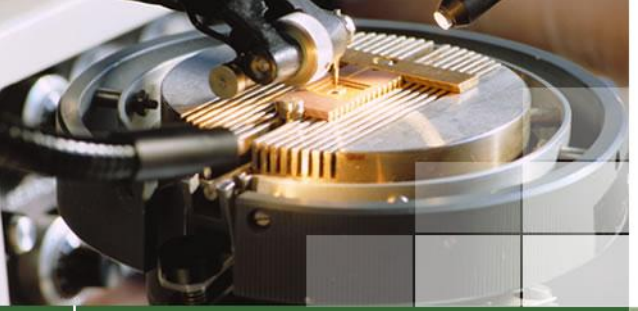


POLITECNICO DI MILANO

Dipartimento di
Elettronica e Informazione

Search Computing @ WebDam Workshop

Stefano Ceri, Politecnico di Milano

(the land of ERC grants in CS)

Motivating Examples

- “Who are the strongest candidates in Europe for competing on software ideas?”
- “Who is the best doctor who can cure insomnia in a close-by hospital?”
- “Where can I attend an interesting scientific conference in my field and at the same time relax on a beautiful beach nearby?”

This information is available on Internet, but no software system is capable of computing the answer.

Queries span over **multiple semantic domains** and require **composing ranking of results.**

Search Computing

- **Search computing** is a *new multi-disciplinary science which will provide the abstractions, foundations, methods, and tools required to give answer multi-domain queries on the Web*
- Emphasis on:
  - **Search services**. These are software services producing ranked information. Ranking is essential for search service composition.
  - **Search Integration.** The objective is not to build new search systems but instead to integrate a world-wide network of search systems.
- **Web site: www.search-computing.org**

Research Chapters of Search Computing, 1

- **Foundational theories**, rooted into formal disciplines such as mathematics and optimization theory.
- **Statistical models** for estimating the number and qualities of the results produced by a search service.
- **Optimization methods** for determining efficient plans for service integration
- **Software paradigms** for designing and constructing search computing systems.
- **Interaction paradigms** to help user-friendly expression of queries and ranking.
- **Framework** for search-oriented software architectures and their instrumentation.

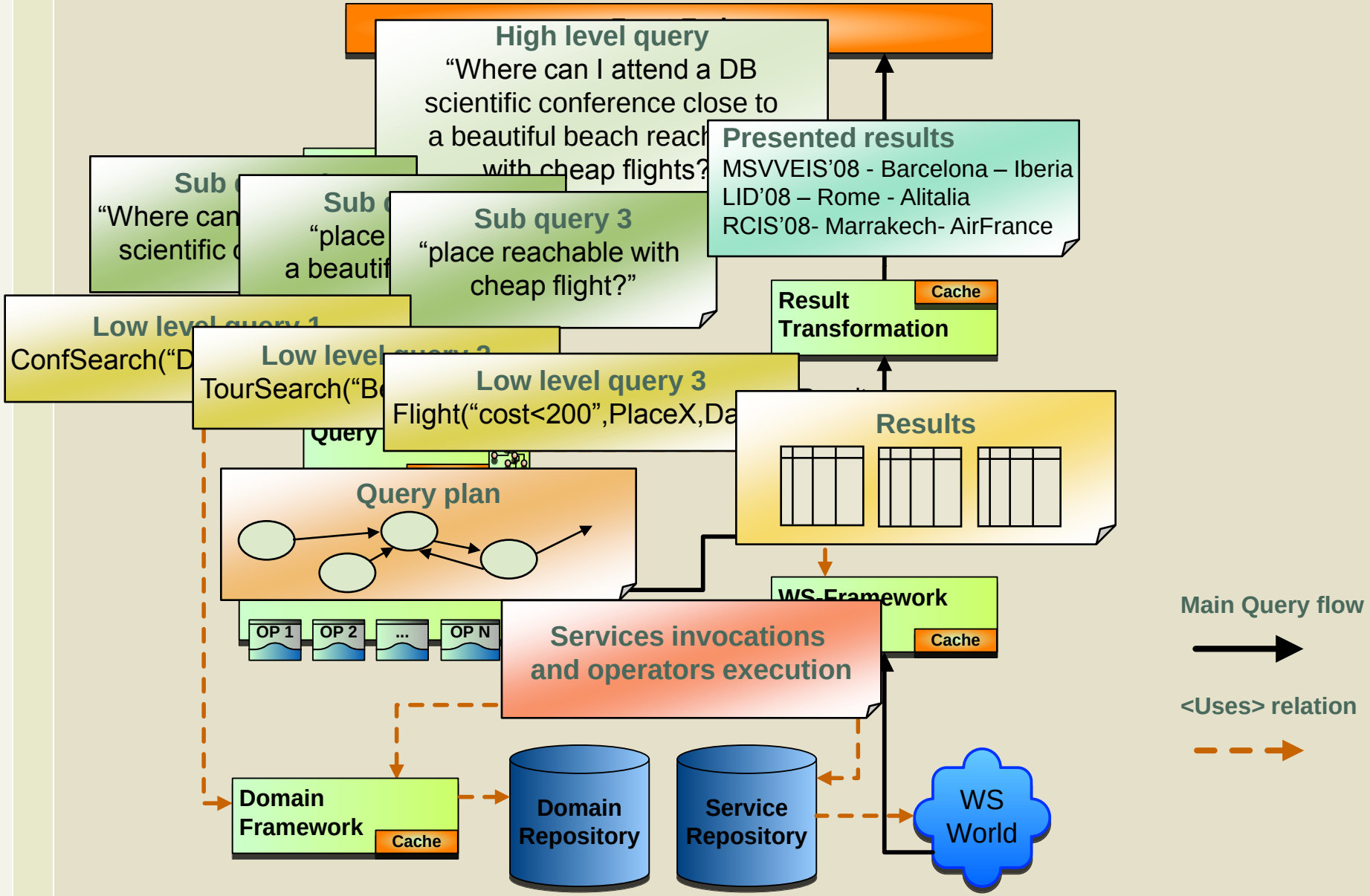
Research Chapters of Search Computing, 2

