Search Computing for E-government

Alessandro Campi, Matteo Aletti, Yao Cheng, Edoardo Colombo,

Piero Franceschini, Mariangela Rizzo, Davide Sanapo

Politecnico di Milano
Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci 32,
20133, Milano, Italy
campi@elet.polimi.it,
[matteo.aletti|yao.cheng|edoardo.colombo|piero.franceschini|mariangela.rizzo|davide.sanapo]@asp-poli.it

***Abstract*—People shop online, compare online, book hotels and flights online. This happens because the data needed to complete these tasks are easily accessible, and a lot of Web sites allow users to query the Web to obtain enough information to be confident. The aim of this work is to propose a framework tailored to extend the internet revolution to public administration. This work is the first step towards an infrastructure allowing people to know in a very easy way the information they need. This paper exploits the Search Computing paradigm. It is a new way for composing data. While state-of-art search systems answer generic or domain-specific queries, Search Computing enables answering questions via a constellation of cooperating data sources, called search services, which are correlated by means of join operations. Search Computing aims at responding to queries over multiple semantic fields of interest; thus, Search Computing fills the gap between generalized search systems, which are unable to find information spanning multiple topics, and domain-specific search systems, which cannot go beyond their domain limits.**

Keywords-search; integration; query; e-government

I. INTRODUCTION

One hundred years ago, sending 10 words cost 50 dollars. Today gigabytes of information can be sent for a fraction of that cost, and huge quantities of information can be sent and received without it costing anything. This has changed the way people shop, the way people travel, the way people do business. The internet revolution has actually gone all the way through societies in many different ways, but it hasn't yet touched the way states are governed [1].

People shop online, compare online, book hotels and flights online. This happens because the data needed to complete these tasks are easily accessible and a lot of Web sites allow users to query the Web to obtain enough information to be confident. The aim of this work is to propose a framework tailored to extend the internet revolution to public administration. This work is the first step towards an infrastructure allowing people to know in a very easy way the information they need. People could search what operations work out properly, what records doctors have, the cleanliness of hospitals, who does best at infection control, etc.

Moreover, our work is a step forward to the transparency of public administration, and toward a real accountability of public services, and real awareness of citizens about their governments politics. The Missouri Accountability Portal [2] is an example of a Web portal that made available online all the data of one state in America. Every single dollar spent by that government is searchable, is analyzable, is checkable. Any business that wants to bid for a government contract can see what currently is being spent and possibly can offer to deliver it in a cheaper way.

Finally, our work allows one to relate objective data available online or offline with news articles, blog posts or other comments available online enriching objective data with information about the mood of people (as shown in Figure 1). In today's politics the need of a fast reaction to discontent is paramount and the possibility to relate numbers, facts, and sentiment analysis is very important.

Our proposal exploits the Search Computing paradigm [3]. It is a new way for composing data. While state-of-art search systems answer generic or domain-specific queries, Search Computing enables answering questions via a constellation of cooperating data sources, called search services, which are correlated by means of join operations. Search Computing aims at responding to queries over multiple semantic fields of interest; thus, Search Computing fills the gap between generalized search systems, which are unable to find information spanning multiple topics, and domain-specific search systems, which cannot go beyond their domain limits. Paradigmatic examples of Search Computing queries are: "Where is the school closest to my home, offering a high teaching quality and a good food service?", "Who is the best doctor who can cure insomnia in a nearby public hospital?", "Which are the highest risk factors associated with the most prevalent diseases among the young population?". These queries cannot be answered without capturing some of their semantics, which at minimum consists in understanding their underlying domains, in routing appropriate query subsets to each domain-expert search engine, and in combining answers from each engine to build a complete answer that is meaningful for the user.

A prerequisite for setting such goal is the availability of a large number of valuable search services. We could just

 ISBN: 978-1-61208-287-5

wait for SOA (Service Oriented Architecture) to become widespread. However, in the public administration scenario, very few data are offered by services designed to support search, and, moreover, a huge number of valuable data sources are not provided with a service interface. In this paper, we tackle the important issue of publishing service interfaces suitable for Search Computing so as to facilitate the widespread use of data sources on the Web and to simplify their integration in Search Computing applications.

The goals of our proposal is to create a virtuous system where the Public administration shares its data with citizens (increasing the transparency of the government and the trust in the public administration). Citizens analyze the available data to learn something but also to provide hints to the public administration helping to offer better services.

