other and where they have to look for certain information [5]. It seems somewhat intuitive that if we are to handle a specific type of information request optimally, the search engine needs to be fine-tuned. This may imply many changes, ranging from changing the search space to the designing of an appropriate interface [6]. The main research question in this work is:

How does metadata-based Search perform compared to existing conventional search methods?

To answer this question the following sub questions need to be answered.

1. *What functionality should a metadata-based Search engine have?*
2. *How to design and build a metadata-based Search engine?*

1.2 Structure

This master thesis contains six chapters.

<u>Chapter 2</u> reviews the related theoretical background and prevailing perspectives firstly of tagging mechanism and then assesses information retrieval concerns of four searching practices that are more related to this work. The chapter concludes with a brief review of web mashups.

Chapter 3 presents and elaborates on the theoretical scaffold of this master thesis by seeking to untangle intrinsic properties of the concepts of boundary artifacts and boundary spanning tactics. This is structured by firstly introducing the notion of boundaries in computer-mediated settings, then review the bibliography on what boundary artifacts are and at last how search mechanisms impress boundary spanning tactic.

Chapter 4 deals with the implementation of our system according to the theoretical guidelines described in the previous chapters. First, comes a brief overview of the development platforms and the software technologies used for this implementation. Then it is quoted a brief description of the digital spaces that taking place in the current implementation. The chapter closes with a description of the system`s architecture.

Chapter 5 firstly depicts the implementation by presenting the basic user interface and explaining the basic functionality features of this search mechanism. It continues with two use-case scenarios took place in real settings and expanded the functionality features of our system by further implementation for exploiting domain-specific information.

Chapter 6 consolidates the findings of the present work and provides answers to the research questions.

Chapter 2 - Background and related works

In order to motivate the present research, this chapter will first review related works and prevailing perspectives on the use of metadata in search applications and services. Although such metadata may take several forms, our primary concern is on user tags and tagging mechanisms. Following this, we will briefly review the implications of tags on searching and the variety of practices that emerge as a result. Finally, the last section will be devoted to the concept of “mashups”, which is a highly relevant to the present work. The above will set the focus on certain limitations that drive the current effort.

2.1 – Tagging mechanisms

In recent years, social and collaborative tagging has attracted substantial research attention. There have been works investigating why people tag, what tags mean and users’ motivations and how social annotations, more general, can affect positively web search procedures and enable better results. Gupta et al. [7] claim that users tag for various reasons, including future content retrieval, Contribution and Sharing, Attract Attention, Play and Compete, Self-Presentation (Self Referential Tags), Opinion Expression, Task Organization, Social Signalling, Technology Ease and Money. Following this, the authors [7] classify tags by kind or genre as Content-based, Context-based, Attribute tags, Ownership tags, Subjective tags, Organizational tags, Purpose tags, Factual tags, Personal tags, Self-referential tags and Tag bundles.

Irrespective of their purpose or genre, tags are inherently related to categorizing and classifying resources. According to [8] “categorization divides the world of experience into groups or categories whose members share some perceptible similarity within a given context”, where category composition depends on the context and on the user of the organization; “classification involves the orderly and systematic assignment of each entity to one and only one class within a system of mutually exclusive and non-overlapping classes”. Chiara et al. [9] stated after Jacob [8] that the process of categorization is generally perceived as being less precise than classification: the placement of an item within a classification structure indicates precise global information about that item, whereas placement of an item within a categorization structure may represent partial information about the item, which is to be interpreted locally, or within a given context.

One other issue is that tags associated with a resource generally are in a random order without any importance or relevance information. A unique aspect of tagging systems is the freedom that users have in choosing the vocabulary used to tag objects: any free-form keyword is allowed as a tag. This may limit the effectiveness of these tags in search and other applications. Moreover, the lack of relevance information in the tag list has significantly limited the application of tags. For example, in Flickr tag-based search service currently cannot provide the option of ranking the tagged images according to relevance level to the query. However, relevance ranking is important for image search, and all of the popular image search engines, like Google and Live, rank search results by relevance [10].

Social tagging find many applications [7]: It is particularly useful for both personalized and enterprise search – the latter application is called enterprise bookmarking. The reason is that social tags act as multi-faceted descriptors of content and, thus, improve its findability. Especially in the context of enterprise bookmarking, social tags affect the way people and organizations share information through intranets and document management systems This offers improved decision-making support both to groups of decision makers within an enterprise and across the enterprise, allowing them to use tags to enhance the findability of content without waiting the (usually slow) process of its formal categorization and cataloging. Other applications in the business domain include using tags as source of information when ranking web sites, since the number of social tags assigned to a site can measure its popularity. Moreover, social tags provide faster and more thorough indexing of web sites, because they allow the discovery of sites that have not been yet linked by others or crawled by search engines. Finally, social tags can support the classification of fast changing information, such as blog entries, whereas they can become valuable for the task of social interest discovery, i.e., finding users' interests and their communities [11].

2.2 - Searching practices

There is extensive work in the literature, which tries to analyze web searching concerns and user web search behavior from several different perspectives. Streams of research try to catch how search engines' architecture work and rank the result to fulfill user's request.

Although this concern is crucial for information retrieval research, it is not the focus of this work. There is plenty of the literature that examines user behavior and interactions between technology and other peer users.

When search technologies are used for professional search (i.e. search for a professional reason or aim) there are a number of characteristics which differentiate them from web search: lengthy search sessions which may be suspended and resumed, different notions of relevance, different sources searched separately, and the use of specific domain knowledge [6]. However, the overwhelming majority of the literature agrees that web search results lack of accuracy in combination with the user's request query for information. In this work we categorize search practices in four discreet subcategories: social search, semantic search, thematic and enterprise search.

2.2.1 - Social search

Social search refers to a search method in which people collaborate to find the information they are looking for. Models have been created to understand the process of Social Search and to get insight in the anatomy of social search engines [12]. In this context, Social Search is defined by Evans and Chi as "information seeking and sense-making habits that make use of a range of possible social interactions: including searches that utilize social and expertise networks or that may be done in shared social workspaces. This notion certainly encompasses collaborative co-located search, as well as remote and asynchronous collaborative and collective search" [12].

According to another work of the same authors [13] any search that contains social interactions is social search and especially in the field of web 2.0, where share social workspaces exist, involving social data mining. They conclude with a definition, claiming that: "social search is an umbrella term used to describe search acts that make use of social interactions with others. These interactions may be explicit or implicit, co-located or remote, synchronous or asynchronous [13]."

McDonnel and Shiri [14] define that the term social search refers to the use of social media to aid finding information on the internet. Typical examples include using social bookmarks (e.g., tags) to influence the ranking of search engine results, or searching a social network to help people find experts on a specific topic.

2.2.2 – Semantic search

Finding the right piece of information on the Web is often a nightmare. In searching the Web for specific information, users get lost in huge amounts of irrelevant material and may often miss the relevant matter. According to Guha et al., [15] semantic search attempts to augment and improve traditional search results by using data from the Semantic Web. The availability of large amounts of structured, machine understandable information about a wide range of objects on the Semantic Web offers some opportunities on traditional search.

In a conventional programming environment the developers will write queries based on knowledge of the search criteria, data structure (e.g. relational database) and the query language. A truly optimized semantic search however requires a search engine that can write itself queries to be fed back into the query engine. This means that a query language alone will not benefit a search system but rather is highly dependent upon the base data structure it works upon and the top-level system interacting with it [16].

Semantically-aware search engines, and in particular those that use ontologies as enabling technologies, have gained considerable interest in the last few years but the actual fulfillment of the vision is still unclear. While ontology-based semantic search systems have been shown to perform well in organizational semantic intranets [17], there have not yet been convincing attempts at applying semantic search to the web as a whole [18] and almost always semantic systems are restricted to a limited set of domains or they use just one specific domain ontology at a time.

However, part of the literature focuses on semantic annotations in web search. According to Berlanga et al., [19] Semantic annotation is the process of linking the meaning of unstructured data to concepts that are unambiguously described in a knowledge resource. Tagging mechanisms allow activity in this direction by annotating resources with meaningful tags.

2.2.3 – Thematic and enterprise search

Thematic search refers to search systems, that their results consist of concrete set of results e.g., geographical data, photos, videos, etc. Caramia et al. [20] in their work claim that the focus of information retrieval research is switching from quantity (maintaining and indexing large databases of web pages and quickly selecting pages matching some criteria) to quality (identifying pages with a higher quality of accuracy for the user). Such a trend is motivated by the natural evolution of Internet users, who are now more selective in their choice of search tools and may be willing to pay the price of providing extra feedback to the system and wait more time for their queries to be better matched.

In addition to, enterprise search allows users in an enterprise to retrieve desired information through a simple search interface [21]. While Internet search engines have been highly successful in content retrieval, enterprise search remains difficult. For instance, the intranet search engine deployed by IBM before 2011 returned no relevant results in its top 50 results for about 66% of user queries [22]. Due to the challenges and significant benefits of building a reliable enterprise search engine, enterprise search has been a topic of attention among research community. According to Hawking [23] “the ultimate goal of an enterprise information retrieval system is to respond to a request by searching all the resources which may possibly contain useful answer (and which the searcher is entitled to see) and to present search result in a form or order which is of maximal utility to the searcher”.

2.3 – Web Mashups

This section reviews a highly related concept of web 2.0, the mashup concept. The term Web 2.0 is commonly associated with web applications that facilitate interactive information sharing, interoperability, user-centered designs, and collaborations on the World Wide Web (WWW). People can share their thoughts, interests, photos, video clips, and others through social network applications like Facebook, MySpace, Flickr, Instagram, and Twitter. They publish their views in blogs and get instant responses and feedback from the e-communities.

Mashup history begins with DJ mashups of songs and with web 1.0 portals. The oldest mashup on Programmable Web (http://www.programmableweb.com) was added in 2005. The mashup ecosystem [24] may be seen as linking mashups and web Application Program Interfaces (APIs) (see Figure 1). An API is a protocol that can be used to communicate with and access data from other software components [25]. [26] conceive it as configuration of service providers, mashup authors, and users without any central authority. Web mashup applications are an interesting genre of interactive web applications that has become common recently. Web mashup applications are applications that integrate various data, presentations, and functionalities from two or more sources through APIs. It not only combines the data but also the process or view from several websites to provide information that could not be easily obtained by manually browsing the websites separately [27, 28].

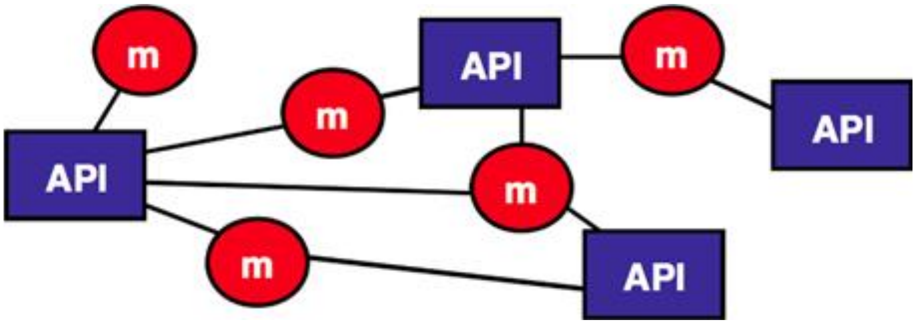

Figure 1: API - Mashup network [24]

Making mashups involve five processes: data retrieval, source modelling, data cleaning, data integration and data display [29]. They aim to combine these sources to create useful new applications or services. Content and presentation elements typically come in the form of RSS or Atom feeds, various XML formats, or as HTML, Shock-Wave Flash (SWF), or other graphical elements. Publicly available APIs (in JavaScript, for example) typically provide application functionality [30].

There are several mashup tools developed to help end-users who lack programming skills in making mashups. Vegemite [28] is an extension of a web automation tool with a spreadsheet-like environment that uses direct manipulation and programming-by-demonstration techniques to automatically populate tables with information collected from various web sites such as Yahoo Maps locations. Potluck [31] is a web user interface that mainly merges information from different sources by indicating fields and subsets of data and lets users explore and identify subsets of data by faceted browsing. Marmite [32] is a plug-in for the Firefox web browser that operates on online information with four ways: sources, filters, processors and skins. “Sources” retrieve data from a web service, perhaps forming a query to a database on behalf of the user. Source operators include a wizard that guides a user through the process of scraping information off a web page. A “processor” augments or transforms data that flows through it. A “filter” removes elements that do not match some user-specified criteria. “Sinks” are final destinations for the data, such as visualizations. MashMaker [27] is a web-based tool that allows end users to create web mashups by browsing around. MashMaker suggests functions (mapping, folding and filtering) that a user might apply to their data, based on the functions that other users have applied. Mashroom [33] is an end-user-oriented programming environment that works with nested tables as the data structure and defines a set of visual mashup operators to offer a spreadsheet-like programming experience. The case study presented makes combination of movie-centered information from “googlemoviesDS” and “IMDB”. SituMash [34] is a system for developing situational mashups according to user situation, defined as the configuration of accessible widgets depending on user location and schedule, supporting the automatic composition of widgets in response to user’s situation changes.

2.3.1 – Search Mashups

Search Mashups are also likely to improve over time, not only because of Google-like search mechanisms but also because of an emerging trend toward online communities of mashup taggers and bloggers, following the style of the social Web [30]. In the context of web search, mashups could be described as the efforts that are targeted towards the provision of extended functionality to the traditional paradigm of a search engine through the combination of data deriving from multiple sources. Such functionality is often realized as innovative query construction/refinement services aiming at enhancing users’ search experience [3].

Related works such as Braga et al. [35] proposed a visual service mashup language for graphically composing and automatically executing queries over search services. The proposed language lets users declaratively specify a query and mash up registered services in a drag-and-drop fashion to compose that query. The authors proposed a physical service access plan for the Web service composition's execution needs, such as generating a schedule of series or parallel service invocations, orchestrating such invocations, and joining data from different services into a ranked output. Rosenberg et al. [36] proposed the Bite language based on a lightweight process composition model for both Web data-driven applications and Web workflow composition. Bite combines SOA process composition principles with REST architectural requirements and workflow functionalities. It lets users implement RESTful service composition and interactive workflows. Riabov et al. [37] introduced MARIO (Mashup Automation with Runtime Orchestration and Invocation), a tool that simplifies data mashup composition that uses a tag-based search abstraction for automatic goal-driven composition of flows. MARIO allows users to explore potentially composable data mashups and preview composition results. The presented evaluation scenario focuses on searching Yahoo Answers and Yahoo News.

2.4 – Summary

To sum up, the available literature indicates that web search is a field which is intensively researched. Topics of current interest include efforts to improve the accuracy of search results as well as new techniques for specifying search conditions and formulating search queries. Both these issues may be addressed by semantic annotations and especially with tagging mechanisms. To this effect, there are prominent design issues pending attention.

Firstly, it is of paramount importance for users to be able to specify as accurately as possible what it is that one is searching for?” In the majority of cases, recent works offer solutions by designating the type of resources, such as images or videos. Nonetheless, there are no ways to anchor search to special type of digital objects such as calendar events, specific video frames, etc.

Secondly, it is equally crucial for users to be able to designate the scope of search by declaring “where am I searching?” Such an designation is typically addressed either by assuming a single search space such as the World Wide Web or by “bounding” the searching process to specific digital spaces, where the desirable information may exists. In subsequent sections of this thesis, we address explicitly the challenges involved in scoping the search process to bounded services and advance a proposal which rests on the construct of boundary.

Finally, the search process can be substantially improved when the information sought may be anchored by domain specific categories such as tags, metatags or other semantic anchors. This is an active area of research that seeks to extend searching beyond mere keywords or the use of native tags qualify resources such as images, videos and files resident in a digital space. The rationale rests on the fact that these forms of anchoring information are not adequate to retrieve the desirable search results that are spread across several digital spaces. One possible solution is to explore the construct of metatags that cross boundaries of digital services, thus enabling a kind of boundary spanning that is emergent and dynamic since it blends digital resources with linguistic intentions of users.

Chapter 3 - Approach and Theoretical scaffolding

This chapter attempts to build up the theoretical scaffold of this thesis by elaborating on the construct of boundaries. The objective is to advance an alternative conception of boundaries that extends beyond the static demarcations of time and space typically found in the Information Systems literature. To this end, our current thinking is motivated by recent claims of many researchers who conclude that although the concept of boundaries has not been treated in the literature with the thoroughness it deserves, it is gaining attention and is likely to establish a key design concept of information systems [38]. Manheim et al. [39], posit that people often adapt to working across boundaries and learn to do so effectively and efficiently. In terms of system design, this is witnessed in recent works towards e-infrastructures and inter-organizational information systems where it is convincingly argued that established management information systems (MIS) lack the capacity to cross over organizational borders [40, 41].

3.1 - Boundary characterization and distributed work

Boundaries are a central idea in social sciences and important for the understanding of relationships within and between social systems and structures. They are also experienced in everyday life and in a multitude of forms. A typical example of boundaries encountered in everyday life is the fence that separates pieces of land or the border between two neighboring geographical divisions. In both cases the boundary or border is perceived in a specific way, although it may be defined by different material or manufacturing means. For example, a fence that anchors access limits or property ownership can be manufactured by either a low wire mesh or a high wall of cement. However, each case signifies different capabilities or skills with respect to crossing the boundary.

In a similar vein, boundaries are meaningful in organization sciences. To function effectively, organizations need to have "balance between differentiation and integration" of organizational and departmental systems that characterize them. This highlights the need for special care and attention to the boundaries between structural components. For example, sales departments have a specific language and objectives that necessarily differ from those of the engineers in production line or managers. At the same time, these different organizational departments must be aligned in such a degree as to achieve the smooth operation of the business or organization. The task of aligning organizational systems is often assigned to key persons in such places, where in order to improve their performance in the group need to coordinate and facilitate communication and relation management with other members and other groups involved [39]. In a sense, these people undertake to span mental boundaries that separate groups of people who either are interconnected and/or supplement each other. Such boundaries differ from the classical definition of the term that is found in other disciplines such geography or in everyday life.

