Collaborative Business Intelligence

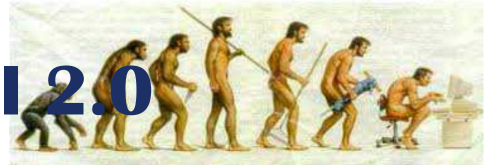
Stefano Rizzi

University of Bologna - Italy

Summary

- The challenges of BI 2.0
- Approaches to collaborative BI
  - Warehousing approaches
  - Federative approaches
  - P2P approaches
- A new approach: *Business Intelligence Networks*
  - Motivating scenario and envisioned architecture
  - Research issues
  - A mapping language
  - Query reformulation
- Summary and open issues

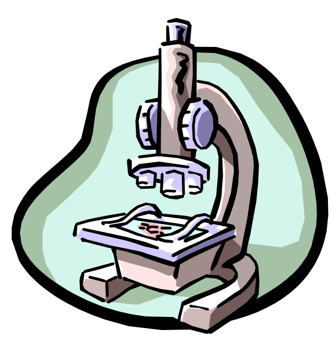
From BI 1.0 to BI 2.0



- *Business intelligence* (BI) transformed the role of computer science in companies from a technology for storing data into a discipline for timely detecting key business factors and effectively solving strategic decisional problems
- In the current changeable and unpredictable market scenarios, the needs of decision makers are rapidly evolving
- To meet the new, more sophisticated user needs, a new generation of BI systems (BI 2.0) has been emerging

Issues in BI 2.0

- BI as a service
- Real-time BI
- Situational BI
- Collaborative BI
- Pervasive BI
- ....



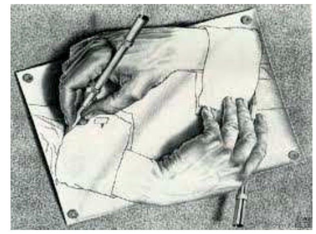
Motivation

- In a distributed business scenario, where multiple partner companies/organizations cooperate towards a common goal, traditional BI systems are no longer sufficient to maximize the effectiveness of decision making processes
- Two main requirements arise:
  - **Cross-organization monitoring and decision making**
    Accessing local information is no more enough, users need to transparently and uniformly access information scattered across several heterogeneous BI platforms
  - **Pervasive and personalized access to information**
    Users require that information can be easily and timely accessed through devices with different computation and visualization capabilities, and with sophisticated and customizable presentations

Collaboration [Wiki]

- Collaboration is working together to achieve a goal
  - It is a recursive process where two or more people or organizations work together to realize shared goals —this is more than the intersection of common goals, but a deep, collective, determination to reach an identical objective— by sharing knowledge, learning and building consensus
- Most collaborations require leadership
- Teams that work collaboratively can obtain greater resources, recognition and reward when facing competition for finite resources

Collaborative BI



- Collaboration is seen today by companies as one of the major means for increasing flexibility and innovating so as to survive in today uncertain and changing market
- Companies need strategic information about the outer world, for instance about trading partners and related business areas
- Users need to access information anywhere it can be found, by locating it through a semantic process and performing integration on the fly
- This is particularly relevant in inter-business collaborative contexts where companies organize and coordinate themselves to share opportunities, respecting their own autonomy and heterogeneity but pursuing a common goal

But...

- ...most information systems were devised for individual companies and for operating on internal information, and they give limited support to inter-company cooperation
- ...traditional BI applications are aimed at serving individual companies, and they cannot operate over networks of companies characterized by an organizational, lexical, and semantic heterogeneity

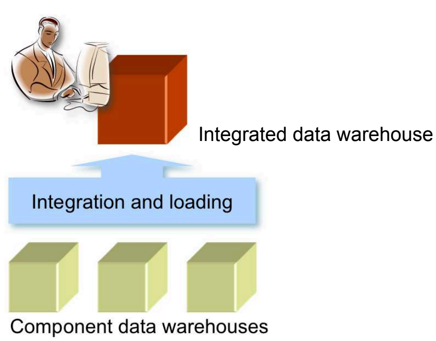
need for innovative approaches and architectures

Data warehouse integration

- Data warehouse integration is an enabling technique for collaborative BI; it provides a broader base for decision-support and knowledge discovery than each single data warehouse could offer
  - Large corporations integrate their separately-developed departmental data warehouses
  - Newly merged companies integrate their data warehouses into a central data warehouse
  - Autonomous but related organizations join together their data warehouses to enforce the decision making process