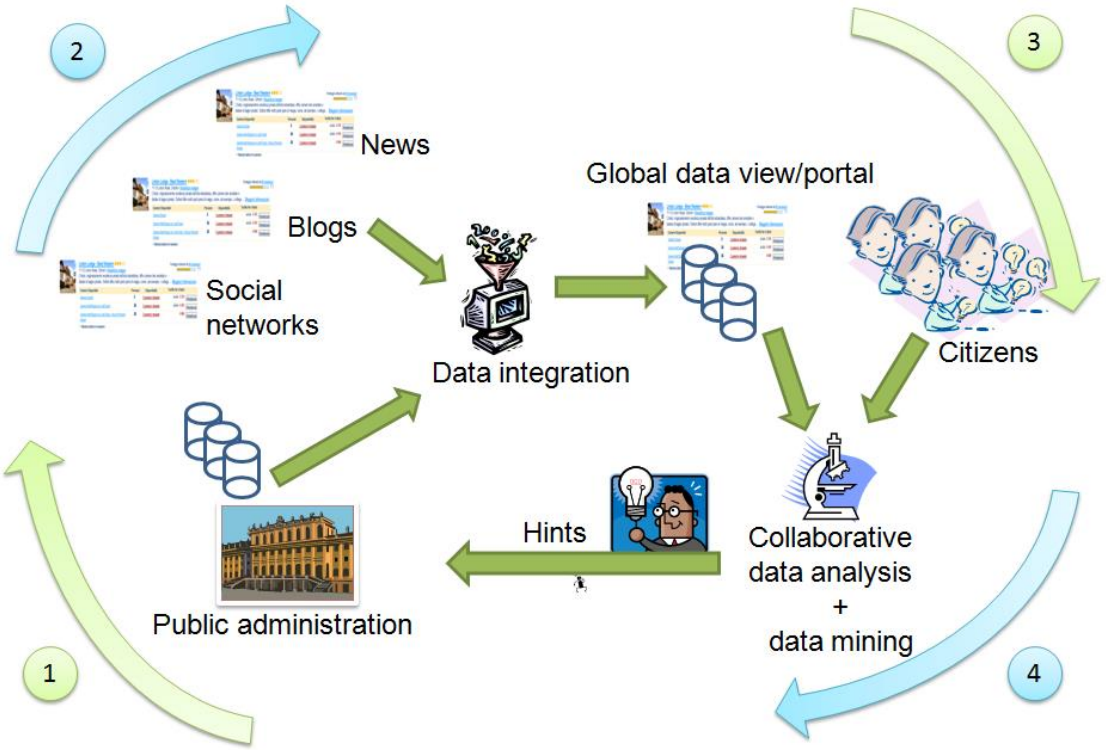

Figure 1: The data life cycle

II. THE FRAMEWORK

Figure 2 shows an overview of our framework, composed of several sub-frameworks. As shown in the figure, the data framework provides the scaffolding for wrapping and registering data sources. The core concept of the data framework are Service Marts. The user framework provides functionality and storage for registering users, with different roles and capabilities. The query framework supports the management and storage of queries: a query can be executed, saved, modified, and published for other users to see. The execution framework is responsible of an efficient execution of the previously defined query plan. Finally, the invocation framework masks the technical issues involved in the interaction with the data sources, e.g., the Web service protocol and data caching issues.

The core of the framework aims at executing multi-domain queries. The query manager takes care of splitting the query into sub-queries (e.g., "Who is the best doctor that cures insomnia?" ; "Who are the doctors of a certain hospital?"; "Which hospital is close to my house?") and bounding them to the respective relevant data sources registered in the data framework repository (in this case, hospitals could be retrieved using GeoPlaces, places close to my home using Google Maps, …); starting from this mapping, the query framework produces an optimized query execution plan, which dictates the sequence of steps for executing the query. Finally, the execution framework actually executes the query plan, by submitting the service calls to designated services through the service invocation framework, building the query results by combining the outputs produced by service calls, computing the global ranking of query results, and producing the query result outputs in an order that reflects their global relevance.

To obtain a specific application, this general-purpose architecture is customized by users, supported by appropriate design tools. The development process involves users with different roles and expertise (see Figure 3).