In the Information Technology (IT) sector, boundaries are also important in designating computing capacity and work procedures but also tactics for distributing or sharing digital artifacts. In the former case where boundaries are intended to qualify computing capacity and work procedures, they offer a means for anchoring independent islets of information separated by technical means such as type of operating system, application genres or data models. When boundaries designate tactics for distributing or sharing data, they invoke considerations on quality attributes that determine how data are mobilized and transferred from one system to another. Although both notion of boundary are quite prominent in the Information Systems literature, results are far from being filly consolidated. In most cases, boundaries are conceived as static demarcations of place, time or organizational settings [42-45], while it is also common for empirical studies to suggest (implicitly) that boundaries are always problematic for individuals and teams [39]. For instance, it often occurs either the same boundary to have different effects with the lapse of time, or some kinds of boundaries to cause problems in a group, but not to another [39]. Although this view is beginning to be challenged, a fact that cannot be disputed is that in the areas of information systems boundaries and boundary spanning remain open issues [38].

3.2 – The term "boundary" in computer-mediated settings

To start with, we should recall that computers provide a category of tool that mediates the execution of human routines. Such routines may be invoked by individuals or group of peers who may be co-located or dispersed across geographic locations. In the latter case (i.e., collaborative work amongst distributed peers), boundaries demarcate the different locations, local settings such as time zones and social contexts in which collaborators find themselves in. (ref).

Through computer mediation some boundaries may be alleviated, others can be relocated or re-aligned. For instance, the advent of digital networks and the Internet in many cases alleviate (or lessen the importance of) proximity boundaries. On the other hand, time remains an issue in need of negotiation between partners involved, but such negotiation can be easily handled with today's technical offerings. Although these boundaries are widely acknowledged in the Information Systems scholarship, they are not the only ones or the most prominent type catalyzing the distributed organization of work. The emergence of mobile devices and the plethora of internet services for communicating, networking and collaborating create new demands for sharing resources and mobilizing information across physical and digital boundaries. Consequently, there is a compelling need to reorient and conceptualize boundaries as dynamic qualifiers of context rather than static demarcations of time and geographic space. By context here, we refer not only to the physical settings in which virtual work is executed but also the digital spaces implicated when information is fetched from a certain host and transferred through the network so as to become intelligible in another. The unique features qualifying such mobility and information transfer raise a multitude of research challenges that are increasingly acknowledged by researchers.

One prominent challenge is coined by the term boundary spanning capacity which is used to designate the technologies' embedded capacity to cross the boundaries of digital spaces constituted by 'bounded' systems and services. Although today the basic directions in the areas of Information Technology and collaborative technologies recognize the need of boundary spanning design, they adopt a rather vague and quite general approach for treating boundaries as a design construct. In this work, the construction given to the term of boundary is more dynamic and thus the analysis undertaken aims to mark a designing system effort that either bypass and/or mutate existing boundaries or allow the establishment of new ones, creating new opportunities for collective action. Early works by Computer-Supported Cooperative Work (CSCW) scholars conceived of boundary spanning in relation to boundary objects [46]. Subsequent research points to the need for extending CSCW inquiries beyond the information processing properties of boundary objects [47] to account for the wider context in which these artifacts operate. At core, these efforts are rooted in the quest to understand the factors that anchor the intelligibility of boundary artifacts, thus making it easier to discern what boundaries are dissolved or re-located.

3.3 – Boundary artifacts

The concept of boundary spanning artifacts originally presented to the research community of artificial intelligence to better understand the distributed decision making by Star [46]. In this definition boundary objects are presented as objects that have the ability both to span different social worlds and to adjust to the specific requirement information of each of them. This ability emerges from some key features seen in such boundary artifacts. More specifically, these artifacts tend to be rather malleable to respond to local needs and constraints of different groups who use them. Moreover, they appear strong enough to maintain a minimal common identity irrespective groups, population or visual approach. This common identity allows them to present weak constitution when treated for common use, while transformed into powerful artifacts when they fall within partial use. Finally, these materials may be either abstract or concrete. In each case, despite the different semantics of different social worlds they have a structure that is quite common in more than one world, allowing them to be identified and function as a medium of translation. The creation and management of boundary artifacts is a key process for the development and management of cohesion between intersecting social worlds [47].

The literature records series of artifacts that meet the above conditions and act as boundary objects. For example, empirical research in the area of collaborative technology highlight the prototypes [48], two and three-dimensional design sketches [47], electronic forms [49], data storage (repositories) [46, 47, 50] and various types of models (class diagrams of UML [50], CAD/CAM [48] as effective boundary artifacts. However, although the concept of boundary artifacts and their features have received wide acknowledgement by the scientific community, several studies are recorded that seek to expand the basic clarification of the term and proposed new interpreter tools that either enrich or redefine individual properties. Indicative examples of emerging concepts are the intermediary objects [51] and the boundary negotiating artifacts [49].

A more recent concept that moves on to expand the clarification and the role of boundary artifacts is the term of imbrication. The basic idea is that materials of different type can be assembled in structures that stand alone and become perceived as single. The appearance of the term in information system design feeds a new approach based on the use of appropriate representations bridging digital worlds and services, that although they constructed for a specific purpose, can be merged and serve a new functional objective.

3.4 – Metadata-based search as boundary spanning tactic

Figure 2 attempts a graphical depiction of boundary spanning functionality as from its technological status. In particular, it is formulated the hypothesis that such boundary spanning functionality can be implemented either by embedding appropriate code or through aggregation of content that exists in external sources/spaces or through the integration of applications and/or services.

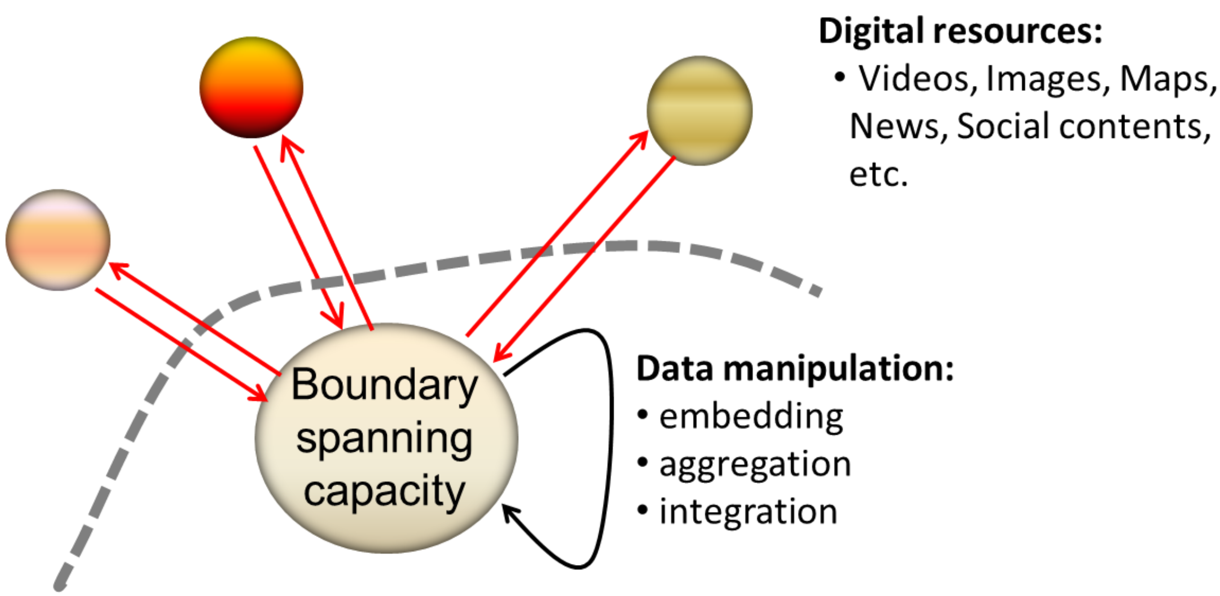

Figure 2: Boundary spanning capacity

Web search mechanisms can be viewed as boundary spanning artifacts as they gather information across multiple sources irrespective the boundaries (weak or strong) that should span. In Information Retrieval research it has been extensively investigated how, based on a query, the relevant resources can be returned. Traditional techniques look at the content of both the query and complete resource set and try to find matches based on this content. This way, a subset of search results is returned, namely the relevant resource set [5]. The noticeable here is that often this process does not take into account the issue of finding the resources.

Many efforts have been done with crawlers and page ranking mechanisms, that they are not capable of determining the meaning of the search queries. However there are very few efforts on how boundaries could be spanned to retrieve the best matching resource set for a user. Thus, it is not always easy to dynamically specify the domain of search (i.e., range of places or repositories to be searched) or to set search conditions other than keywords.

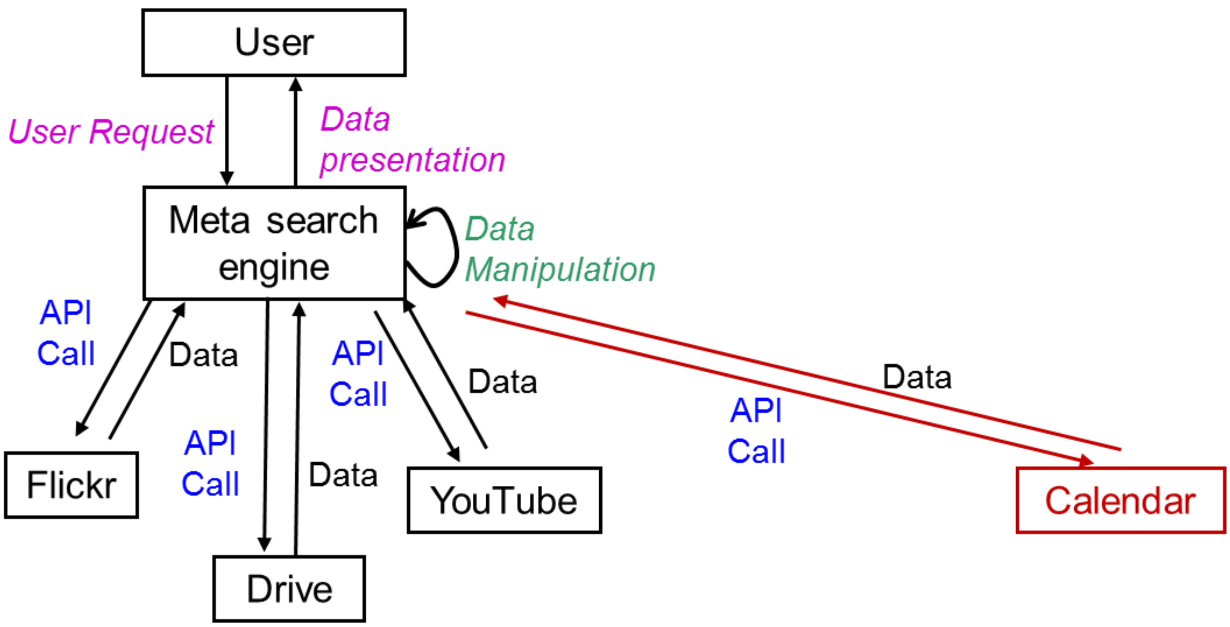

Figure 3: Engineering scaffold

In this work we attempt to build a system that collects resources across several digital spaces. Although such spaces are progressively reaching maturity levels that provide for interoperability, they still operate as "bounded" systems. As a result digital resources tend to become "private" assets which in turn constrain their capacity to traverse boundaries and become appropriated in contexts other than those supported by their host service. Figure 3 attempts to depict the engineering perspective that our system follows so as to enable boundary spanning capacity and create collections across bounded systems according to user's parameters and ambitions. The tactic we follow to bridge the boundaries of all the digital spaces (Flickr, YouTube, Drive and domain-specific repositories such as custom enhanced digital calendars) that have been selected is the use of their public APIs.

One basic boundary that a web search engine first encounters is the information accessibility from their host web location. What is happening at the moment is that more and more information is becoming available via Application Programming Interfaces (APIs). The prototype search mechanism proposed in this work makes an extensive use of public APIs from several discreet and bounded digital spaces. This APIs function a key role that enable to us to span the boundaries of each of those digital spaces that we collect information. In that way, external boundaries for us are bridging different worlds and compose something new. Moreover our search mechanism consists of several search parameters that enable or disable scopes and options for each search iteration according to user’s choices. In that way it is involved the notion of internal boundary spanning, where boundaries that exist inside our system define the search process along with the search results.

Chapter 4 - Implementation strategies

This chapter focuses on the development and implementation of a prototype of a search mechanism that follows the engineering scaffold described in the previous chapter. Firstly, we present the development platforms and the technologies used for this implementation. Then, we briefly comment on the digital spaces being exploited describing their scope and public APIs. Finally, we elaborate on the search component's architectural underpinnings.

4.1 – Technologies and digital spaces

This section is about presenting the development platforms and the technologies we exploited for the implementation of the search mechanism. First we present the development platforms and the development environments. Then, we provide a brief overview of the programming languages and libraries that we used during this implementation. Then, we briefly present the digital spaces that we chose and why.

4.1.1 – Development platforms, environments and technologies

Our search mechanism was developed using Google App Engine platform while it was also integrated in the Liferay Portal to serve specific requirements of a R&D project. Figure 6 depicts the early and late version of the system's basic user interface. As seen, there are some considerable changes both in the UI and the features provided, which we will analyze in the next chapter.

Google App Engine

Google App Engine is a Platform as a Service (PaaS) that lets programmers build and run web applications on the same scalable systems that power Google applications. Using Google App Engine, there are no servers to maintain for the developers as they can write their application code, test it on their local machine and upload it to Google with a simple click of a button or a command line script. Once the application is uploaded to Google, developers take advantage of the high performance infrastructure and Google hosts and scales the uploaded application for them. Additionally, a Software Development Kit is provided for a variety of programming languages (e.g. Java, Python, PHP, Go, etc.) along with a secure sandbox environment, that simulates all services on local computer, APIs and libraries and deployment tools that allow developers to upload applications on cloud and manage all different versions of them.

The SDK manages an application locally, while the Administration Console manages it in production. The Administration Console uses a web-based interface to create new applications, configure domain names, change which version of an application is live, examine access or error logs summarize the query load and many more features (see Figure 4).

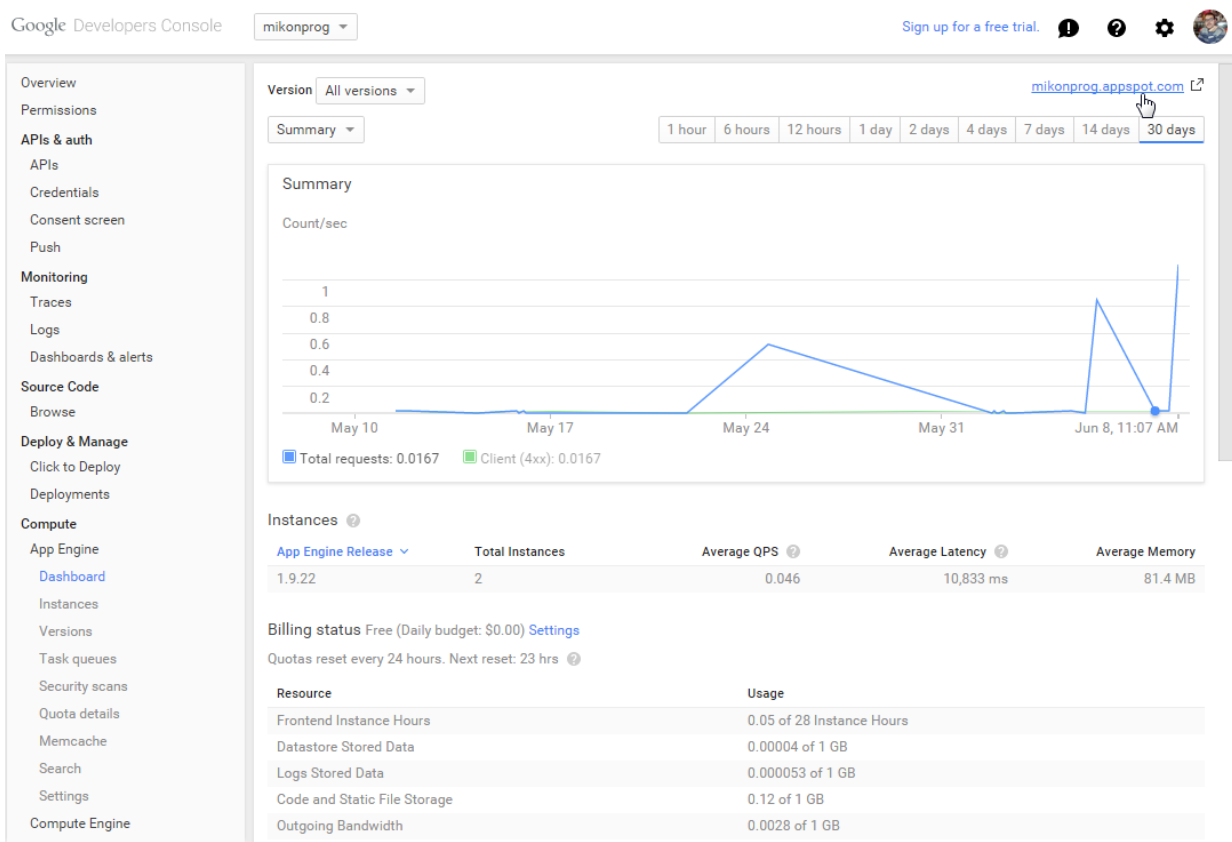

Figure 4: Google Developers Console

The local setting development process took place in Eclipse Integrated Development Environment (IDE). Eclipse IDE was selected for several reasons. Firstly, the Eclipse Foundation, in particular, has long espoused the plug-in approach and garners the prize for supporting the most technologies of interest to Java developers. Eclipse is the dominant Java IDE by all measures, especially in terms of adoption and the size of its plug-in ecosystem. In our work, we focused specifically on Google Plugin, which was the most crucial factor of our choice, due to the integration to Eclipse and its feature maturity against other IDE's Google Plugins.

Google Plugin for Eclipse enables developers to build and deploy cloud-based applications. It is a set of software development tools that simplify App Engine development providing features simple as “new web application wizard” and “one-click deploy to App Engine”.

Liferay Portal

Search mechanisms such as the one presented in this thesis frequently become integrated in enterprise solutions such as ERPs to facilitate domain-specific purposes. To this effect, we have chosen to examine the integration of the search mechanism as an add-on option of the Liferay Content Management System’s web ecosystem. Liferay Portal includes a built-in web content management system allowing users to build websites and portals as an assembly of themes, pages, portlets/gadgets and a common navigation. Liferay's support for plugins extends into multiple programming languages, including support for PHP, Ruby and JAVA portlets. This is one reason why it was chosen to host our search mechanism.