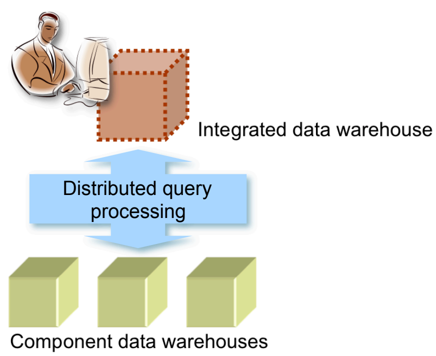
Approaches

- Warehousing approaches
  - all components to be integrated share the same schema, or a global schema is given
  - the integrated data are physically materialized



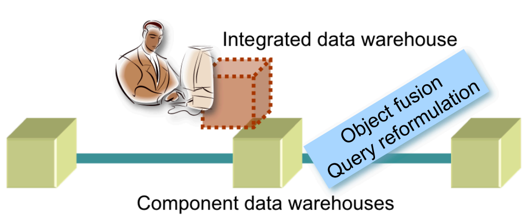
Approaches

- Federative approaches
  - all components to be integrated share the same schema, or a global schema is given
  - integration is virtual



Approaches

- P2P approaches
  - they do not rely on a global schema to integrate the component data warehouses
  - necessary in contexts where the different parties have a common interest in collaborating while fully preserving their autonomy and their view of business
  - each peer can formulate queries also involving the other peers, typically based on a set of mappings that establish semantic relationships between the peers' schemata



A look at the literature

- OLTP
  - In data exchange, data structured under one source schema must be restructured and translated into an instance of a different target schema, that is materialized (Fagin et al., 2003)
  - In data integration systems, data from different sources are combined to give users a virtual unified view (Lenzerini, 2002); in this case query processing requires a reformulation step
  - Peer Data Management Systems (PDMSs) have been proposed as architectures to support sharing of operational data across networks of peers while guaranteeing peer autonomy, based on *semantic mappings* that mediate between the heterogeneous schemata exposed by peers (Halevy et al., 2004)

A look at the literature

- OLAP
  - In this context, problems related to data heterogeneity are usually solved by ETL processes that load data into a single multidimensional repository, but some works are specifically focused on strategies for data warehouse integration and federation:
    - Check of *coherence*, *soundness*, and *consistency* of two dimensions (Torlone, 2008)
    - *CubeStar* (Albrecht & Lehner, 1998); *Skalla* (Akinde et al, 2003); *Multi data warehouse* (Berger & Schrefl, 2006)
    - Discovery of inter-hierarchy mappings (Banek et al., 2008)
    - P2P warehousing of XML content (Abiteboul, 2003)
    - Multidimensional P2P network and OLAP query rewriting (Vaisman, 2009)
    - Semantic layer to enable communication among different components (Kehlenbeck & Breitner, 2009)

Business Intelligence Networks

a joint work with Golfarelli, Mandreoli, Penzo, Turricchia

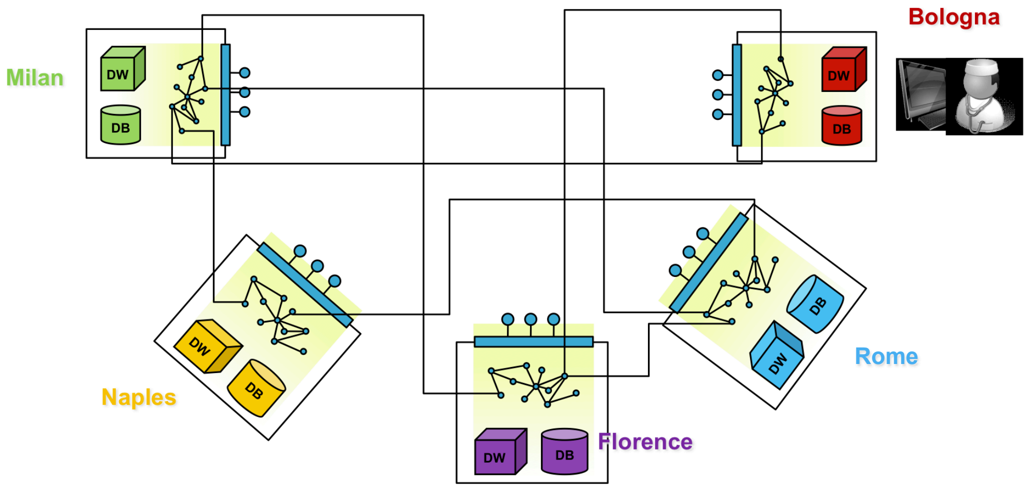
- *Business Intelligence Network* (BIN):
  an architecture for sharing BI functionalities across a dynamic and collaborative network of heterogeneous and autonomous peers
  - Each peer is equipped with an independent BI system, that relies on a local multidimensional schema to represent the peer's view of the business and exposes decision support functionalities aimed at sharing business information
- Main benefits to the corporate world:
  - enhance the decision making process and create new knowledge
  - build new inter-organizational relationships and coordination approaches
  - efficiently manage inter-company processes and safely share management information