 ISBN: 978-1-61208-287-5

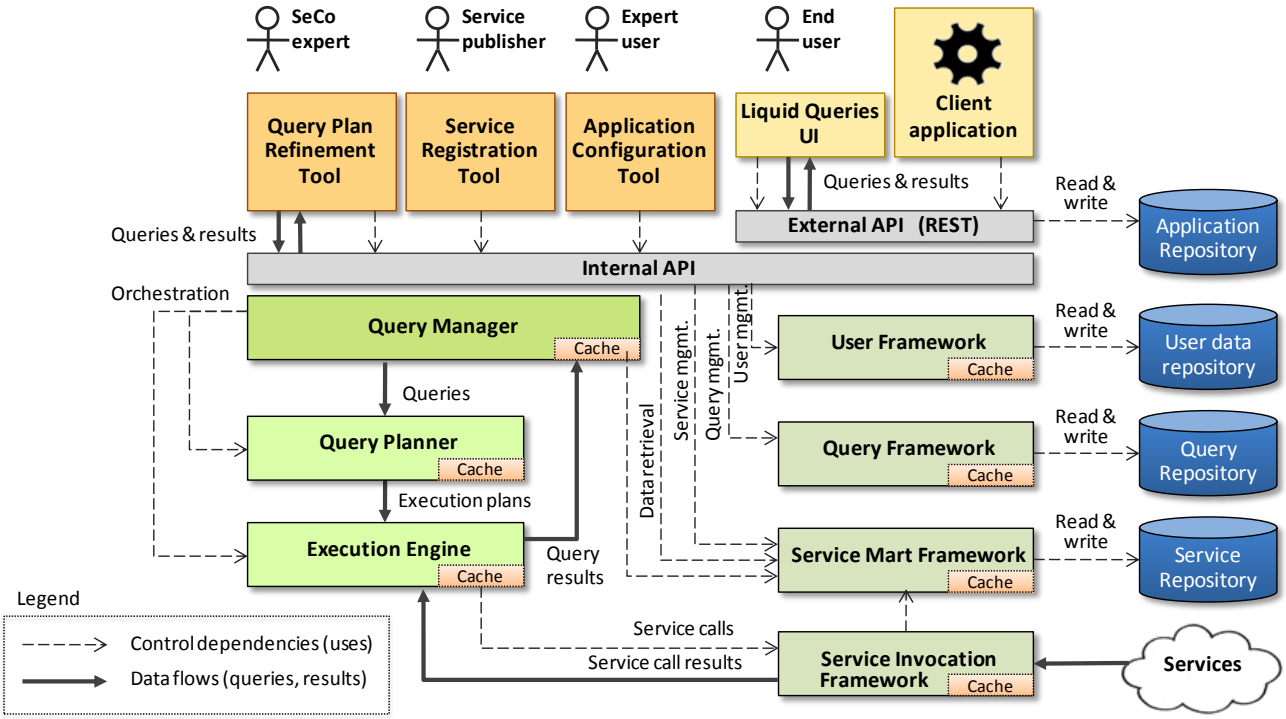

Figure 2. The framework (more details in [3])

Data Publishers: they are in charge of implementing mediators, wrappers, or data materialization components, so as to make data sources compatible with the service mart standard interface and expected behavior; they register service mart definitions within the service repository, and declare the connection patterns usable to join them.

Expert Users: they configure Search Computing applications, by selecting the service marts of interest, by choosing a data source supporting the service mart, and by connecting them through connection patterns. They also configure the user interface, in terms of controls and configurability choices for the end user.

End Users: they use Search Computing applications configured by expert users. They interact by submitting queries, inspecting results, and refining/evolving their information need according to an exploratory information seeking approach, which we call Liquid Query [4].

The development process steps lead to the final application accessed by the end user. The Liquid Query interface, instantiated during the application configuration phase, supports the "search as a process" paradigm, based on the continuous evolution, manipulation, and extension of queries and results; the query lifecycle consists of iterations of the steps of query submission, when the end user submits an initial liquid query; query execution, producing a result set that is displayed in the user interface; and result browsing, when the result can be inspected and manipulated through appropriate interaction primitives, which update either the result set (e.g., re-ranking or clustering the results) or the query (e.g., by expanding it with additional service marts or requesting for more results). This approach to development takes into account the trend towards empowerment of the user, as witnessed in the field of Web mash-ups [5]. Indeed, all the design activities from service registration on do not ask to perform low-level programming.

III. CONCLUSIONS AND FUTURE WORKS

This proposal is a step forward towards the possibility to exploit all the information we have about government and society for a better understanding of the actual situation. This work exploits the Search Computing paradigm in order to allow people to relate objective data available online or offline with news articles, blog posts or other comments available online enriching objective data with information about the mood of people.

In the future, we envision the spreading of software components and methodologies for e-government. In particular, we would like to study how to make the search of data related to e-government as easy as a Google (or Yahoo! or Bing) query.

IV. ACKNOWLEDGMENTS

This work was partially supported by Search Computing ERC Grant.