Figure 5: Liferay portal ecosystem

Liferay provides a powerful and flexible CMS to make fundamental changes to the way we do business. It is one of the most popular CMS in the market for managing and administering website content and is recommended by many industry experts. Despite its many other robust features, many users make use of Liferay Portal just for its content management system, whether it be web content management or management of file-based content. It comes with many features that make content management task simple (see Figure 5) and is used for both developing personal as well as enterprise websites. In particular, Liferay's content management features can basically afford:

- Web Publishing that allows to easily maintain fully functional websites.
- Unified Documents & Media that houses documents, video, audio, images and other media types from one place. It can be used across an enterprise, within a specific group or for a single individual. Enterprise-wide repositories allow groups to store assets, tag them with key words, lock them, search for and leverage them in web pages, or download them for use offline.
- Live Page Editing and Scheduling, where pages from a live site can be edited and previewed without affecting what is seen on the public site, then scheduled for future publishing all within the online editor.
- Integrated Collaboration Tools such as wikis, message boards, blogs, activity tracking, instant message, e-mail, shared calendar, pools, announcements and alerts, tags and categories.
- Advanced Workflow and Approval Processes that allows users to produce their own personal workflow and define the amount of approval paths based on their own distinctive business requirements and operational units.

Another key feature of Liferay which determined its selection is the fact that it is open source offering many capabilities for tailoring, customizing and building portlets from scratch.

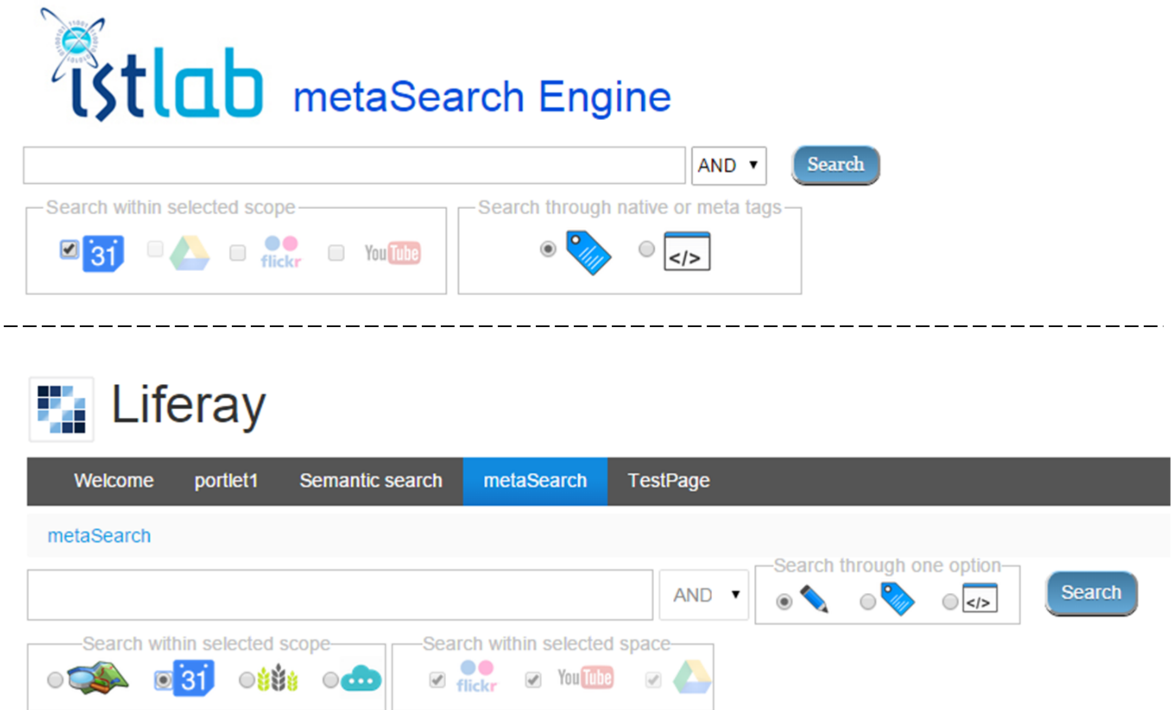

Figure 6: System UI, early (Google AppEngine) and late (Liferay portal) version

Java Servlet Technology

The Java servlet architecture provides an excellent framework for server-side processing [52]. Servlets are the Java platform technology of choice for extending and enhancing web servers. They provide a component-based, platform-independent method for building Web-based applications, without the performance overheads and process limitations. Furthermore, servlets have access to the entire family of Java APIs including a library of HTTP-specific calls and receive all the benefits of the mature Java language, including portability, performance, reusability and crash protection. Unlike proprietary server extension mechanisms (such as the Netscape Server API or Apache modules), servlets are server- and platform-independent. This leaves users free to select a "best of breed" strategy for servers, platforms, and tools.

Today servlets are a popular choice for building interactive Web applications. Third-party servlet containers are available for Apache Web Server, Microsoft IIS, and others. Servlet containers are usually a component of Web and application servers, such as BEA WebLogic Application Server, IBM WebSphere, Sun Java System Web Server, Sun Java System Application Server, and others. Servlets are most often used to:

- Process or store data that was submitted from an HTML form
- Provide dynamic content such as the results of a database query
- Manage state information that does not exist in the stateless HTTP protocol, such as filling the articles into the shopping cart of the appropriate customer

4.1.2 – Programming languages and libraries

In this section we will briefly quote the programming languages, along with their specific libraries and public APIs used to the development process within this context.

Java and JSP: Java is a high-level, general-purpose, concurrent, class-based, object-oriented computer programming language with platform independency to be its key characteristic. Java Server Pages (JSP) is an extension of Java`s servlet technology created to support authoring of HTML and XML pages, making it easier to combine fixed or static template data with dynamic content.

JSON and json-simple: JavaScript Object Notation (JSON) is an open standard, language-independent format that uses human-readable text to transmit data objects consisting of attribute-value pairs. Code for parsing and generating JSON data is readily available in large variety of programming languages, including Java. To this end, we chose json-simple as a third-party library that helped us to easily encode or decode JSON text.

JavaScript: JavaScript is an object-oriented dynamic computer programming language, commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed.

jQuery and jQuery UI: jQuery is a fast, small, feature-rich JavaScript library that simplifies things likes HTML document traversal and manipulation, event handling, animation and Ajax. It provides an easy-to-use well-documented API that works across the most popular browsers (i.e., Chrome, Firefox, Internet Explorer, Safari and Opera). jQuery UI is a curated set of user interface interactions, effects, widgets and themes built on top of the jQuery library.

4.1.3 - Digital spaces and public APIs

In this sub-section we present the digital spaces that we used to this implementation along with their public API affordances. For the selection we took into consideration the maturity of the public APIs they provide so as to enable feasible searching opportunities by third party applications and other services. At first sight, it is worth noticing the apparent functional dissimilarity between these spaces, however in the next few sections we will illustrate how they can be semantically glued to serve the purposes of the present research.

Flickr

There are a lot of supporters claiming that Flickr is the “king” of photo hosting and sharing. One major factor for this aspect is the enormous storage space that provides even for free accounts, which is one terabyte. To put that in perspective, it's a thousand times what you get with a free Picasa account (which offers 15GB along with all Google services) and it can hold over 400,000 8-megapixel photos or over 200,000 16-megapixel images. In sheer volume of photo-sharing activity, Instagram has overtaken Flickr, but that service, with its limitation to square mobile phone photos, can't match Flickr's website capabilities, vast number of interest groups, full resolution, and organizational tools. There is also a big gap between the maturity of the public APIs they provide. Facebook, too, has a larger volume of shared photos, but again, if you're serious about photography, you can't live with the image-degrading compression Facebook applies, and the distracting photo presentation.

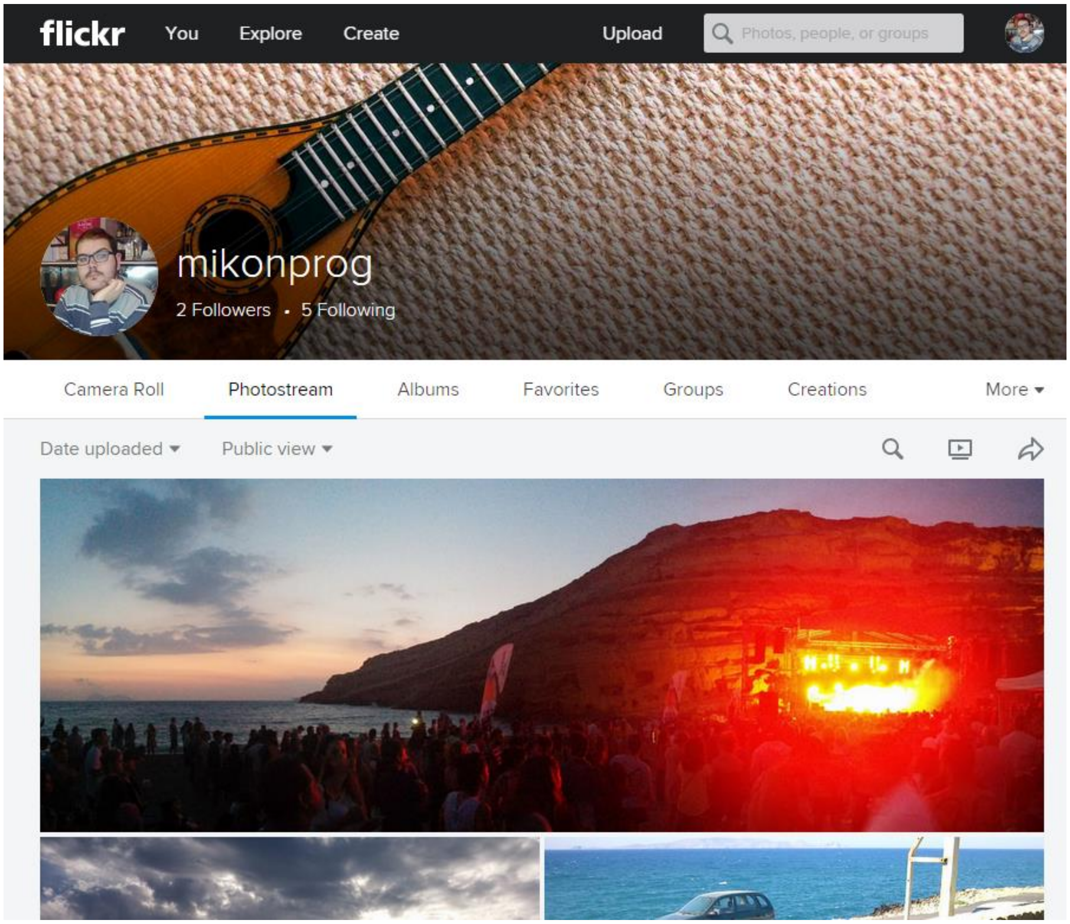

Figure 7: Flickr user's homepage

A Flickr user can upload images, view and /or comment on pictures of others. Users also can tag pictures, submit their photos to an existing group of special interest and/or create a new one. Flickr's user homepage is depicted in Figure 7. Moreover Flickr is transparent, i.e., each name, group name and any descriptive label or tag is a hyperlink that can be used for browsing the web and if it has not set private all content is publicly visible and in some cases modifiable. Like many other social media sites, Flickr also allows users to define others as "friends" or contacts and provides an interface to be presented in one place the latest images made by friends.

Flickr's public API provides a good documentation enabling developers to communicate with Flickr in order to perform retrieval or processing operations to data objects such as:

- **Activity:** recent activity on photographs owned or has been annotated by the connected user.

- **Blogs:** Flickr blog recovery involving the connected user or upload photos to one of them.
- **Cameras:** known brands and digital camera models.
- **Collections:** structures aggregate many different albums of a user.
- **Favorites:** the photos marked as favorite by the connected user.
- **Galleries:** photos assemblage belonging to other users and not to the connected one.
- **Groups:** in a group can participate several members and add many photos.
- **Interestingness:** list of photographs have shown great interest in the community of Flickr for a date selected by the connected user.
- **People:** people in the community of Flickr.
- **Photos:** the basic structure element of Flickr.
- **Comments:** comments regarding specific photos or have been made by certain users.
- **Geo:** information concerning the location where a photo was taken.
- **Photosets:** albums that aggregate multiple photos of a user. The basic structure of assemblage in Flickr.
- **Places:** information on places have been reported, related or shown in a photo.
- **Tags:** tags that have been added to a photo to determine it thematically.

A typical tag-based search API call appears below, where we call it with tags: "organic, farming":

```
https://api.flickr.com/services/rest/?method=flickr.photos.search&ap
i_key={your_key}&tags=organic%2C+farming&format=json&auth_token={tok
en}
```

YouTube

YouTube has become the largest video sharing web service. It is a digital space that allows users to participate in new ways in sharing, commenting and watching videos. Users can upload, share and search videos, add them to existing playlists and/or create new ones to connect with people who have similar interests. A YouTube's channel snapshot is show in Figure 8. As YouTube expands new features are added to facilitate social networking between users. Additionally, users can add description and tags to video they upload to better describe their content. YouTube is the most popular video hosting service and sharing, collecting a rate close to 20% of global Internet traffic. It is considered one of the largest video libraries in the world and each visitor can watch any of them without become a member. However every user with a Google account automatically becomes member on YouTube and has the right to upload videos to his/her channel limited up to 10 minutes and less than 1 GB each.

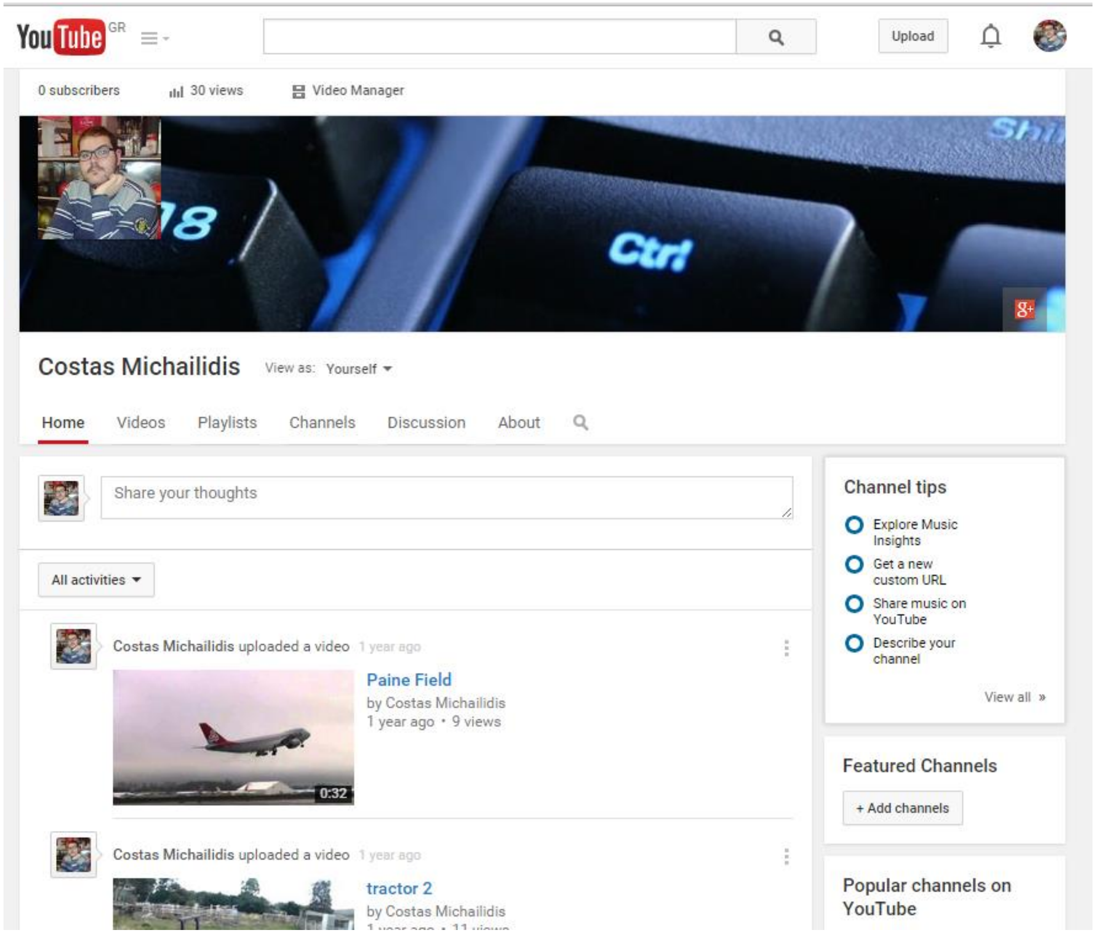

Figure 8: YouTube user's channel

YouTube has a smart auto-fill search mechanism that makes it easy for someone to find the video they are looking. Videos are played through a feature-rich player that can be embedded on any web page in a simple and fast way. Also it gives its users the opportunity to comment on a video or reply to another comment another user while supporting positive or negative voting possibilities both on videos and comments or replies. Finally each video can be added to playlists of the connected user and be shared on other social networks (e.g., Facebook, Twitter, Google+, etc).

Google following the same policy to all their services, provide for YouTube a free and well-structured public API (https://developers.google.com/youtube/v3/docs/) that is already in version 3.0, allowing developers to perform retrieval or processing operations in the following type of resources:

- **Activities:** contains information on activity in a specific channel or a specific user.
- **ChannelBanners:** contains the URL of the image that comes as a banner on a selected channel.
- **ChannelSections:** video subset contained in a selected channel. Such subsets may be the latest videos that have been uploaded, the most popular or videos belong to specific playlists.
- **Channels:** a YouTube channel.
- **GuideCategories:** list of categories associated with a selection of YouTube channels.
- **PlaylistItems:** the objects (usually video) contained in a playlist. Contains information on how these objects behave in the list (e.g., playback order etc.).
- **Playlists:** playlists, which are the main YouTube's assemblage structure. A list aggregates many videos and may, if chosen by the owner, to be shared to other users.
- **Search:** contains the list of the results (videos, channels, or playlists) a particular search.
- **Subscriptions:** a user’s subscriptions to various channels of YouTube. Used to alert a user when a new video has been added to the subscribed channel or when another user has commented or voted for a video channel.
- **Thumbnails:** icons of various sizes that are mapped to specific resources (e.g., channels, lists, video etc.).
- **VideoCategories:** categories which could be matched with a video of a user.
- **Videos:** a video is the main data structure of YouTube.