Business Intelligence Networks

- Features:
  1. Users transparently access business information distributed over the network in a pervasive and personalized fashion
  2. Access is secure, depending on the access control and privacy policies adopted by each peer
  3. Participants are collaborative, even if with different grades
  4. Inclination to collaboration does not reduce autonomy of participants, who are not subject to a shared schema
  5. A BIN is decentralized and scalable because the number of participants, the complexity of business models, and the workload can change

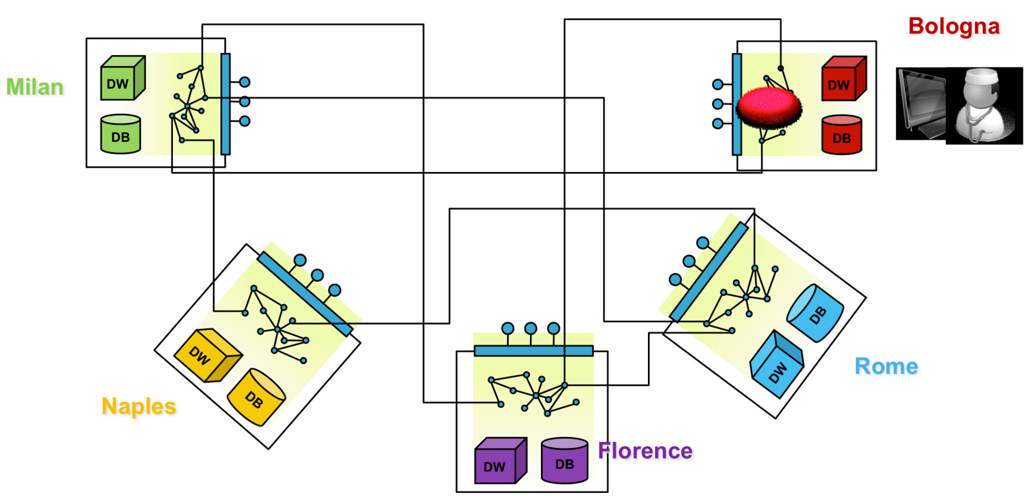
A typical user interaction

A user formulates an OLAP query *q* by accessing the local multidimensional schema of her peer, *p*