- **Semantic domain knowledge** for dealing with terminological aspects in composing search engine results.
- **Higher-order rankings** for prioritizing search objects and services.
- **Personal and social aspects** for setting ranking in relationship to individuals and context.
- **Business models** for pushing and developing search computing economy.
- **Legal and privacy issues** concerned with search sources integration.
- **Advanced computational architectures** for search computing.

Results before Search Computing

- **FUNDING (2007-08): PRIN NGS (New Generation Search)**
  - **Politecnico Milano (National Coordination)**
  - **University of Roma 3**
  - **Free University of Bolzano**
- The brick: **join of two search services**
  - Information Systems, March 2008
- The framework: **multi-domain query optimization**
  - International Very Large Data Bases Conference, Auckland (NZ), August 2008
- The interface: **mash-up based interaction**
  - IEEE-Internet Computing, November 2008
- Optimality: **top-K extraction in rank aggregation**
  - Currently submitted

Search Computing paradigm: overall view



Search Computing architecture: incremental prototyping

Prototype 4:
High level queries

Prototype 3:
Mapping and presentation

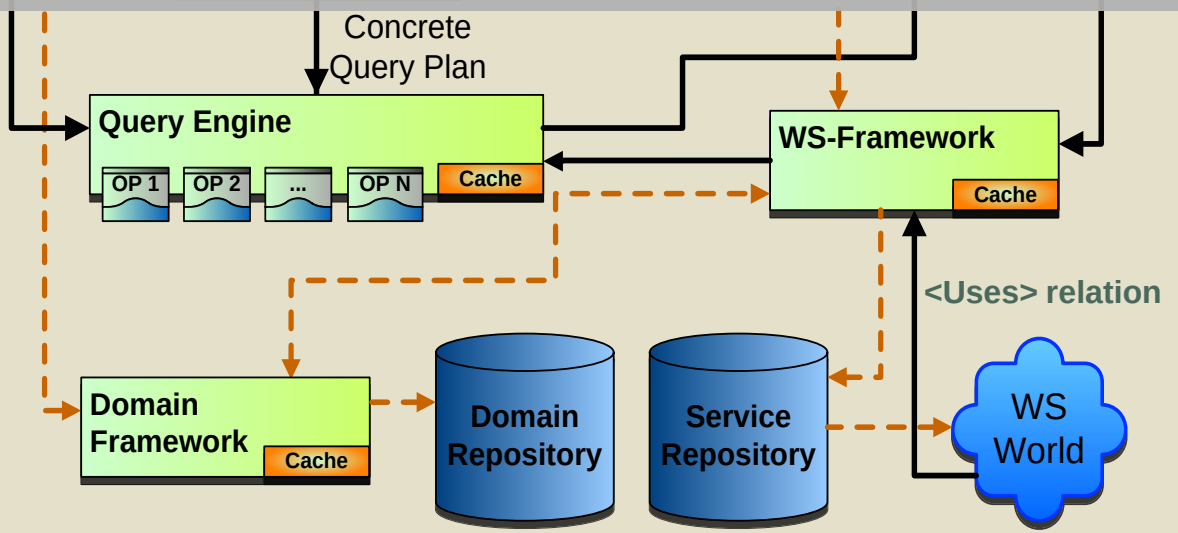
- mapping to domains
- presentation of results