- **Watermarks:** watermarks are images displayed when playing a video of a specific channel, while the exact time and their occurrence durability is fully customizable.

A sample API request that searches for specific channels and retrieve information for the channel associated with the indicated username can be:

```
https://www.googleapis.com/youtube/v3/channels?part=snippet%2Cconten
tDetails&forUsername={Google_username}
```

Freebase

Freebase is a large collaborative knowledge repository consisting of metadata composed mainly by its community members. It is an online collection of structured data harvested from many sources, including individual “wiki” contributions. Freebase aims to create a global resource, which allows people (and machines) to access common information more effectively (http://en.wikipedia.org/wiki/Freebase#cite_note-nytimes-2). The Freebase homepage is shown in Figure 9. An entity is a single person, place, or thing. By exploiting all these mechanisms, we try to connect semantically resources from different digital spaces.

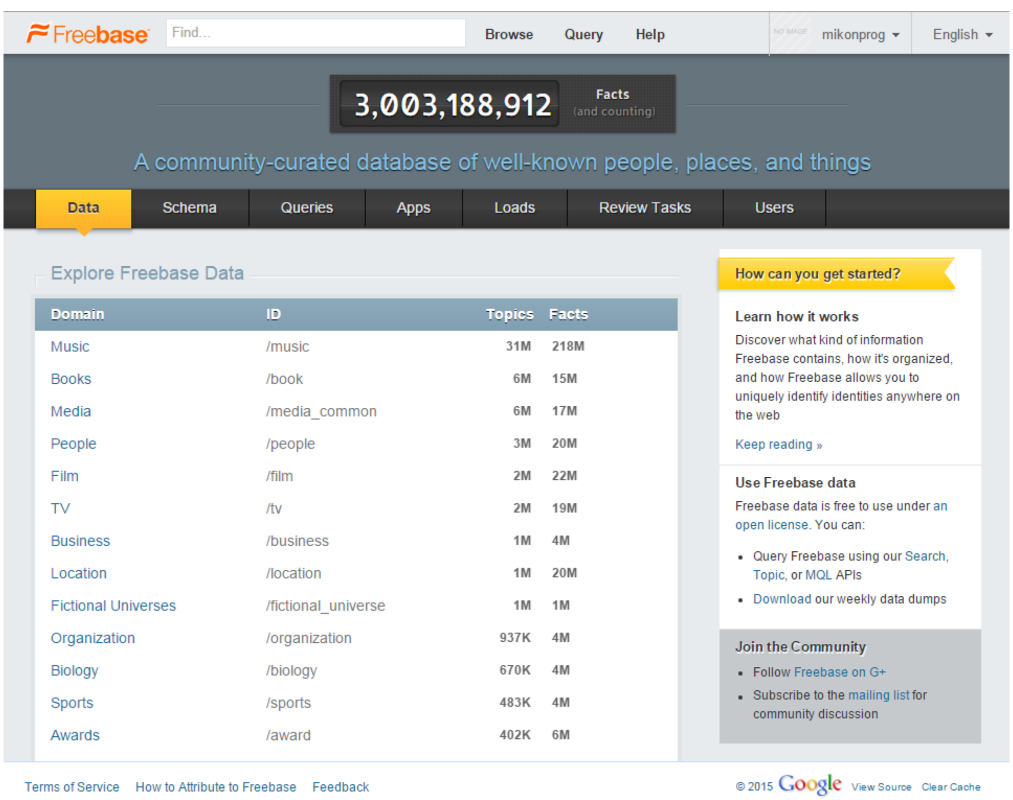

Figure 9: Freebase homepage

Freebase contains tens of millions of topics, thousands of types, and tens of thousands of properties. By comparison, English Wikipedia has over 4 million articles. Each of the topics in Freebase is linked to other related topics and annotated with important properties like movie genres and people's dates of birth. There are over a billion such facts or relations that make up the graph and they're all available for free through the public API:

- **Search:** search entities according specific filter such as people, cities, films, etc.
- **Reconciliation:** Matching items in a dataset so that they can be loaded into Freebase**.**
- **MQL:** metaweb query language enable developers retrieve more meaningful information such as collections of entities that share some common attributes or relations, specific set of facts about an entity, etc.
- **Topic:** this service will return all the known facts for a given topic including images and text blurbs. It also support filtering so that it only returns the property values of interest, building topic pages and short summaries of an entity.
- **RDF:** This service allows applications to retrieve a subgraph of data connected to a specific Freebase object.

An example of a Freebase Topic API call is presented right below and the response of it with the name of the topic and all the details is illustrated in **Error! Reference source not found.** as a JSON Object and how information are presented in Freebase:

```
https://www.googleapis.com/freebase/v1/search?query={a_query}&key={y
our_key}&filter=(any+mid:)
```

Google Calendar

Google Calendar is the service offered by Google for free to those who have a Google account to organize and manage their schedule from anywhere regardless of device or operating system. This is done by adding events with a specific start date and time and specific period of their calendar.

In Google Calendar a user can view the scheduled events with five different ways (daily schedule, weekly schedule, monthly schedule, four days view and agenda). Figure 10 depicts user’s home UI with monthly view of the calendar. It supports the adding of several different calendars to categorize events, while the user can choose which calendars will appear or not each time in the main window. In addition to time and duration, the connected user can add additional calendars containing the specific event, location taking place, title and description. Finally, for each event, the user has the ability to add reminders via e-mail or pop-up windows on specific time before the start of the event and send invitations to other users who would like to attend as guests to this event.

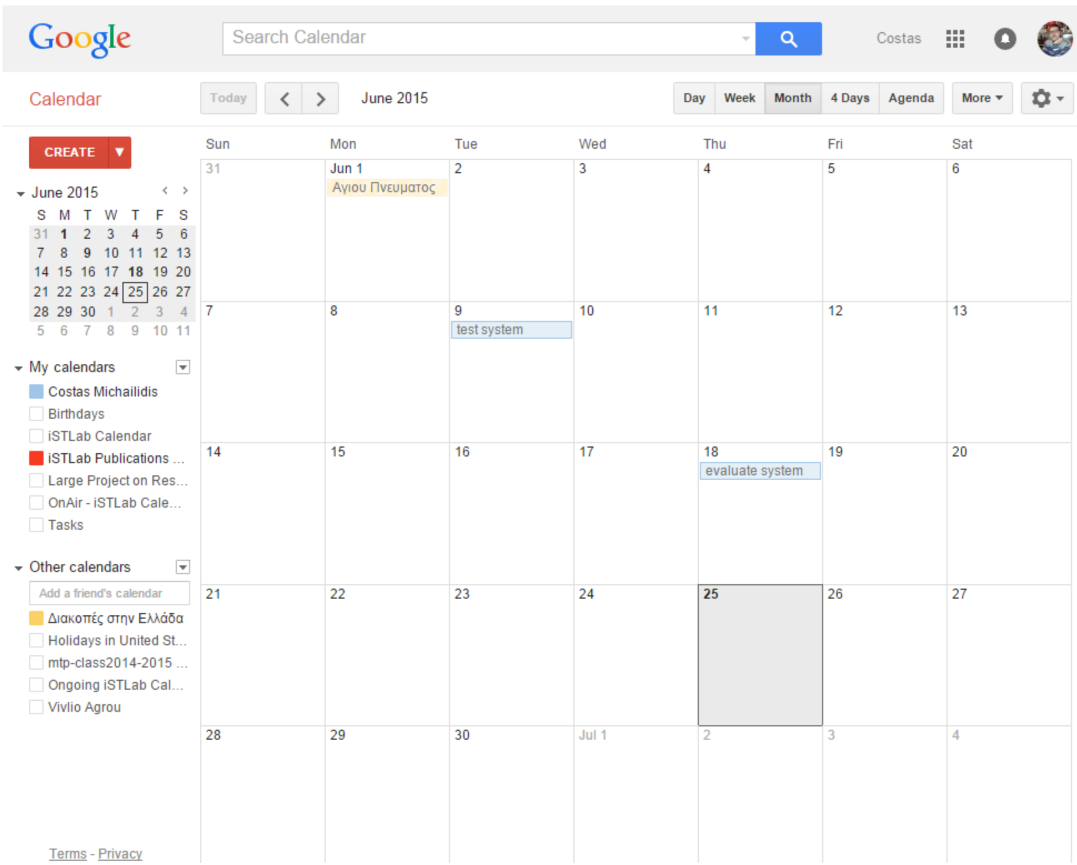

Figure 10: Google Calendar home UI

The public API provided by Google for Google Calendar service is free and well-structured (https://developers.google.com/google-apps/calendar/v3/reference/). Its current version is 3.0 and offers developers the ability to create applications that will manage resource types such as:

- **Acl:** Access Control Rules.
- **CalendarList:** a list of user's calendars.
- **Calendars:** a calendar is the basic assemblage structure of Google Calendar.

- **Colors:** manages the colors assigned in specific calendars or events.
- **Events:** an event is the basic data structure of Google Calendar.
- **Freebusy:** returns information about scheduled events at a certain time for a certain calendar.
- **Settings:** the settings of each user (e.g., time zone) for Google Calendar service.

An example that returns metadata for a specific calendar is:

```
https://www.googleapis.com/calendar/v3/calendars/{calendarId}
```

Google Drive

Google Drive is a cloud storage service of digital files and the backbone of Google's cloud services. It is provided free and automatically to anyone who has a Google account and offers free 15 GB of storage space in which can be stored and shared digital files of any type (documents, presentations, spreadsheets, images, videos, tracks, etc.).

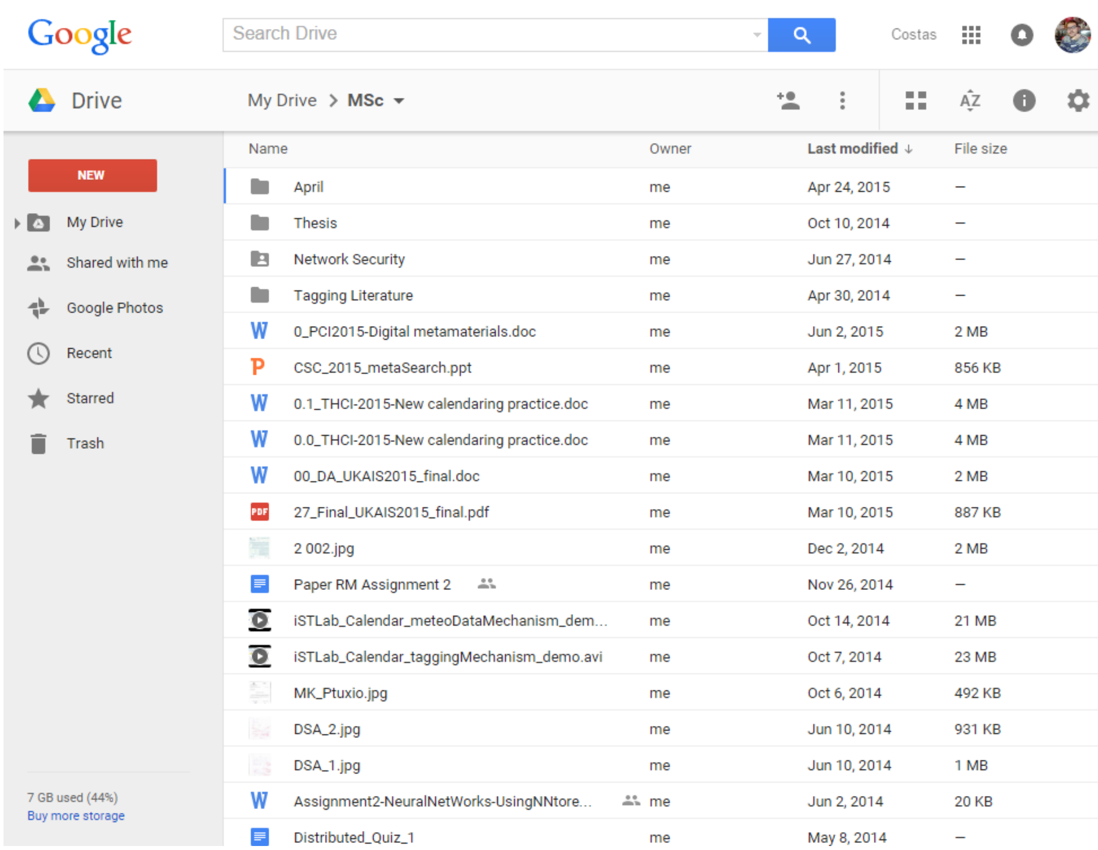

Figure 11: Google Drive home UI (web version)

In Google Drive stored files can be accessible from anywhere, only by the owner of each account until they are shared out to other users. A snapshot of Google Drive web interface is shown in Figure 11. For each file and each user we can choose the permission rights for a specific file of folder by selecting one of the three available options (view only rights, view and commenting rights and edit rights). Google Drive also interoperates with other services provided by Google as Gmail (storing attachments directly to Google Drive) or Google Docs (collaborative document editing), while providing the ability to categorize files into folders. At last, users who have the rights can view the activity history (what action has done, when and by whom), helping them understand in retrospect any changes that have taken place on files or folders.

The public API offers for Google Drive is a subset of Google Drive SDK and enables developers to create, edit, delete, search and share files through custom (or third party) applications. The well-structured API is in the third edition and provides advanced recovery methods and processes of the following resource types:

- **File:** a reference to a specific digital file with specific properties such as name, icon, type, creation date, owner, recovery address, etc. It is the basic data structure of Google Drive.
- **Parent:** a reference to the parent folder of a specific file. A file can have many parent folders. It is the basic assemblage structure of Google Drive.
- **Child:** a reference to a folder's child.
- **Permission:** authorization information for accessing a file
- **Revision:** a specific version of a file.
- **Change:** representation of a change to a file.
- **About:** an object with user information and related settings.
- **App:** list of additional applications installed to Google Drive and file types that any such application supports.
- **Comment:** representation of a comment on a file in Google Drive.
- **Reply:** representation of a reply to a comment made on a file.
- **Property:** a (custom) field of a file.
- **Channel:** a representation of a notification channel used to monitor any changes to a specific resource.

- **Realtime:** representation of a modern cooperation model which is assigned with a file.

An API call requesting information about a file's metadata by ID is following:

```
https://www.googleapis.com/drive/v2/files/{fileId}
```

Google Maps

Google Maps is another Google service that enables users to search, explore, and find the way around the world with the use of an interactive web map (Figure 12). Users can search for a place or type of places and also click the search box to get directions and view recent searches. Google Maps API for web applications provides several services such as:

- **Static Map:** Delivers maps (Street View panorama or cartographic map) as static images for embedding into pages.
- **Directions:** Provides directions between locations.
- **Distance Matrix:** Provides travel distance and time between a matrix of locations.
- **Elevation:** Elevation data for all locations on the surface of the earth, including depth locations on the ocean floor (which return negative values).
- **Geocoding:** Provides the ability to convert textual addresses into geographic coordinates.
- **Places:** Provides a range of capability to work with places in google maps. This includes check ins, add/remove places to your service, find nearby places to a location and get detailed information about a place. The places api is currently "experimental" so be careful creating production applications on this api.
- **Drawing on the map:** Users can add objects to the map to designate points, lines, areas, or collections of objects. The Google Maps API names these objects overlays. Overlays are tied to latitude/longitude coordinates, so they move when you drag or zoom the map.

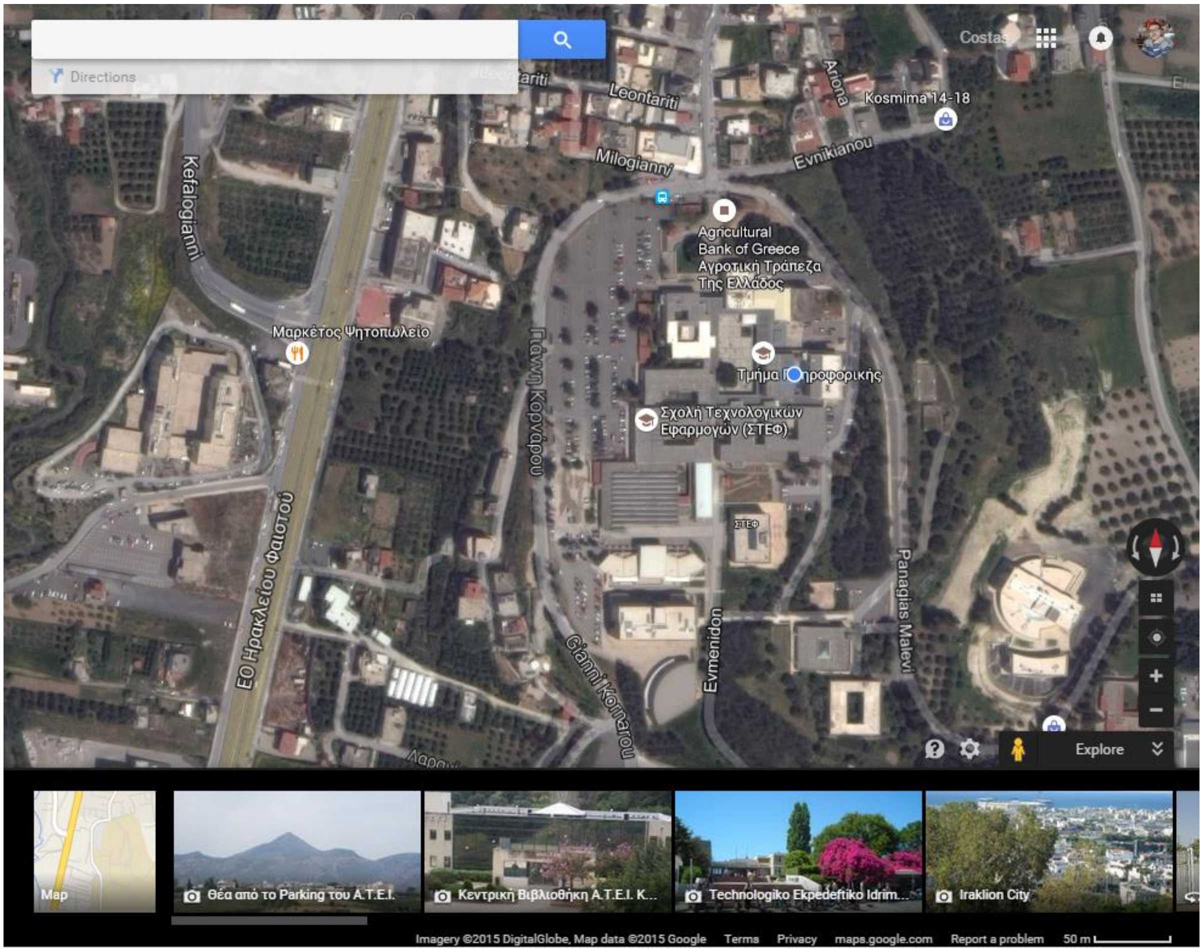

Figure 12: Google Maps home UI

4.2 – System architecture

In order to support our approach by this thesis, we have devised the architectural pattern, which is summarized in Figure 13. Motivated by attempts to advance business connectivity in inter-organizational settings [38] and aiming to disentangle novel boundary spanning tactics [53], we have developed a system which implements thematic and metadata-based search across different digital spaces and domain-specific repositories. At present our interest is concentrated on digital resources, such as photos, videos, documents, etc., retained across different digital spaces including the digital spaces/services outlined in the previous section. In Chapter 3 we presented the perspective we follow (Figure 3) to bridge the selected digital spaces. A pre-requisite is that access to these resources is granted through the provisions made by their public APIs. Users can expand their search with the response tags freely.