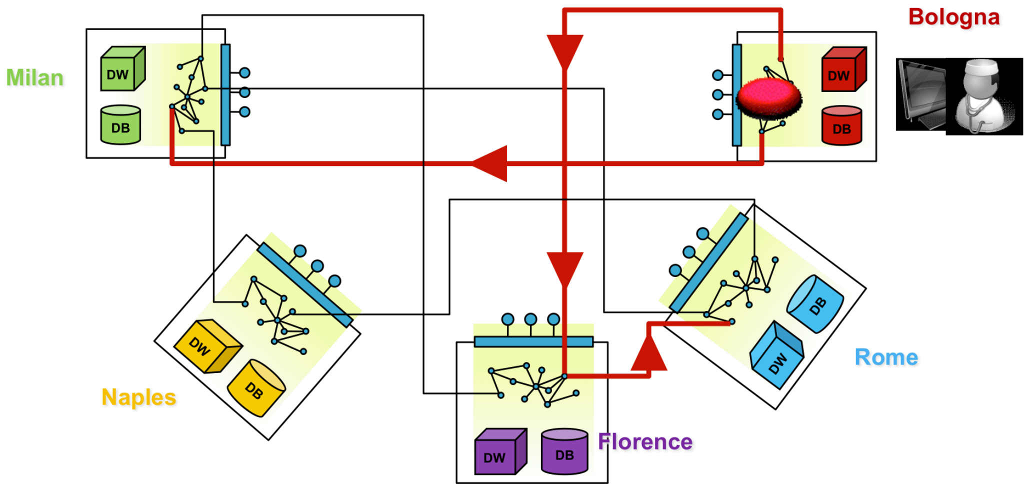
A typical user interaction

Query *q* is processed locally on the data warehouse of *p*



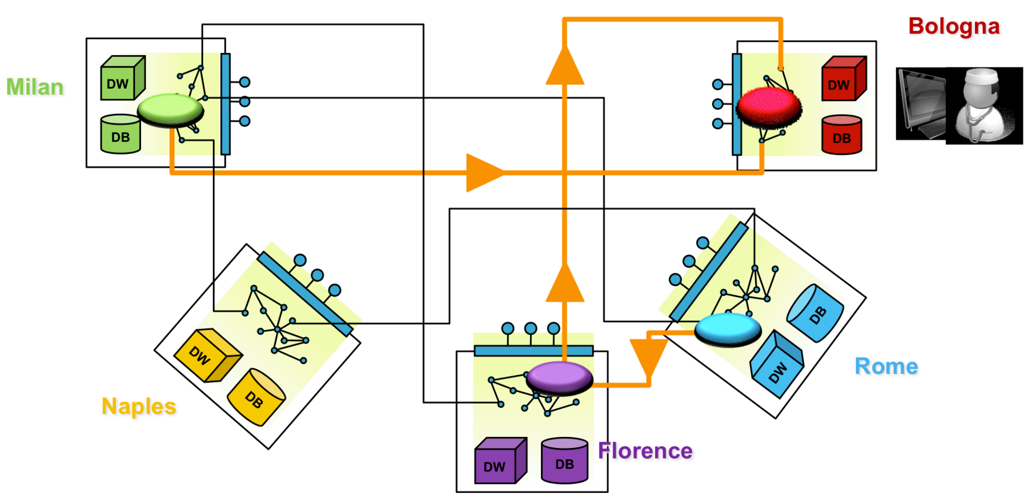
A typical user interaction

To enhance the decision making process, *q* is forwarded to the network



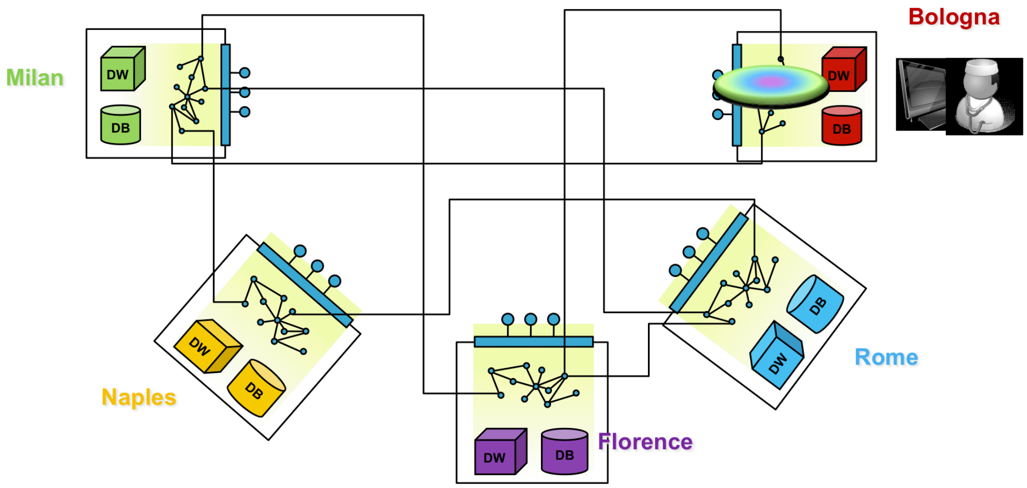
A typical user interaction

Each involved peer locally processes the query on its data warehouse and returns its (possibly partial or approximate) results to *p*



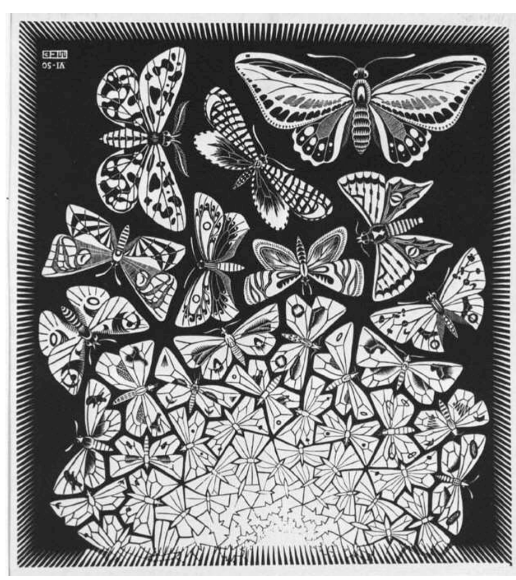
A typical user interaction

The results are integrated and returned to the user based on the lexicon used to formulate *q*