Prototype 2:
Planning

- Automatic optimized query planning

Prototype 1:
Core behaviour of the system.

- Engine-based execution of queries
- Domain repository
- Service repository
- Coarse result presentation



Search Computing at month 6

- **Search is a very competitive arena**
  - Just to name a few newcomers: Bing, Wolfram-Alpha, Kosmics
- **Academic research in search is hard**
  - Even simple ideas require lots of investments to be proven
- After **initial brainstorming and lots of thinking**, our current approach is to stay away from core research in:
  - Global indexing and crawling
  - Semantic web
  - Communities

  ... and instead **focus on our strength**:
  - Data management
  - Query optimization and execution (on scalable architectures)
  - Software/service technologies and tools
  - Process modelling and mining

Reasoning about Search 09

- **UNIVERSAL APPROACHES**
  - Indexing + global page ranking: Google
  - Classification: Yahoo, Bing
  - Semantics: Wolfram-Alpha
- **DOMAIN-EXPERT APPROACHES**
  - **Fixed horizontal composers**: Kosmics – broadcasts the same query to multiple engines and collate results.
  - **Domain-specific meta searchers**: Tuifly – broadcasts the same query to multiple engines, collects and ranks results.
  - **Fixed vertical composers**: Expedia – given known compositional patterns between flights, hotels, cars, travel-related events –broadcasts modified queries to data sources, collects, composes and ranks results.
  - **Search computing systems**: extending the compositional pattern used by fixed vertical composers Expedia in a very specific context (travels) to arbitrary contexts, with multiple domains, many search engines, and greater query variance, but with known composition methods.

What are the assets of search computing?

- A **standard model for search services** (service-mart), with almost-flat representation and with suitable parameters for computing query cost/time.
- A **registration strategy** consisting of providing, for each pair of services and each composition semantics, a “composition set-ups” (service properties that should be compared).
- A **standard model for composition**, based on the notion of join between web services, and **several composition operations (join methods)** for associating a query with execution strategies.
- A **query optimization strategy**, consisting of methods for determining a plan, i.e. selecting the involved services, inferring the compositional semantics to be used, and determining the best composition operations.
- A **service scheduling environment**, consisting of methods for executing a plan, i.e. iterating service invocation, computing compositions, evaluating global rankings, determining stop conditions, caching results, enabling backtracking and recomputing.
- **Liquid presentation of query results**, enabling browsing of results and sophisticated controls for, e.g. asking more results, rolling up, drilling down, augmenting the query in given dimensions.

Current focus: Simplified instantiations of search computing

- Parametric query, fixed choice of services, fixed composition.
  - E.g.: “best trips for a soccer supporter who wants to follow the team on a road game to a given “city” and also find in the city of the game a good hotel, cheap and fast rountrip transports, and “rock music” event within “2” days from the game.”
- Composable query, variable choice of services, fixed composition once services are chosen.
  - E.g.: queries allowing users to focus their interests on offers in june about monuments, sport events, concerts, museums, hiking trails, beaches, fairs,… thus finding a city or area within Europe where the top offers matching their interests are present, and at the same time there is an affordable option for roundtrip and hotel stay in the city or area, and good average climate in june.

From queries to processes

Start from a multi-domain query:

- Search for the best movie-theatre combination where the movie must be an action movie (ranked by stars) and the theatre must be close to home.

and then …

- Once several candidate movies are located, look for their actors, their directore, other films directed by that director, and so on…
- Once several candidate theatres are located, look for close-by pizzeria, for transportation, for parkings…

with search options…

- enable a user to dynamically impose search ordering (first choose the movie then the theatres)
- enable backtracking (if theatre location is not satisfactory after investigation go back and change theatre)

From queries to goals

If a query is being asked what does the user really want?