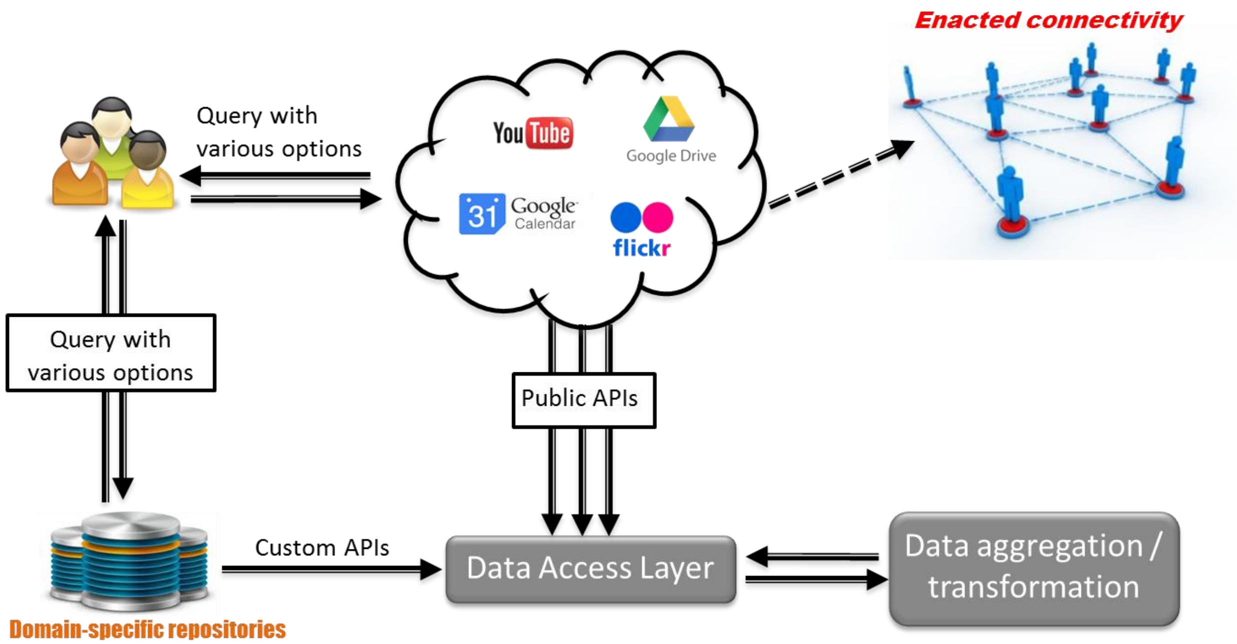

Figure 13: System architecture

The Data Access Layer (DAL) is responsible for the interoperation with external digital spaces by utilizing their public APIs [54]. Consequently, DAL is constrained solely by the capabilities offered by these APIs. Depending on the chosen digital spaces, DAL may request information on the logical structure of digital resources but also meta-data (if supported) by issuing appropriate synchronous or asynchronous API calls. Part of DAL's functionality is implemented as servlets appropriate for each online service. When a servlet is called from the front end, it combines the parameters of this call to construct an appropriate URL. Then using this URL, it issues HTTP requests (either GET or POST for retrieving or storing) to the respective service`s public API. Finally, the response to this request is properly transformed into JSON format and sent back to the front end.

Chapter 5 - System functionality and use case scenarios

5.1 – System functionality

So far, we have presented the technologies we used and the architectural pattern of the proposed solution. This sub-section presents our application and sheds more light into how we managed to re-construct the thematic searching practice. To this end, we will first discuss the application`s basic UI and then proceed by describing the process of searching.

5.1.1 – Basic UI

As every search mechanism the basis UI of this application is a custom search form, which meets several criteria and options that bound each search process according to user's needs. Search criteria consist of several scope and refinement options. There are three basic types of search (Figure 14: (1)):

- Conventional keyword search
- Tag based search, which refers to tags that exist in resources' native digital spaces (e.g., Flickr tags)
- metaTag based search

The notion of metaTags refers to custom tagging mechanisms that take place in domain-specific repositories.

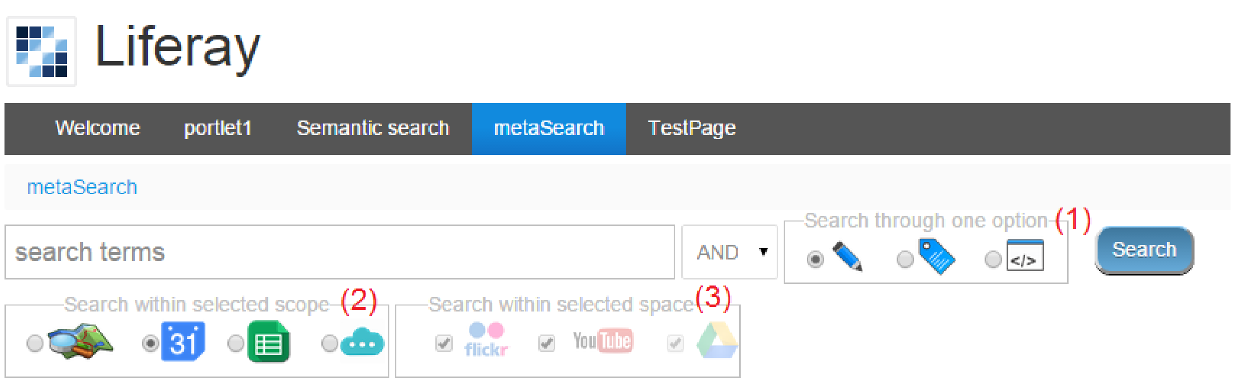

Figure 14: System's basic UI (search form)

There are also discreet scopes that this application takes into consideration to search in (Figure 14: (2)):

- Domain-specific relational databases with locations, that convert into maps
- Enhances and domain-specific Google calendars, that support functionality of storing, tagging and commenting digital resources for their events
- Domain-specific data hosted in Google spreadsheets
- Cloud based digital spaces Figure 14 (3) (e.g., Flickr, YouTube)

5.1.2 – Aggregating and presenting search results

In this section we will demonstrate with screenshots each different search type combined with the results returned and how they are presented to the user.

Searching cloud-based digital spaces

To start with, as the implementation started we present a tag-based search across Flickr and YouTube with the terms olive and harvesting (Figure 15).

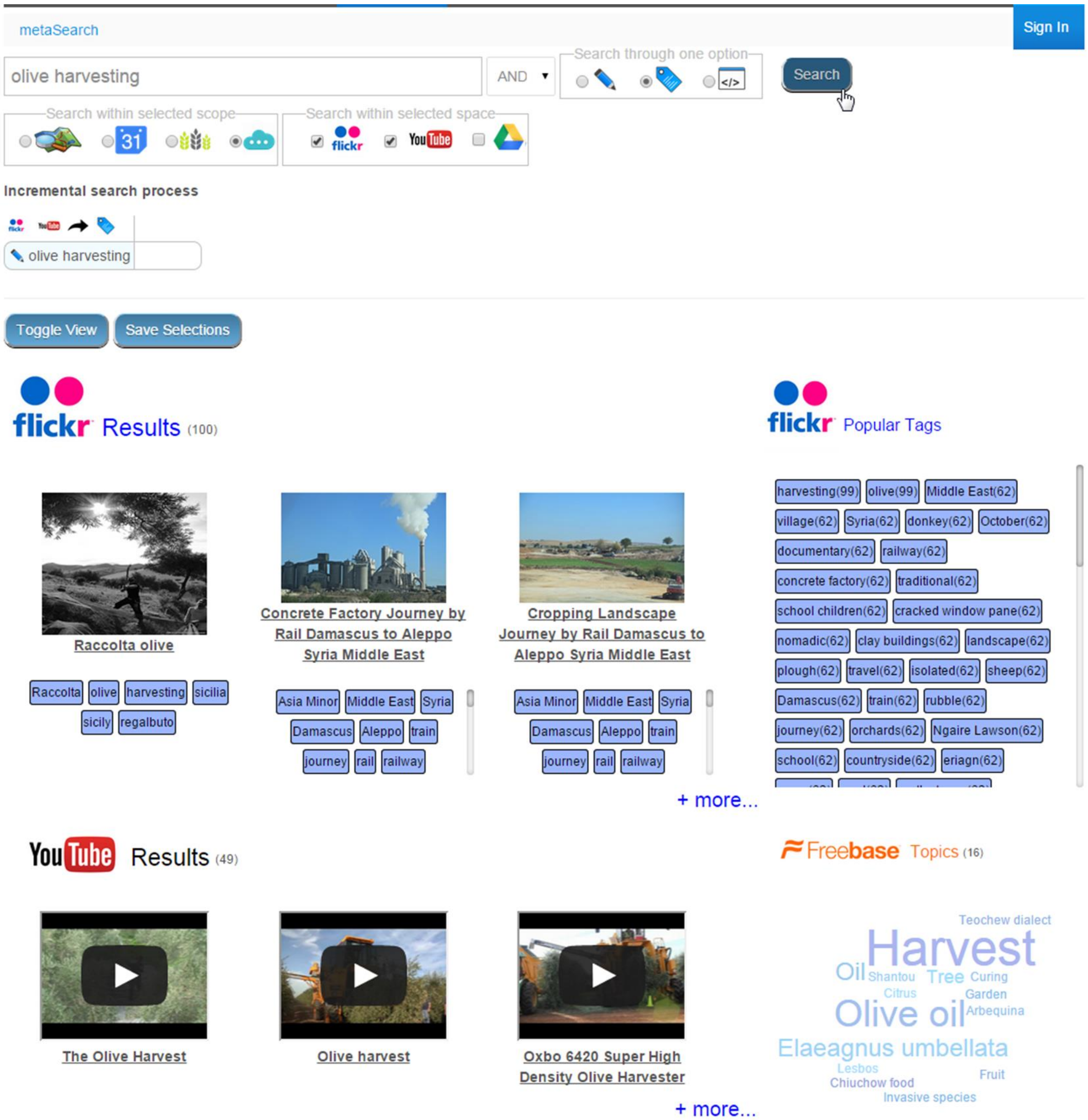

Figure 15: Flickr and YouTube search results

As presented in Figure 15 the results are presented with some meaningful information. Firstly there is appeared a search history bar that retains the terms of search, the digital spaces that search iteration took place and the type of search. In this case we can see the process bar been created with the terms "olive" and "harvesting" and above these terms it is shown that this search iteration took place in Flickr and YouTube and it is a tag-based search. Next, there are two buttons. The first one called "Toggle View" implements an alternative representation of search results (Figure 16) as a graph, which has in the middle the search terms and around them there are two more basic nodes each of whom gathers the result of each different digital space. According to [55] and the result aggregation approaches they analyze we manage to do a grouping action by the retrieved information type (i.e., image or video) instead of traditional ranking result aggregation approaches.

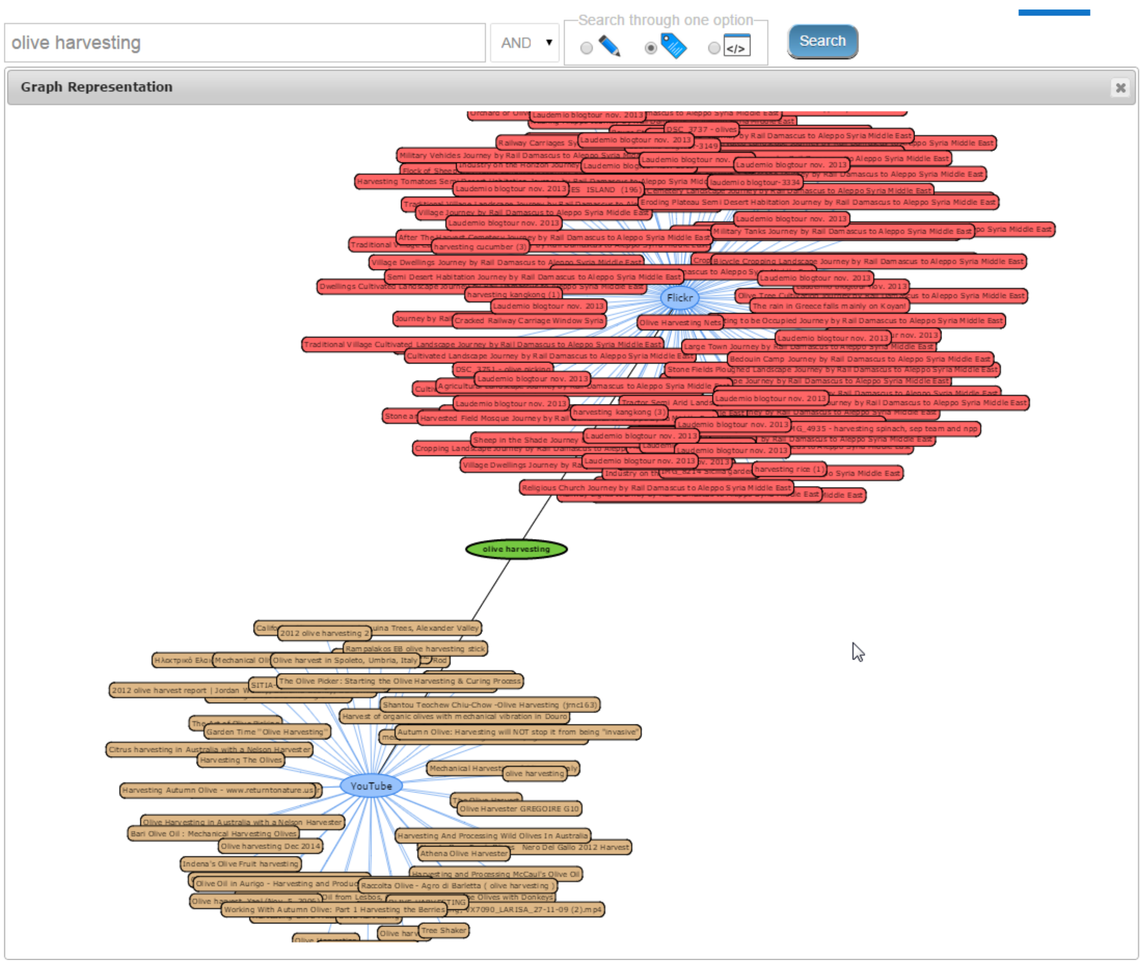

Figure 16: Flickr and YouTube results as graph representation

The second button called "Save Selections" implements a store mechanism that invokes a maintenance ability, which stores in a specific place in Google Drive virtual referents of the selected resources for future use. The stored file is a custom JSON object in a certain format. An indicative file is shown in Figure 17, where stored three selected photos and three selected videos.

```json
{
    "documents": [
        {
            "title": "Wilt",
            "resource": "flickr",
            "url": "https://farm4.staticflickr.com/3935/15468096195_086b289f12.jpg"
        },
        {
            "title": "Heaven and Earth",
            "resource": "flickr",
            "url": "https://farm4.staticflickr.com/3931/15280544039_fb90a1f634.jpg"
        },
        {
            "title": "Raspberry Picking",
            "resource": "flickr",
            "url": "https://farm6.staticflickr.com/5598/15463571395_4a7bdb544a.jpg"
        },
        {
            "title": "Organic Farming, Australia - Learn how to farm organically",
            "resource": "youtube",
            "url": "http://www.youtube.com/embed/cWsn6BDkCHM?controls=2"
        },
        {
            "title": "Richard Wiswall discusses the business of organic farming",
            "resource": "youtube",
            "url": "http://www.youtube.com/embed/36P1ZVSFFfo?controls=2"
        },
        {
            "title": "What is Organic",
            "resource": "youtube",
            "url": "http://www.youtube.com/embed/BebNsezt6r0?controls=2"
        }
    ],
    "user": "mikonprog"
}
```


Figure 17: Custom JSON form

The next section of Figure 15 is the basic search results aggregation representation, which is conceivably separate for each digital space. As shown Flickr photos are presented with their individual tags. Also, there is a collection of all Flickr photos' tags in the right of Figure 15 named Flickr popular tags, where a list of tags is presented and ranked by popularity. The interesting here is that we can see how many photos have been tagged with the queried tag or another specific tag. Each tag is functionally clickable and triggers a new search iteration.

Following, YouTube results section follows similar representation logic with some changes. Each video presented is a discreet embedded player that enables user to watch them from this user interface, instead of visiting YouTube site. However, there are not individual tags for videos. This is because YouTube as a service and YouTube Data API do not allow users to see a video's tags if it is not uploaded by themselves. In other words, only the uploader of a video has access to its tags. However, to provide semantic annotations YouTube makes use of the notion of 'topic' in coordination with Freebase and Topic API. This distinction is mapped at the level of the supported public APIs. In particular there is information related to Freebase topics inside each YouTube’s video metadata and by calling Freebase Topic API we manage to present video semantics. An example of a Freebase Topic API response is presented in Figure 18.