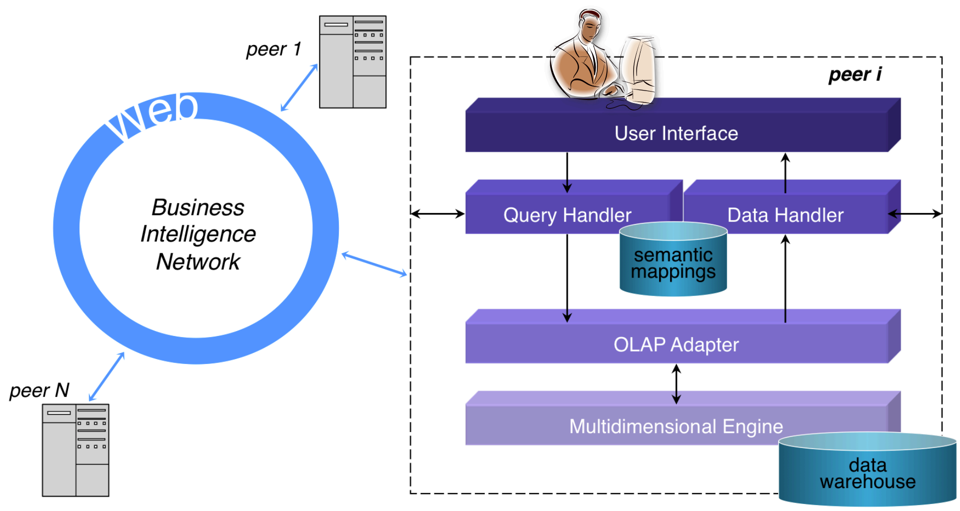


How to deal with heterogeneity?

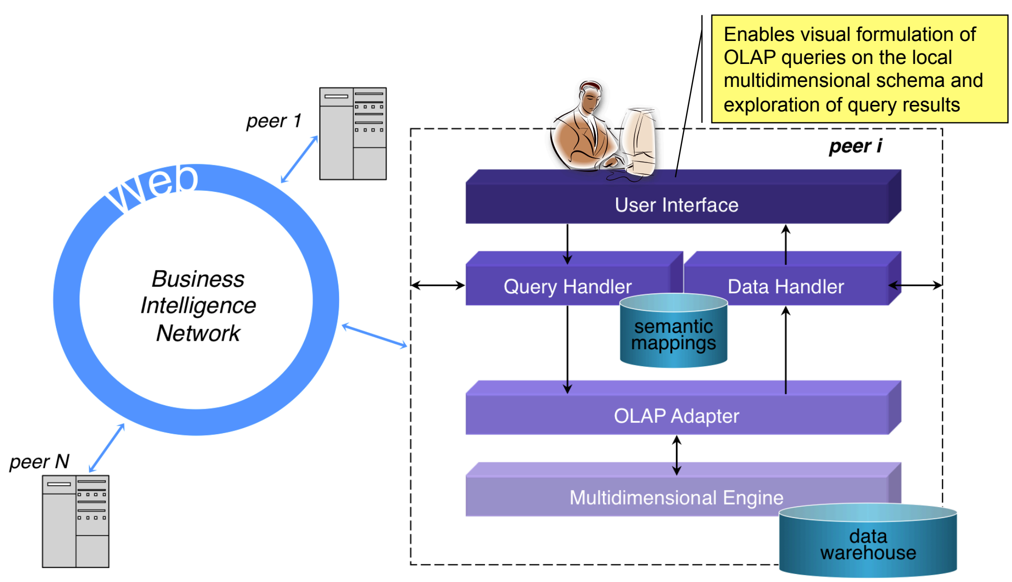
- Before a query issued on a peer can be forwarded to the network it must be first reformulated according to the multidimensional schemata of the destination peers