- Composition set-ups can give the most likely directions of extension of the current query
  - These can be observed/mined from multiple query instantations and then suggested in “ranking order”
- Disaggregation of global rankings and association with results can suggest the most promising direction of improvement for the user:
  - The one with fewer answers
  - The one whose ranking in the answers did not drop much
- Disaggregation of the query + results by services may enable inspecting/changing one at a time
  - Exploring the search space by steps, with a sort of “pivotal” exploration (at every new search, a new dimension goes on focus but the dimension previously on focus is fixed)

Future Focus (far away)
Search Computing in the Universe

- An “exploratory” approach to search computing, starting from NL queries and combining:
  - Lightweight semantics (inspired by Wordnet) for service description
  - Open-source NLP queries and processors for query analysis and decomposition (aiming at using a mix of syntactic methods and of light semantic annotations for mapping sentence chunks to domains of interests)
  - Distance-based and clustering methods for mapping queries to services.
- Will measure distance between “discovered mappings” and “intended mappings” while the query broadens to enlarge more and more domains.
- Will enable us to understand the pros and cons of a service-oriented approach to search in a global sense

The next six months

- Foundations, foundations, foundations!
  - Service marts: motivation, theory, design, source wrapping, materialization and incremental maintenance
  - Join methods: theory, efficient implementation, bio-inspired methods
  - Optimization: plan selection through decision trees, strategy analysis and comparison
  - Execution (panta rhei): producer-consumer system with service scheduling, context, caching, run-time controls
  - Interfaces: liquid queries and liquid results

  Together with a fast prototyping attitute and the objective of delivering the core of the technology in the next six monts

- Software engineering methods and tools for search computing

  For enabling an advanced user to deploy search-computing apps with few clicks

- Human-computer interaction for search computing

  For enabling end-user navigation on result combinations

Search Computing Workshop, June 17-19, 2009

First Day

09:00 - 09:30 Registration

09:30 - 10:30 Introduction to the two ERC projects

Stefano Ceri (Politecnico di Milano) - SECO: Search Computing

Carlo Ghezzi (Politecnico di Milano)- SMSCom: Self-Managing Situated Computing

10:30 - 11:00 Coffee break

11:00 - 12:30 Joint ERC presentations

Jeff Magee (Imperial College, UK) - Engineering Self-Managed Systems

Ricardo Baeza-Yates (Yahoo! Research, Barcelona) - The Next Generation of Search

12:30 - 14:00 Lunch

14:00 - 15:45 Rank Aggregation

Ihab Ilyas (University of Waterloo): Supporting Ranking in (Uncertain) Database Systems

Davide Martinenghi (Politecnico di Milano): Cost-Aware Top-K Join Algorithms

Adnan Abid (Politecnico di Milano): Evolutionary Techniques for Join Strategies

15:45 - 16:15 Coffee break

16:15 - 17:45 Service Registration and WebSite Wrapping

Georg Gottlob (Oxford University): Web Data Extraction: The Lixto Project and Future Plans

Alessandro Campi (Politecnico di Milano): Service Marts: a Service Framework for Search Computing

Second Day

09:00 - 10:30 Joint ERC presentations

Norman Paton (University of Manchester): Dataspaces and Search Computing: New Paradigms or Step Changes

Dick Taylor (UC Irvine, USA): Runtime software adaptability

10:30 - 11:00 Coffee break

11:00 - 12:45 Service composition

Fabio Casati (University of Trento): Universal Mashup Languages

Daniele Braga (Politecnico di Milano): Join methods and query planning for Search Computing

Michael Grossniklaus (Politecnico di Milano): A producer-consumer model of query execution

12:45 - 14:00 Lunch

14:00 - 15:30 Search computing architectures

Ioana Manolescu (INRIA, PARIS): Of Distributed Queries: Models and Systems

Marco Brambilla (Politecnico di Milano): Liquid queries

Evening: Joint ERC recreational activities: a walk on the hills around Como/Brunate. The idea is to reach Brunate by funicular (http://www.funicolarecomo.it/), hike from Brunate to the top of Monte Boletto, which enjoys a great view of the lake. We will have dinner at:

LOCANDA DEL DOLCE BASILICO, Mulattiera s. Maurizio 24, 22034 - Brunate (CO)