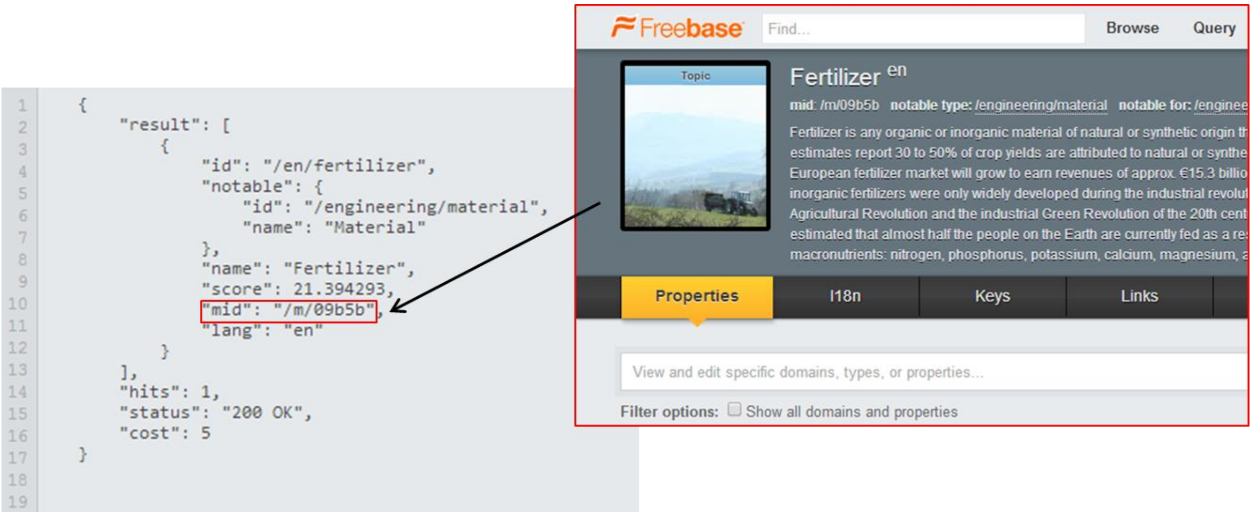

Figure 18: Freebase API response

In the right side of YouTube’s result section appears a tag cloud with the topics related to video results. The size of each word is calculated according on how many videos match the certain topic. The words in the cloud are clickable as well with Flickr tags and if clicked trigger a new search iteration.

Search enhanced Google Calendar

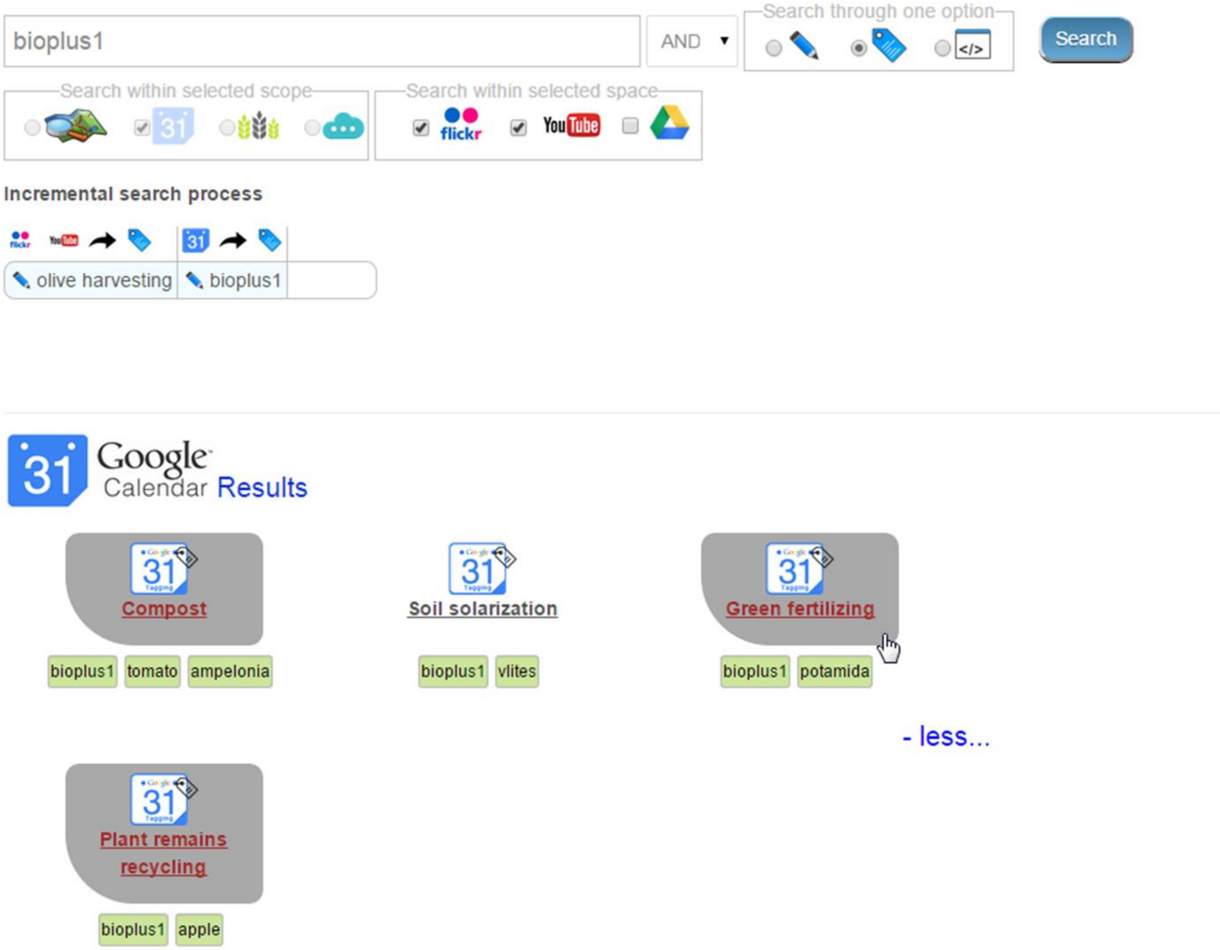

Figure 19: Calendar results snapshot

The next concept implemented by this system is to retrieve information from enhanced Google calendars [56]. The interesting feature at this point is that there is supported a custom tagging mechanism that enables users to add tags on events. Our system taking advantage of this functionality searches such calendars for events by tags. There is also supported conventional keyword based search for events. In figure p we present a search process with the term "bioplus1". As shown, results are presented in a similar rationale with the previous search iteration in Flickr and YouTube. Events retrieved by specific calendar can be selected and searched further about their information and resources that are related with them.

Once one or more events have been selected there is the ability to search resources related to the selected events across different digital spaces. In Figure v are presented the results of the two selected events from Figure m that satisfy the term harvest in their tags. By letting the search field empty there is functionality that returns all available resources in the selected events. The novelty here is that although it seems another conventional tag-based search process it refines the search incrementally as it filters the search results only for a specific set of calendar events. Moreover, the history bar indicates the level of increment as show in Figure 19.

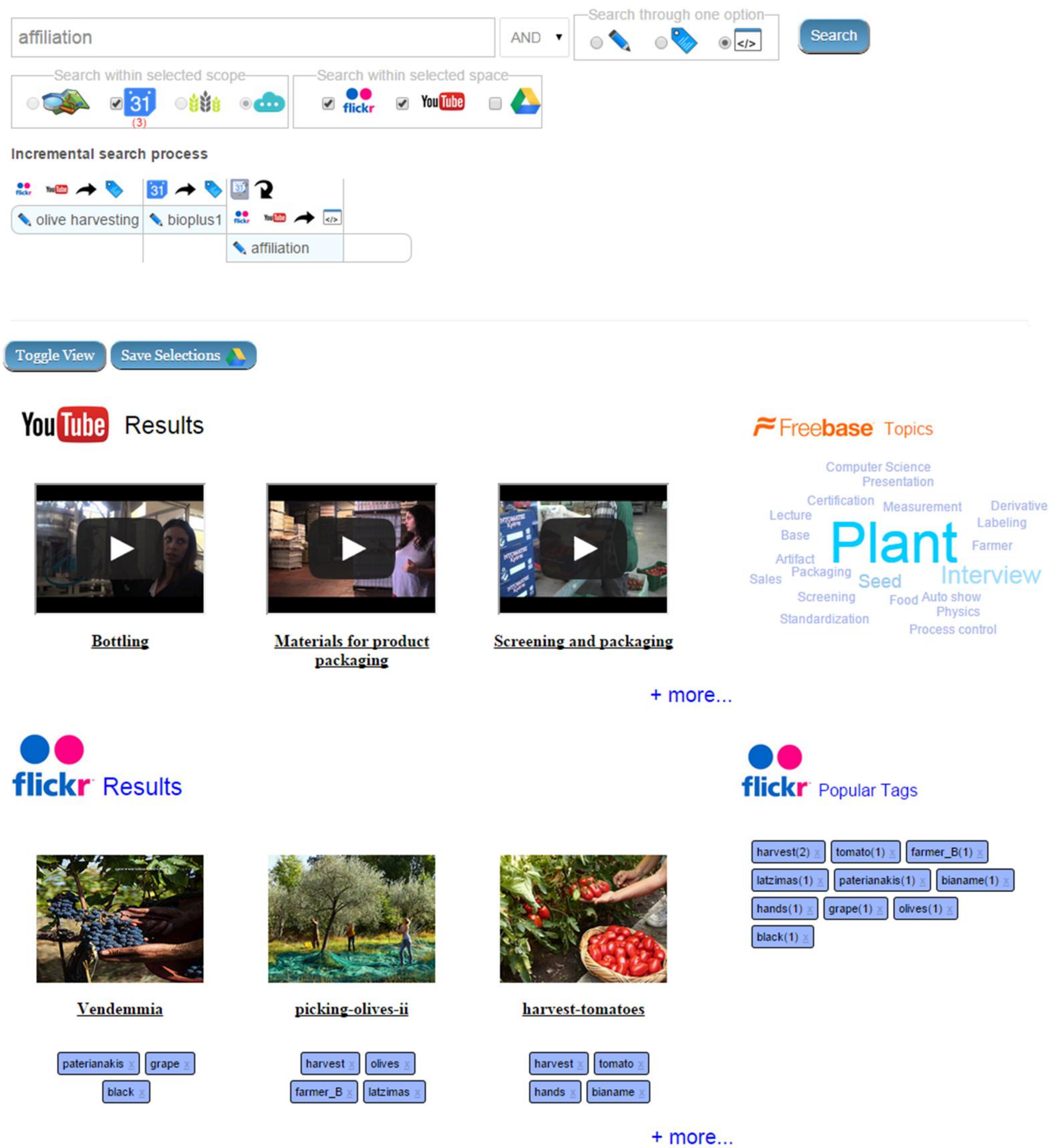

Figure 20: Calendar events' resources search result

Figure 19 and Figure 20 depict a typical search scenario where the user declares intentions (by setting search conditions and anchoring the category of the search condition and the scope of the search) while the system compiles an initial set of search results (as in Figure 19) which can then drive subsequent search refinements (as in Figure 20). Thus, it is made possible to locate calendar events tagged as "bioplus1" (Figure 19) and then compile the digital resources of these events meta-tagged as "affiliation" in the referent event irrespective their native tags retrieved by their host services (Figure 20). As shown, the latter search and retrieval step provides further details about the technology's performative capacity. Specifically, using word and tag clouds the system qualifies emergent relationships as well as the "logic" of establishing these relationships. Thus for instance, in the case of Flickr, the tag cloud summarizes the tags assigned to the search results, while in the case of YouTube it reveals topics from the Freebase and their relative popularity. By this account, the user is exposed to intrinsic properties of the search & retrieval mechanism and obtains an insight into the sociomaterial context of designated objects.

It is also important to note that the search results compiled through this arrangement could not have been assembled otherwise since the search conditions are not part of the resources‟ metadata in their host services. In other words, the specific digital collection presents an emergent configuration of distributed digital objects compiled by using linguistic markers such as tags that convey the collective wisdom of a virtual team.

Searching domain-specific repositories

This system also manages searching in domain-specific repositories that maintain information about specific domains that is manageable though custom APIs. To best present and analyze this type of functionality we will bypass this section and present them in the case studies that are presented below.

5.2 – Use case scenarios

In this subchapter we will demonstrate two use case scenarios that took place in real settings and define further the system functionality.

5.2.1 – Biodrasis project

The first scenario implements a thematic search engine in a research project in the field of organic farming through virtual alliances, which is still in progress. In this use case users can search information about the consortium in several scopes that we will present below.

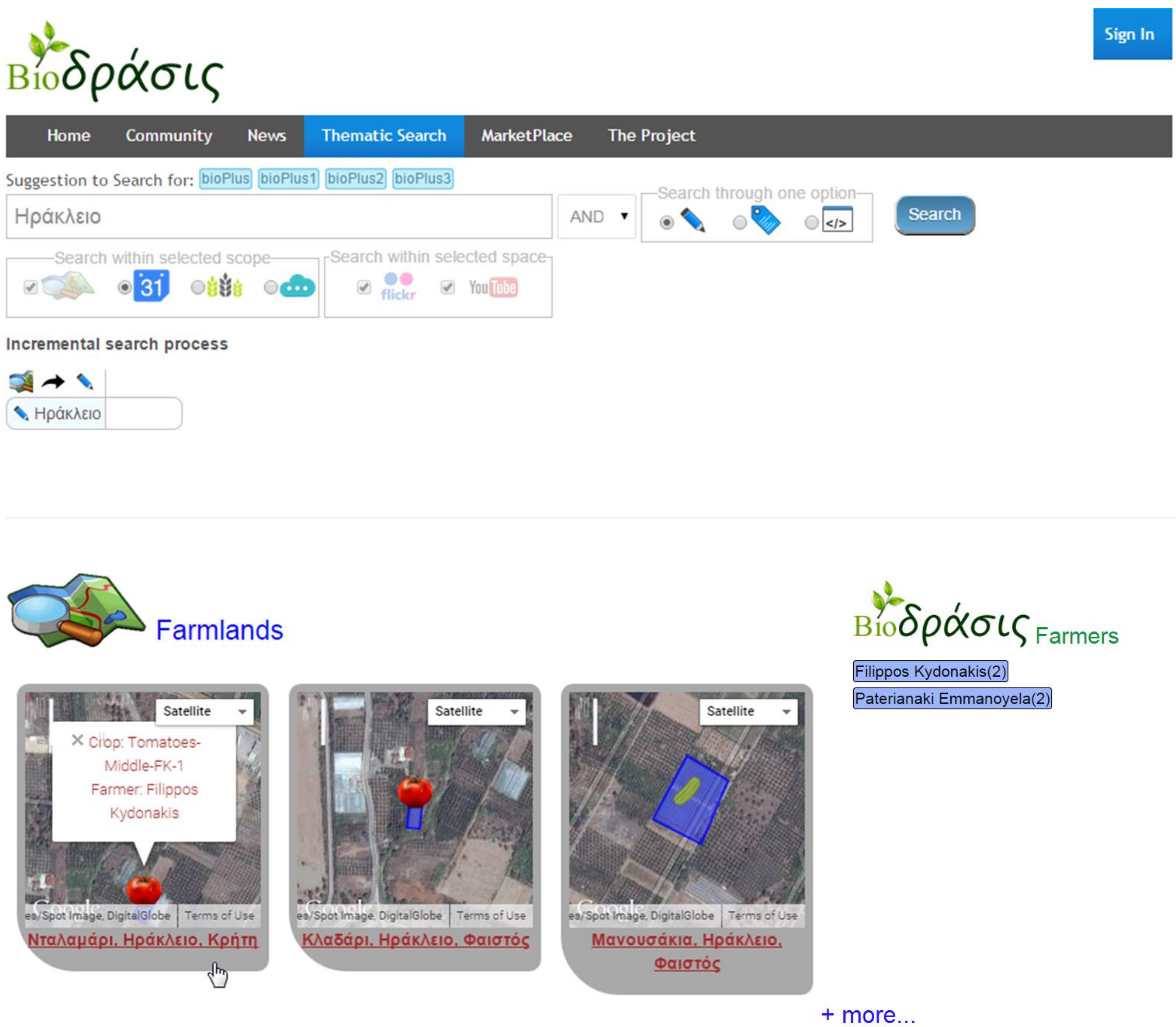

Figure 21: Searching community's farmlands

In Figure 21 it is presented the way we manage to search and present the consortium's registered farmlands. As you can see the basic UI has not major changes, except from some search suggestions that are related with a specific type of activity, which is conducted to some voluntary certification schemes (bioplus) that we will describe further below. The search results in the main section (left) depict farmlands retrieved by a domain specific repository, which is a relational database for the purposes of the project, according the search term.

As shown, each farmland has specific coordinates and we manage to draw on the map their borders as well by using Google Maps Embed API. Moreover, there is information about the name of the farmland, its owner and the cultivation produced in it if exists. In the right section of search results we gather the farmers of the farmland results to have a collective notion of inspection, e.g., how many farmlands with a specific cultivation have Mr. Filippos Kydonakis. To answer this query we should search farmlands with term the cultivation we seek and automatically find Mr. Kydonakis' name in the right section of the results and the number of his farmlands.

Figure 22: Searching farmlands' activities

The farmlands are selectable as shown in Figure 21 and we can search further information about them. This is achieved because there has been implemented a digital farmland book within the project, which is based in the functionality of the enhanced calendar mentioned above and maintains farmland activities (events) and resources as digital objects connected with them respectively. In Figure 22 is presented a new search iteration in some selected farmlands with the term “Solarization”. As shown the search results follow the perspective of the enhanced calendar search we present in previous section with some changes to the UI and the event tags retrieved. It is also worth seeing that the process bar imprints the incremental refinement of the search process. The bronze ladybug depicts information related to the voluntary certification schemes that we will analyze below.