 ISBN: 978-1-61208-287-5

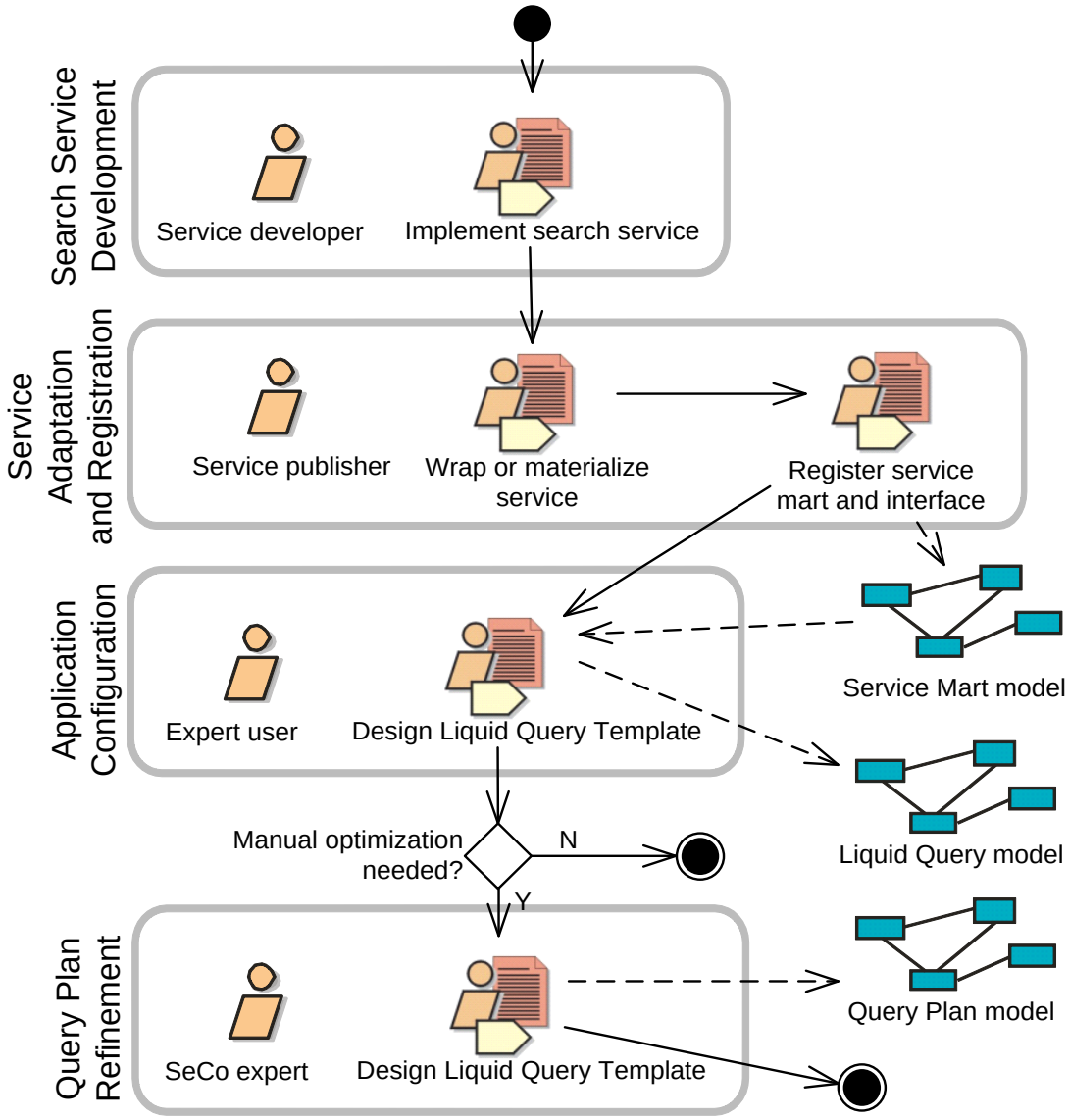

Figure 3. Users (more details in [4])

V. REFERENCES

[1] David Cameron. The next age of government. http://blog.ted.com/2010/02/16/the_next_age_of/ (accessed 1/2/2011)

[2] Map your taxes. http://mapyourtaxes.mo.gov/MAP/Portal/Default.aspx (accessed 1/2/2011)

[3] Ceri, S., Brambilla, M. (eds.). Search Computing Challenges and Directions. Springer LNCS vol. 5950, March 2010.

[4] A. Bozzon, M. Brambilla, S. Ceri, P. Fraternali. Liquid query: multi-domain exploratory search on the web. In *Proc. of the 19th int. conf. on World wide web* (WWW '10). ACM, New York, USA, 161-170.

[5] Braga, D., Ceri, S., Daniel, F., Martinenghi, D. Mashing Up Search Services. IEEE Internet Comp. 12(5) (2008), 16-23.

 ISBN: 978-1-61208-287-5