Third Day

09:00 - 10:30 Business Plan and Technology Watch

Tommaso Buganza (DIG, Politecnico di Milano) and Emanuele Della Valle (DEI, Politecnico di Milano): Technology Watch for Search Computing

10:30 - 11:00 Coffee break

11:00 - 12:30 Priorities in SeCo Research

Moderator: Piero Fraternali (Politecnico di Milano)

12:30 - 14:00 Lunch

14:00 - 15:00 Tutorial

Florian Daniel (University of Trento): Innovation in Web Technology

15:00 – 15:30 Coffee Break

15:30 – 17:00 *Stefano Ceri (Politecnico di Milano):* Workplan, Workshop Conclusions and Wrap-up

Search Computing Challenges and Directions (LNCS, Ceri-Brambilla eds)

Part 1: Vision

- Ceri: Search computing
- Baeza-Yates: Next generation search
- Weikum: Search for knowledge

- Part 2: Technology Watch for Search Computing
  - Dellavalle-Buganza-Gatti: The search engine industry
  - Casati-Daniel-Soi: Mashup technologies
  - Baumgartner-Campi-Gottlob-Herzog: Web data extraction
  - Hedeler-Belhajjame-Campi-Embury-Fernandez-Paton:Dataspaces
  - Bozzon-Fraternali: Multimedia and multimodal information retrieval

- Part 3: Issues in Search Computing
  - Campi-Ceri-Gottlob-Ronchi: Service marts
  - Braga-Campi-Grossniklaus: Join methods and query optimization
  - Ilyas-Martinenghi-Tagliasacchi: Rank aggregation
  - Braga-Grossinklaus-Ceri: Panta Rhei, a query execution environment
  - Brambilla-Ceri-Fraternali-Manolescu: Liquid queries and liquid results
  - Brambilla-Ceri: Software engineering of search computing applications
  - Masseroli-Paton-Spasic: Search computing and the life sciences

SeCo Teams

- **Theory and Methods** (**Davide Martinenghi**, Marco Tagliasacchi). Design of solid methods (with known performance and guarantees) returning top-k query results.
- **Service Registration and Management** (**Alex Campi**, Stefania Ronchi, Andrea Maesani). Registration of new search services, their semantic description, and the production of relevant parameters. We envision informal and quick deployment and registration of search services.
- **Query Processing and Execution Engine** (**Daniele Braga**, Michael Grossnicklaus, Davide Barbieri, Adnan Abid, Mahmoud Abu Helou, several Ms Students, Francesco Corcoglioniti, Ioana Manolescu). Operation-based model of SeCo enabling the mapping of user queries into execution plans and the selection and execution of "optimal" execution plans.

SeCo Teams

- **Tools** (**Marco Brambilla**, Alessandro Bozzon, several Ms students). Developer-oriented and user-oriented tools, demonstrators of the "current" technology throughout the project.
- **Business Models and Technology Watch** (**Emanuele Della Valle**, Roberto Verganti, Tommaso Buganza, Nicola Gatti, Sofia Ceppi). Setting the strategic directions of the project, offering a "technological watch" and then discussing "scenarios" that can lead to better results from all perspectives, including business.
- **Interaction Design** (**Piero Fraternali**, Sara Comai, Maristella Matera, Davide Mazza). Paradigms for improving interaction, design of effective feedback methods for involving users in producing the answer to queries.
- **Concept Team** (**Stefano Ceri** and all the team coordinators). Coordinating the project, deciding planning and milestones, deciding about technological standards, and integrating the various parts. Manage resource allocation, including human resources.

Interesting problems for Web-Dam?

- You are too smart and will figure out yourselves!
- But there are a few observations to make:
  - Search is no longer just about crawling & indexing
  - Search for complex information requires complex data (VLDB panel 3 days ago)
  - Search is opening to the “long tail” of good-quality content available on the Web (with suitable data extraction and servic-ization)
  - Search is converging with process modeling (people want to do tasks) and with social networks (wisdom of the crowds)
  - Search is opening to data mining (discover & anticipate hidden goals)

问题?

Last-minute

- Should WebDam try to integrate search within its scope?
  - Search theory has a probabilistic flavor (e.g. clustering, “closest node”, “best recommendation”) and WebDam seems (to me) to ignore these topics.

    Can search theory find its home in WebDam?
  - In our approach, search requires a number of transformations for bulding alternative formulations upon data sources, that in the end turn to queries.

    Can transformations be formalized?