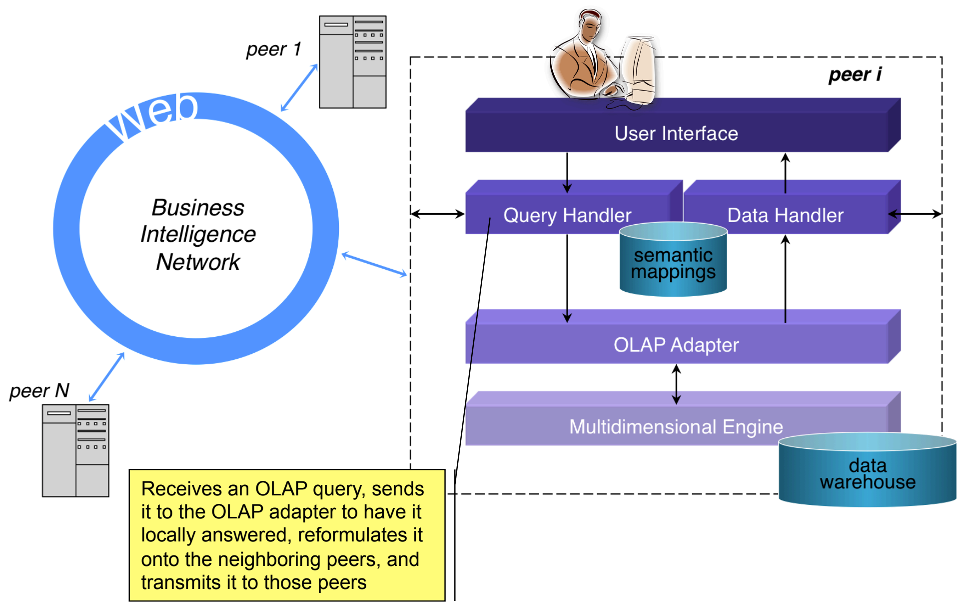
Envisioned architecture



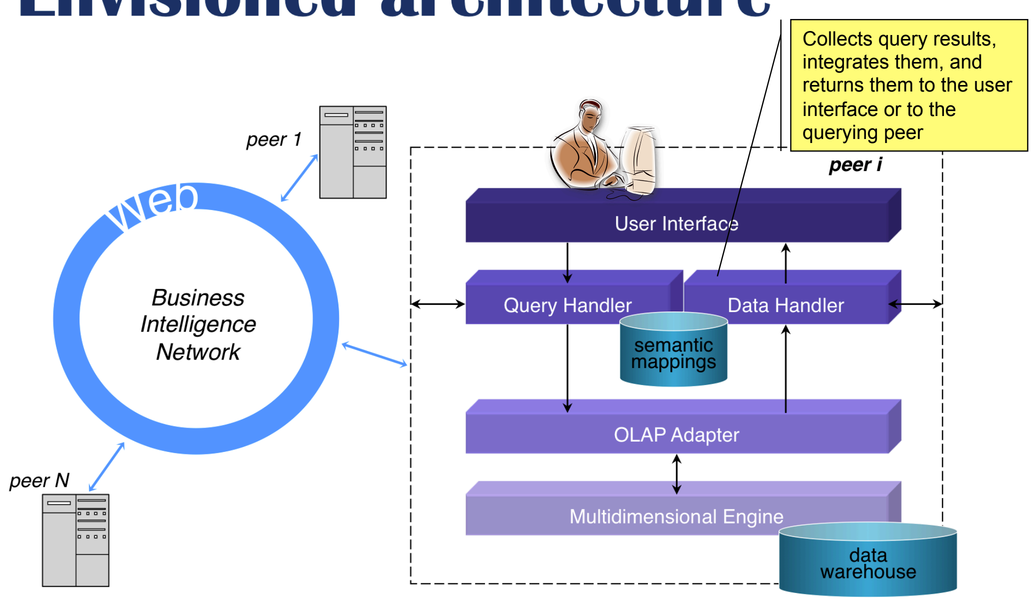
Envisioned architecture



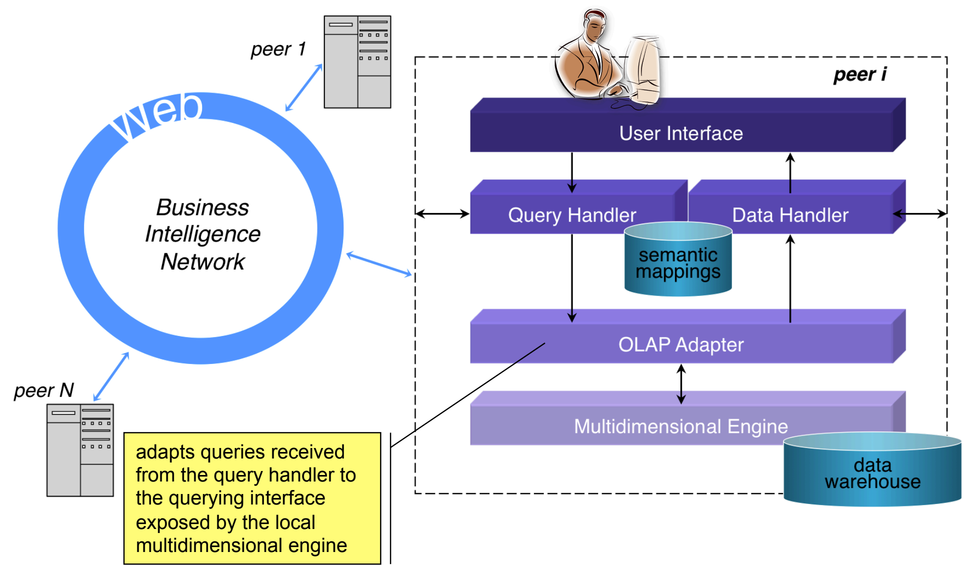
Envisioned architecture



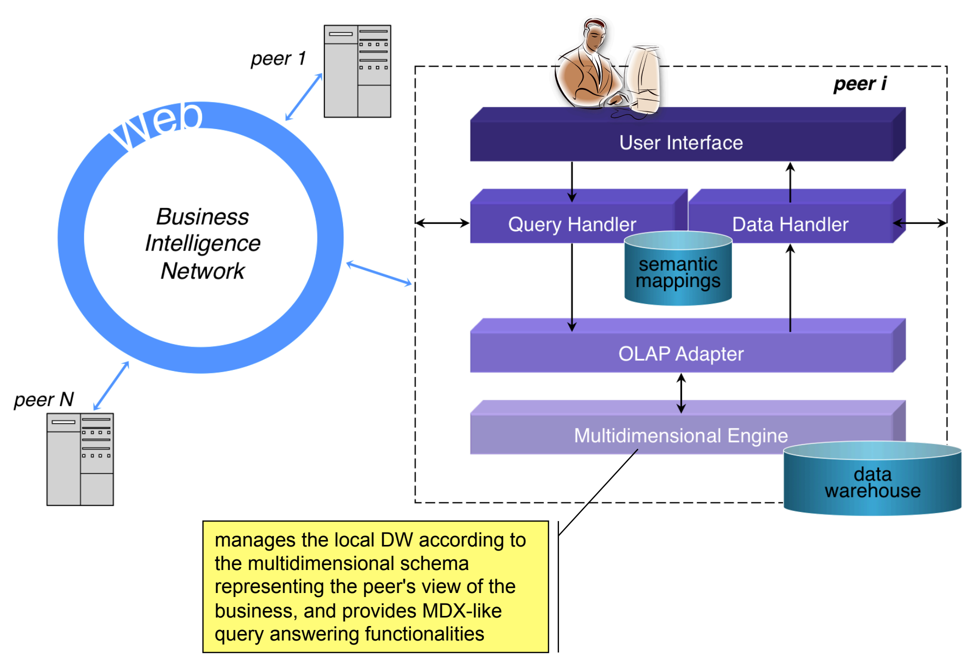
Envisioned architecture



Envisioned architecture



Envisioned architecture



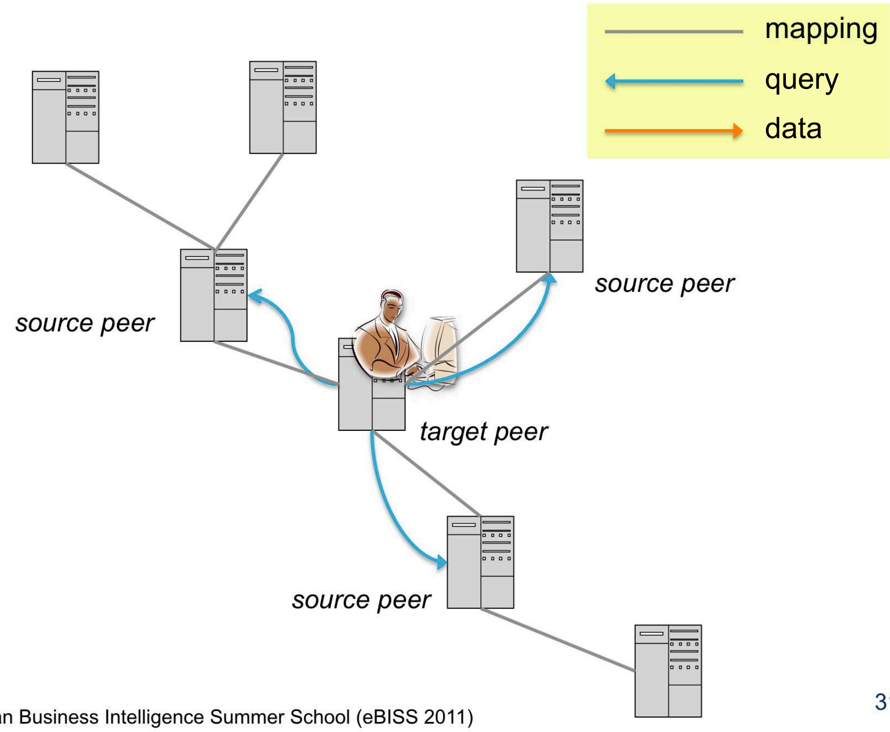
Research issues

- Query reformulation on peers is a challenging task due to the presence of aggregation and to the possibility of having information represented at different granularities in each peer
- To optimize query answering across the network, query routing strategies that forward queries to the most promising peers only are needed