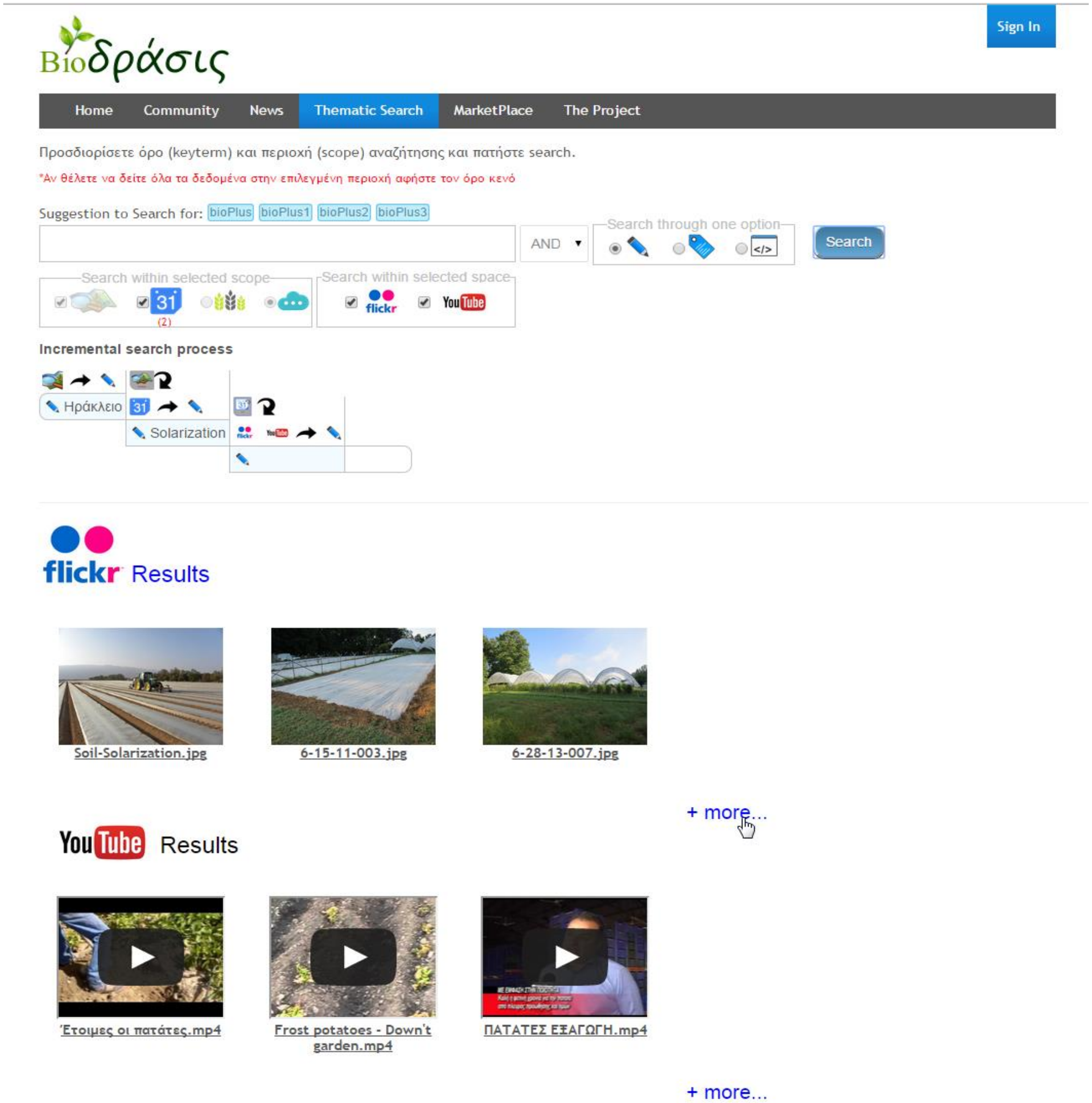

Figure 23: Searching farmland activities' resources

As in the enhanced Google Calendar, there is also here the possibility of searching digital resources that are related with farmland activities respectively. The **Figure 23** shows the resources retrieved from Flickr and YouTube and are related with the events we select above for further search. It is noticeable that there is no search term at this point. This is because there is the ability to retrieve all information (i.e. resources) in the selected scope and space of search by letting the search term empty. As you can see in the search process bar, at this point, the search process is on third level of incremental refinement.

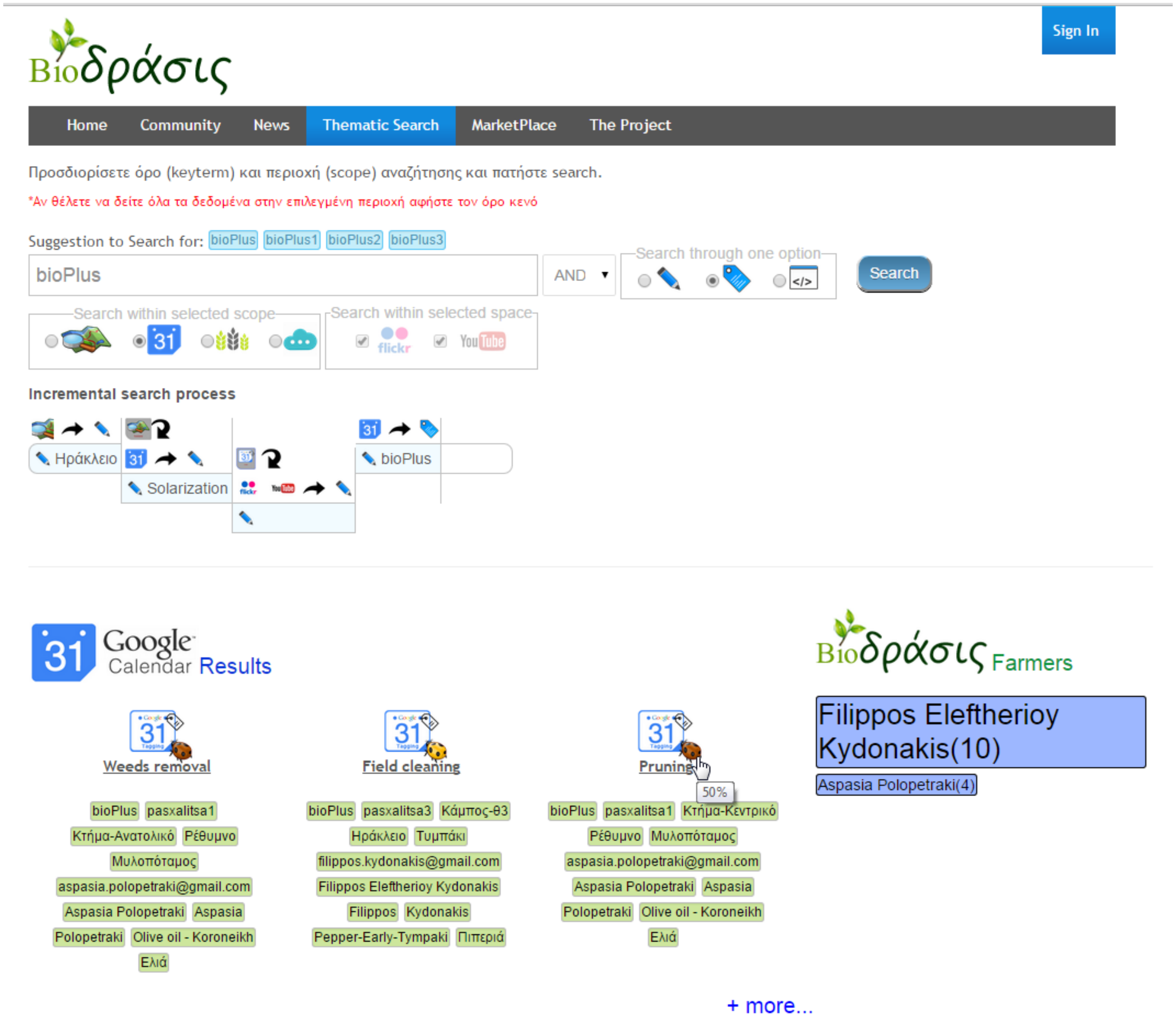

Figure 24: Searching community's farmland books

There is also the possibility to search directly for farmland activities irrespective specific farmlands. In Figure 24 we can see all consortium's farmland activities related to the term "bioplus" in the main result section (left) and the consortium's farmers on the right according on how many of those activities have they done in their farmlands.

In this project there is the notion of voluntary certification schemes. Official agriculture regulations give a minimum resultant of mutually acceptance conditions to recognize that someone's farming produces organic results. This results many organic farmers applying many specialized techniques and initiatives that respond to their ecological sensitivity to market their products under the same brand that have other biological farmers who do not apply appropriate techniques. It was therefore necessary, at this stage, to find a way, where these organic products from most ecologically sensitive farmers will be recorded through some voluntary certification schemes (called also bioplus schemes) and offer to their producers a better goodwill.

We have three levels of bioplus certification and manage to present them with three colors of ladybugs, bronze, silver and gold. As a result, if a search iteration is triggered by using the tag "bioplus1", which is related to the first level of bioplus and is depicted with a bronze ladybug, we collect meaningful information about the consortium. For example, in Figure pi we can extract the information that Mr. Filippos Kydonakis has a more serious concern about voluntary certification (generally and not specifically to a certain level) than Mrs. Aspasia Polopetraki. We can also retrieve information about farmlands and if they have granted a certain level of bioplus by searching farmlands with one of the three bioplus levels as tag on farmlands' metadata.

Another domain-specific medium that this system manages to search and retrieve information is the notion of Crop Profiles. Crop Profiles are certified profiles provided by a competent authority (e.g., an organic farming certification authority) and may be followed by a farmer, an organic farming partnership and/or the whole organic farming community. These profiles aim to guide farming process with specific activities and secure that the result will be correct according to a respective farming plan.

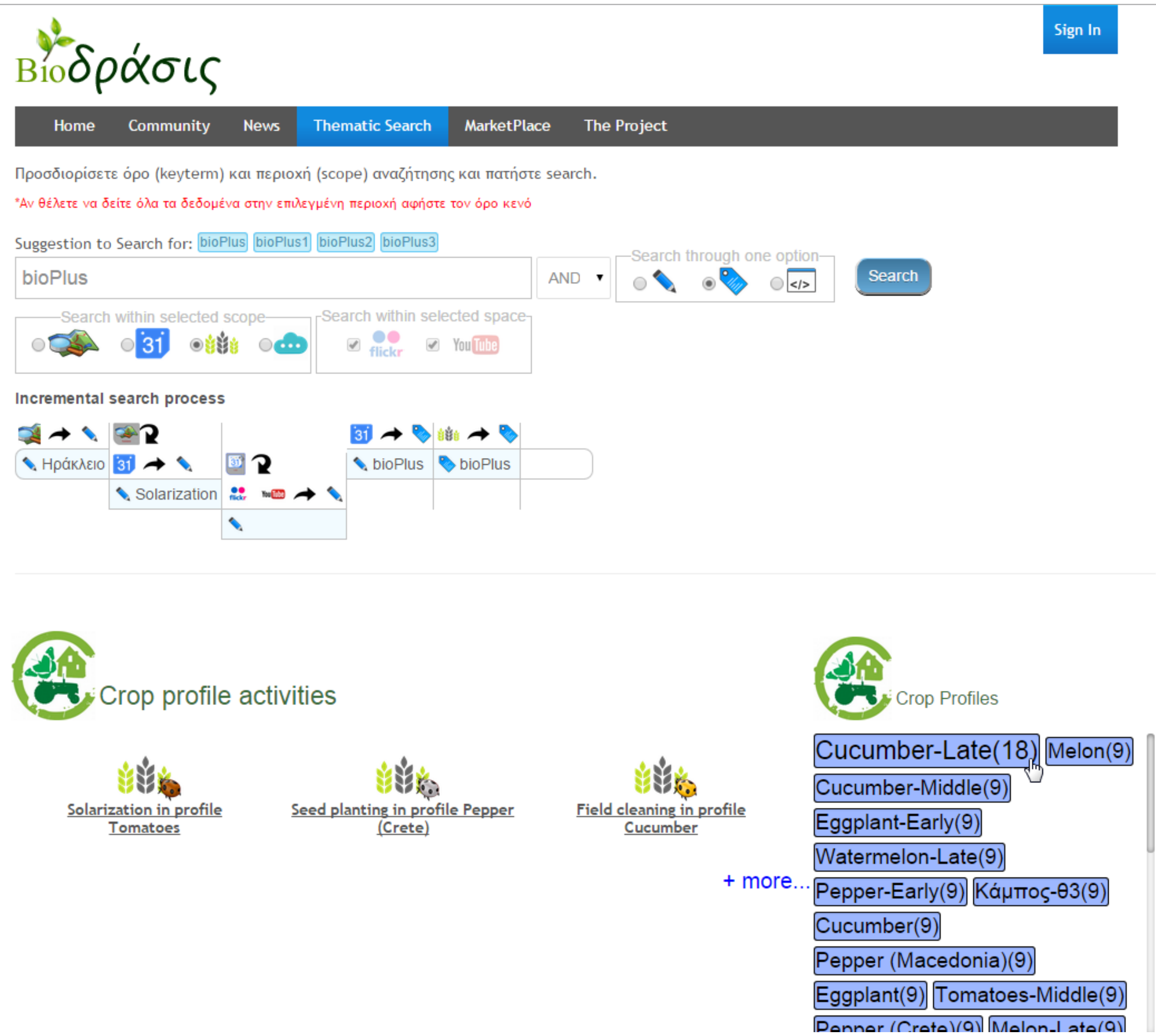

Figure 25: Searching community's crop profile activities

In Figure 25 it is presented the ability to search crop profile activities. The results vary on the term searched for. There is the ability for conventional keyword based search, there is the ability to retrieve all consortiums' crop profile activity by letting the search term empty and there is the ability to retrieve activities that participate in a certain bioplus level or more general in bioplus as Figure 25 depicts. There is also a summative section in the right side of Figure 25 that presents the Crop Profiles according to how many crop profile activities they contain.

5.2.2- Publication retrieval

The next use case presented in a lecture of the course "Computer-supported Collaboration" in the Master's program "Informatics and Multimedia" and highlights the retrieval affordances of this system. The illustrative problem was to find out all journal papers by Mr. Akoumianakis D. since he has joined iSTLab at Technological Educational Institute of Crete.

The tentative approaches were to Google it, to make use of Google Scholar and use the iSTLab's website. The result was that none of these are satisfactory for several reasons. The failure conducted to the conveyance of the search intention in sufficient details, which led to the following aspects:

- I need to specify whom I am searching for (i.e., D. Akoumianakis)
- What is it I am searching for (i.e., journal type publications)
- What is the scope of search (i.e., publications associated with his presence at iSTLab)
- It would be useful to know where these publications reside

Taking into consideration the above aspects we can reformulate the problem and the purpose of search on how can we find and compile results that satisfy a keyword or are qualified with a tag or a metaTag in cloud-based digital services (such as Flickr, YouTube, Google Drive) or domain-specific repositories. As a result the search intention is now expressed by qualifying the type of search condition, designated the scope of the search and specifying the digital spaces of relevance.

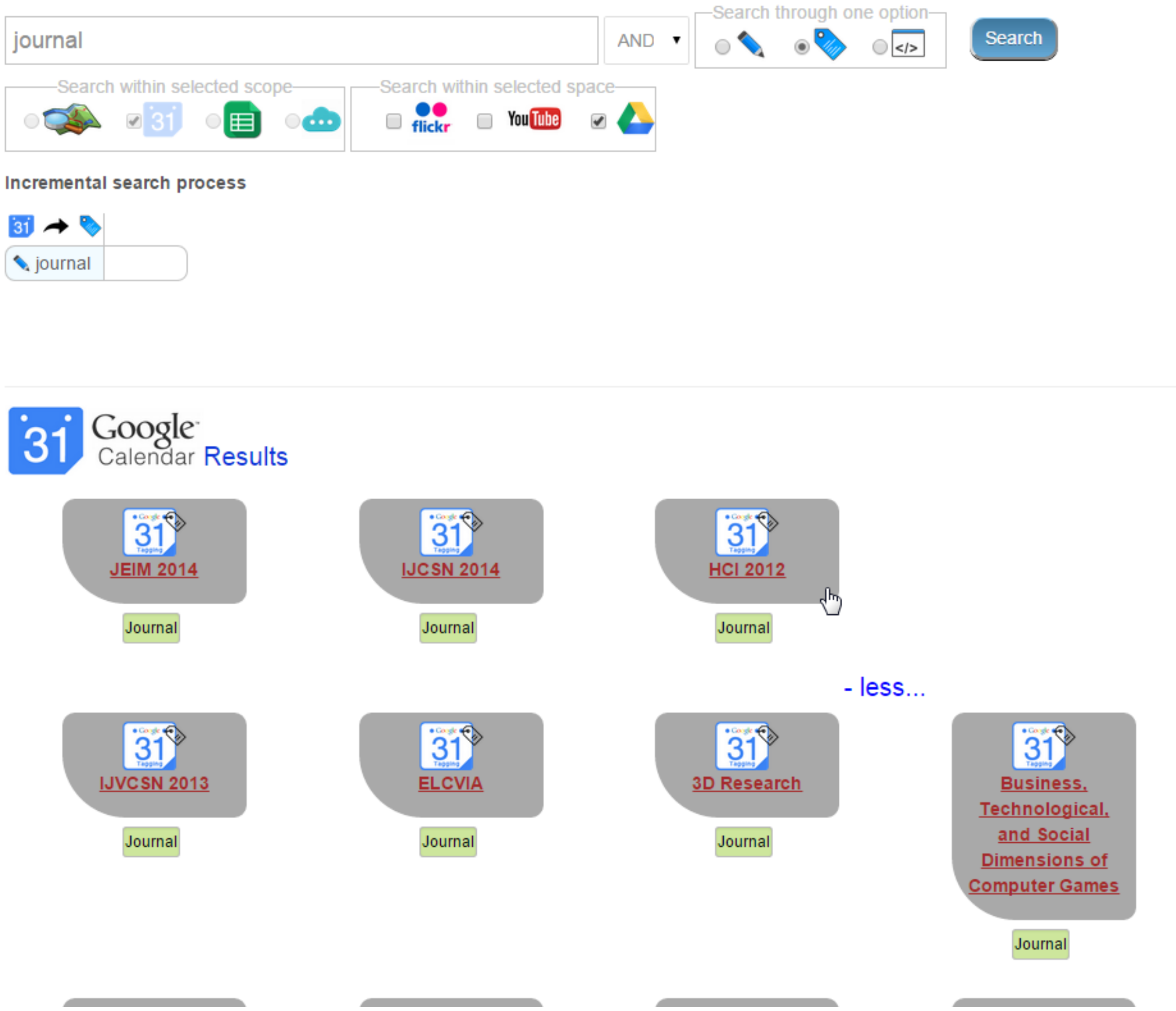

Figure 26: Searching journal type events

Assume now that there exists an academic calendar that hosts events about our research group's publications. The notion of this calendar is following the enhanced Google Calendar we described in previous sections and maintains the papers hosted in Google Drive and related to the events that depict the journals or the conferences we have participated. Figure 26 shows a search iteration that we search all journal events of our academic calendar. The search option is tag-based because the events have been tagged as conference, journal or book chapter respectively and the search scope is Calendar.

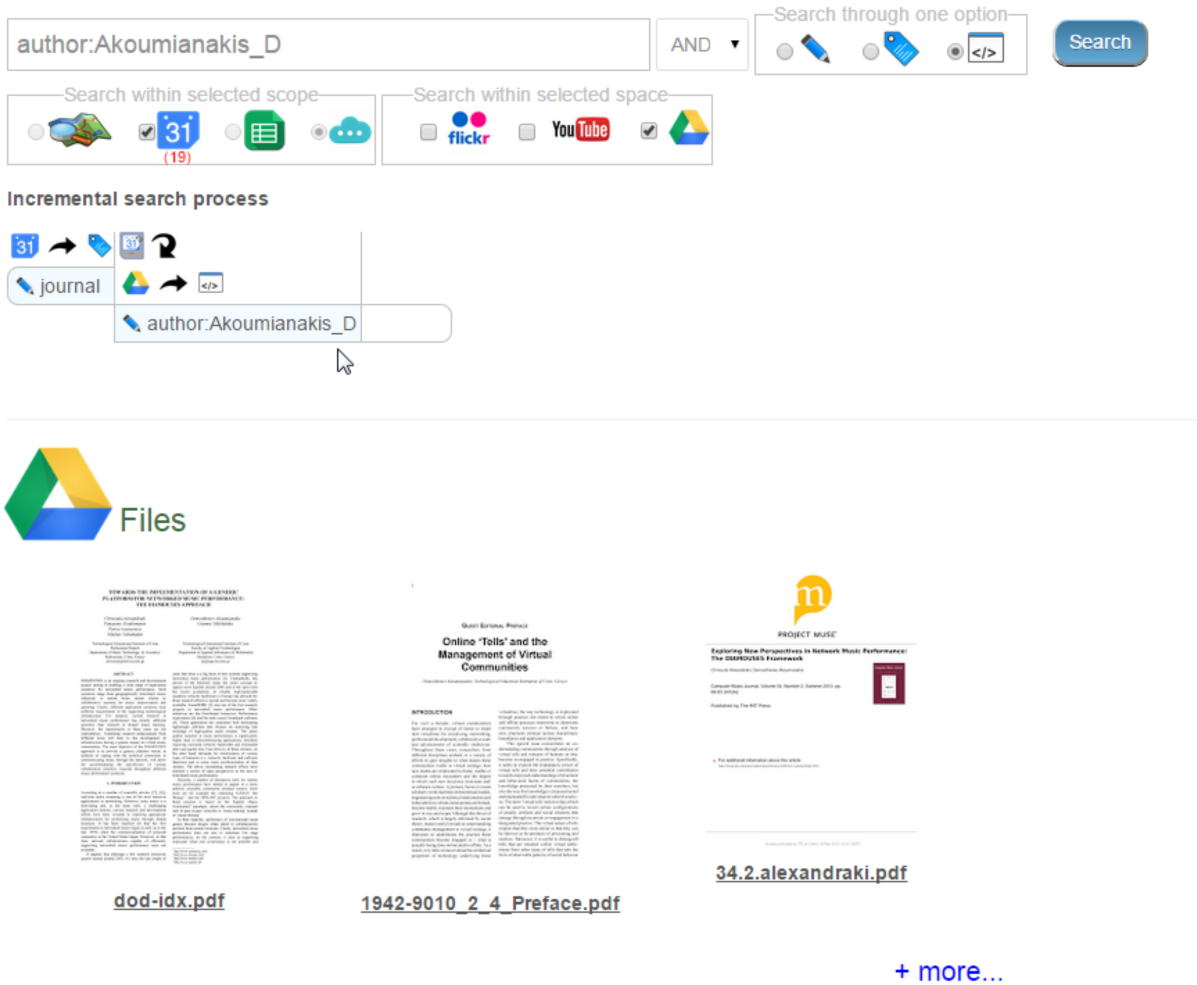

Figure 27: Searching resources across Google Drive (inside selected events)

After selecting all retrieved journal events we refine the search by the author we seek (i.e., Mr. Akoumianakis), we select the metaTag search option and choose the Google Drive icon to specify the digital space hosting traces the designated target events. Figure 27 depicts the result of this query, which is the answer to the initial scenario inquiry.

To sum up, there are some interesting points to notice about our system according to this use case. Firstly, the results returned reside in Google Drive, which does not support any tagging mechanism. Next, the compilation depicted was made using shared emerging vocabulary. It is also noticeable that no other search engine could return this set of resources. Moreover, the system provides explicit triggers indicating how the collection was compiled. At last, it is worth saying that since out calendar allows digital material to span the boundaries of various services, we can anchor publications so as to convey:

- type of resource (manuscript, slideshow, video clip, photos, etc.)
- jargon terms or user defined-qualifiers such ‘call for papers’, ‘project meeting’, etc.
- category of publication such as journal, conference paper, chapter, etc.
- photos albums related to the publication
- video clips supporting the publication

5.3 Summary

This chapter describes the system functionality and the case studies of the present master thesis. It begins with the ambiguous functionality features in section 5.1. On the next section we demonstrate two case studies took place in real settings and further the functionality of our system with more domain-specific search and retrieval features.

The case studies described above can be generalized to different application domains (e.g., health care and music) where user activities may be augmented by accessing resources distributed in various services. Thus, the problem at hand may be abstracted to depict the concept highlighted in the use cases. Specifically, it would be useful for users to maintain collections of semantically relevant resources and obtain access to them by exploiting meta-tags that point to interrelated resources across digital spaces.

Chapter 6 - Concluding remarks

This final chapter consolidates the present work by offering a brief summary of this master thesis and providing answers to the basic addressed research questions. Then, it elaborates on some limitations to propose guidelines for subsequent technological improvements and future research.

6.1 - Consolidation

This thesis aims to study a concept which is broadly open and occupied extensively the research community: how to improve search mechanisms in a metadata-based way, when bounded in specific domains and sources. Our work has advanced a scaffold and a design approach for aggregating the separate result information, each with its own type and source. The scaffold was used to guide design decisions and implementation of a new search mechanism which is constituted by search options with multiple scopes and states and increased searching capabilities that spans digital services.

To this end, we took advantage of information metadata and more specifically several tagging mechanisms as a way of semantics increment to retrieve information that are closer to users' needs. This can be easily understood with the example that a photo when tagged by many different users has more meaningful information rather that an untagged photo. This approach of information retrieval also gained ground by being applied in several discreet sources of information. This is done by enabling users to select available sources from their own by our system's search scopes and spaces forms. The resulting aggregation approach we follow is an assemblage of distributed resources that differs from the traditional ranking approaches of conventional search engines.

The end result was achieved by mashup separate resources (i.e., files, photos, videos and domain-specific information) hosted across multiple digital spaces in a single user interface. The applied actions for representing the results to the users were grouping the resources according to their host digital space along with summative visualizations.

The pilot demonstration took place by reference in recent activities of iSTLab laboratory related with the reorientation of organic farming to a digital practice, coded with possibilities offered by imbricating digital data representations and services.

6.2 – Address research questions

In this ground we answer the basic research question stated in chapter 1 about "*How does metadata-based Search perform compared to existing conventional search methods?*"

The interesting features of the presented search mechanism is not only that it bounds the search to specific digital spaces, but also the fact that it returns a collection of digital resources retained across boundaries, which could not be combined otherwise (i.e., by searching in each of those systems separately). Moreover, it searches through the distributed resources by using emergent categories (i.e., meta-tags) rather than pre-defined and static codes.

As a result, new opportunities for enhanced search capabilities become viable, while the new potentials for human routines emerge to complement the traditional setting. Such capabilities stem from the increased affordances to trace resources distributed across digital service boundaries.

Although the research is still on-going and empirical data on user experience are lacking, it becomes evident that digital trace data ascribe technologies with new capabilities that invoke novel ways people retrieve information in the Internet, not viable otherwise. This was prominently revealed in our case where retooling of the calendar led to new vocabularies (i.e., tags and meta-tags) which in turn rendered searchable the calendar events' digital resources retained in Flickr and YouTube, Google Drive, etc.

6.3 – Limitations and future work

As already discussed, this research brings about theoretical and engineering concerns that shift the focus of design from the conventional tool- or system-perspectives towards a more material orientation. The present research is a step in this direction, and by articulating the concept of mash-ups.

As for the limitations, firstly, we have investigated a specific domain, thus further work is needed covering other design cases. Also this work covered designated services, which however may not be the most appropriate for other design challenges; for instance collaborative music making would probably benefit from other digital services (e.g., Last.fm). Due to the above, and the focus of the thesis, we did not collect enough empirical validity, which is necessary for any attempt to consolidate a general-purpose methodology.

In addition to seeking empirical validity future studies and ongoing work could concentrate on applying the results in other domains of practice (i.e., creative arts, global virtual teams) and exploring additional technology genres (i.e., big data visual analytics) and infrastructures (i.e., Internet of Things).

References

1. O'reilly, T., *What is Web 2.0: Design patterns and business models for the next generation of software.* Communications & strategies, 2007(1): p. 17.
2. *Alexa: The top 500 sites on the web*. Available from: http://www.alexa.com/topsites.
3. Papadakis, I. and I. Apostolatos, *Mashups for web search engines,* in *Semantic Mashups*. 2013, Springer. p. 91-117.
4. Zhou, D., S. Lawless, and V. Wade, *Improving search via personalized query expansion using social media.* Information retrieval, 2012. **15**(3-4): p. 218-242.
5. Buijs, M., *Asynchronous Social Search,* in *Faculty of Science*. 2014, Utrecht University.
6. Lupu, M., M. Salampasis, and A. Hanbury, *Domain Specific Search,* in *Professional Search in the Modern World*. 2014, Springer. p. 96-117.
7. Gupta, M., et al., *An overview of social tagging and applications,* in *Social network data analytics*. 2011, Springer. p. 447-497.
8. Jacob, E.K., *Classification and categorization: a difference that makes a difference.* 2004.
9. De Chiara, R., A. Fish, and S. Ruocco. *Eulr: a novel resource tagging facility integrated with Flickr*. in *Proceedings of the working conference on Advanced visual interfaces*. 2008. ACM.
10. Liu, D., et al. *Tag ranking*. in *Proceedings of the 18th international conference on World wide web*. 2009. ACM.
11. Gabriel, H.-H., M. Spiliopoulou, and A. Nanopoulos, *Summarizing dynamic Social Tagging Systems.* Expert Systems with Applications, 2014. **41**(2): p. 457-469.
12. Evans, B.M. and E.H. Chi. *Towards a model of understanding social search*. in *Proceedings of the 2008 ACM conference on Computer supported cooperative work*. 2008. ACM.
13. Evans, B.M. and E.H. Chi, *An elaborated model of social search.* Information Processing & Management, 2010. **46**(6): p. 656-678.
14. McDonnell, M. and A. Shiri, *Social search: A taxonomy of, and a user-centred approach to, social web search.* Program, 2011. **45**(1): p. 6-28.
15. Guha, R., R. McCool, and E. Miller. *Semantic search*. in *Proceedings of the 12th international conference on World Wide Web*. 2003. ACM.
16. Fatima, A., C. Luca, and G. Wilson. *User experience and efficiency for semantic search engine*. in *Optimization of Electrical and Electronic Equipment (OPTIM), 2014 International Conference on*. 2014. IEEE.
17. Kiryakov, A., et al., *Semantic annotation, indexing, and retrieval.* Web Semantics: Science, Services and Agents on the World Wide Web, 2004. **2**(1): p. 49-79.
18. Fernandez, M., et al. *Semantic search meets the web*. in *Semantic Computing, 2008 IEEE International Conference on*. 2008. IEEE.
19. Berlanga, R., V. Nebot, and M. Pérez, *Tailored semantic annotation for semantic search.* Web Semantics: Science, Services and Agents on the World Wide Web, 2014.
20. Caramia, M., G. Felici, and A. Pezzoli, *Improving search results with data mining in a thematic search engine.* Computers & Operations Research, 2004. **31**(14): p. 2387-2404.

21. Li, Y., Z. Liu, and H. Zhu, *Enterprise Search in the Big Data Era: Recent Developments and Open Challenges.* Proceedings of the VLDB Endowment, 2014. **7**(13).
22. Zhu, H., et al. *Navigating the intranet with high precision*. in *Proceedings of the 16th international conference on World Wide Web*. 2007. ACM.
23. Hawking, D. *Challenges in enterprise search*. in *Proceedings of the 15th Australasian database conference-Volume 27*. 2004. Australian Computer Society, Inc.
24. Yu, S. and C.J. Woodard. *Innovation in the programmable web: Characterizing the mashup ecosystem*. in *Service-Oriented Computing–ICSOC 2008 Workshops*. 2009. Springer.
25. Hinchcliffe, D., *An executive guide to mashups in the enterprise.* JackBe Whitepaper, 2008.
26. Salminen, A. and T. Mikkonen. *Mashups-Software Ecosystems for the Web Era*. in *IWSECO@ ICSOB*. 2012.
27. Ennals, R. and D. Gay. *User-friendly functional programming for web mashups*. in *ACM SIGPLAN Notices*. 2007. ACM.
28. Lin, J., et al. *End-user programming of mashups with vegemite*. in *Proceedings of the 14th international conference on Intelligent user interfaces*. 2009. ACM.
29. Tuchinda, R., C.A. Knoblock, and P. Szekely, *Building mashups by demonstration.* ACM Transactions on the Web (TWEB), 2011. **5**(3): p. 16.
30. Yu, J., et al., *Understanding mashup development.* Internet Computing, IEEE, 2008. **12**(5): p. 44-52.
31. Huynh, D.F., R.C. Miller, and D.R. Karger, *Potluck: Data mash-up tool for casual users*. 2007: Springer.
32. Wong, J. *Marmite: Towards end-user programming for the web*. in *Visual Languages and Human-Centric Computing, 2007. VL/HCC 2007. IEEE Symposium on*. 2007. IEEE.
33. Wang, G., S. Yang, and Y. Han. *Mashroom: end-user mashup programming using nested tables*. in *Proceedings of the 18th international conference on World wide web*. 2009. ACM.
34. Huang, A.F., et al. *Improving end-user programming with situational mashups in web 2.0 environment*. in *Service-Oriented System Engineering, 2008. SOSE'08. IEEE International Symposium on*. 2008. IEEE.
35. Braga, D., et al., *Mashing up search services.* Internet Computing, IEEE, 2008. **12**(5): p. 16-23.
36. Rosenberg, F., et al., *Composing restful services and collaborative workflows: A lightweight approach.* Internet Computing, IEEE, 2008. **12**(5): p. 24-31.
37. Riabov, A.V., et al. *Wishful search: interactive composition of data mashups*. in *Proceedings of the 17th international conference on World Wide Web*. 2008. ACM.
38. Akoumianakis, D., et al. *Enacted virtual partnerships: Imbrications of representations of boundary artifacts and cloud services*. in *Information, Intelligence, Systems and Applications (IISA), 2013 Fourth International Conference on*. 2013. IEEE.
39. Watson-Manheim, M.B., K.M. Chudoba, and K. Crowston, *Perceived discontinuities and constructed continuities in virtual work.* Information Systems Journal, 2012. **22**(1): p. 29-52.
40. Markus, M.L., *Paradigm shifts-E-business and business/systems integration.* Communications of the Association for Information Systems, 2000. **4**(1): p. 10.

41. Monteiro, E., et al., *From artefacts to infrastructures.* Computer supported cooperative work (CSCW), 2013. **22**(4-6): p. 575-607.
42. Cummings, J.N., J.A. Espinosa, and C.K. Pickering, *Crossing spatial and temporal boundaries in globally distributed projects: A relational model of coordination delay.* Information Systems Research, 2009. **20**(3): p. 420-439.
43. Espinosa, J.A., et al., *Team boundary issues across multiple global firms.* Journal of Management Information Systems, 2003. **19**(4): p. 157-190.
44. Levina, N. and E. Vaaste, *The Emergence of Boundary Spanning Competence in Practice: Implications for Information Systems' Implementation Use.* Information Systems Working Papers Series, Vol, 2004.
45. Pawlowski, S.D. and D. Robey, *Bridging user organizations: Knowledge brokering and the work of information technology professionals.* MIS quarterly, 2004: p. 645-672.
46. Star, S.L. and J.R. Griesemer, *Institutional ecology,translations' and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39.* Social studies of science, 1989. **19**(3): p. 387-420.
47. Lutters, W.G. and M.S. Ackerman, *Beyond boundary objects: collaborative reuse in aircraft technical support.* Computer Supported Cooperative Work (CSCW), 2007. **16**(3): p. 341-372.
48. Subrahmanian, E., et al., *Boundary objects and prototypes at the interfaces of engineering design.* Computer Supported Cooperative Work (CSCW), 2003. **12**(2): p. 185-203.
49. Lee, C.P., *Boundary negotiating artifacts: Unbinding the routine of boundary objects and embracing chaos in collaborative work.* Computer Supported Cooperative Work (CSCW), 2007. **16**(3): p. 307-339.
50. Gasson, S., *A genealogical study of boundary-spanning IS design.* European Journal of Information Systems, 2006. **15**(1): p. 26-41.
51. Boujut, J.-F. and E. Blanco, *Intermediary objects as a means to foster co-operation in engineering design.* Computer Supported Cooperative Work (CSCW), 2003. **12**(2): p. 205-219.
52. Hunter, J. and W. Crawford, *Java servlet programming*. 2001: " O'Reilly Media, Inc.".
53. Akoumianakis, D., *Boundary spanning tactics and "traceable" connections in cross-organizational virtual alliances: A case study.* Journal of Enterprise Information Management, 2014. **27**(2): p. 197-227.
54. Bodle, R., *Regimes of sharing: Open APIs, interoperability, and Facebook.* Information, Communication & Society, 2011. **14**(3): p. 320-337.
55. Kopliku, A., K. Pinel-Sauvagnat, and M. Boughanem, *Aggregated search: a new information retrieval paradigm.* ACM Computing Surveys (CSUR), 2014. **46**(3): p. 41.
56. Akoumianakis, D., G. Ktistakis, and C. Michailidis, *Remediation by Design: new linguistic domains for changing Organizational Practices*, in *UK Academy for Information Systems* 2015, AIS: Oxford.