- The strategic nature of the exchanged information and its multidimensional structure require advanced approaches for security
- Mechanisms for controlling data provenance and quality in order to provide users with information they can rely on should be devised
- A mechanism for data lineage is necessary to help users understand the semantics of the retrieved data and how these data have been transformed to handle heterogeneity
- A unified, integrated vision of the heterogeneous information collected must be returned to users through object fusion techniques

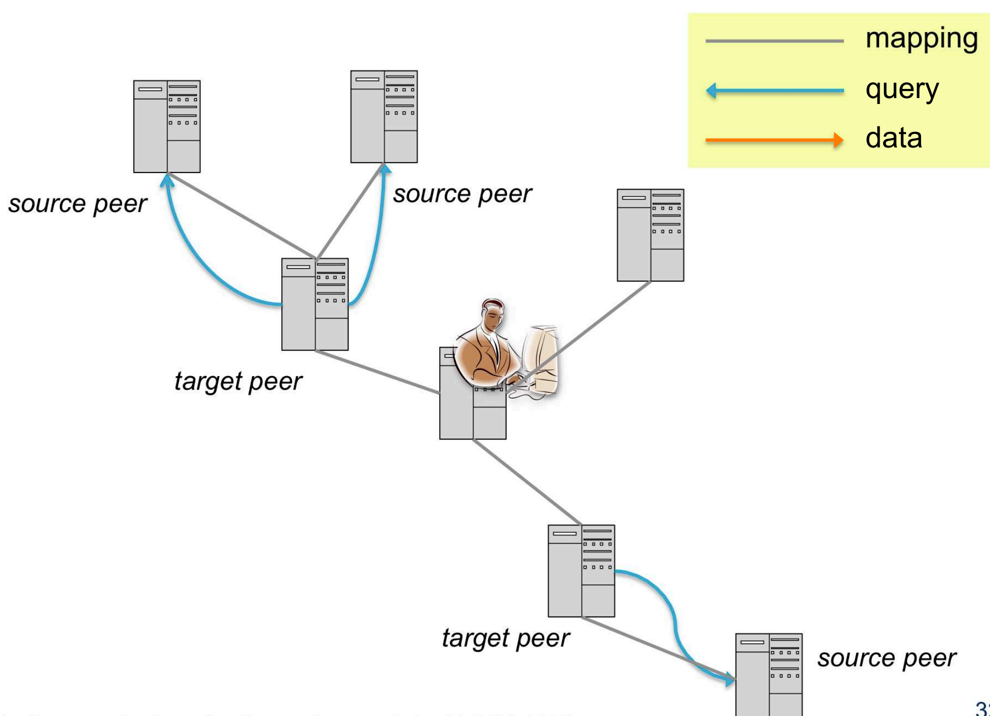
Query reformulation

- Like in PDMSs, query reformulation in a BIN is based on *semantic mappings* that mediate between the different multidimensional schemata exposed by two peers
- Direct mappings cannot be realistically defined for all the possible couples of peers; so, a query issued on *p* is forwarded to the network by first sending it to the immediate neighbors of *p*, then to their immediate neighbors, and so on
  - In this way, the query undergoes a chain of reformulations along the peers it reaches, and results are collected from any peer that is connected to *p* through a path of semantic mappings

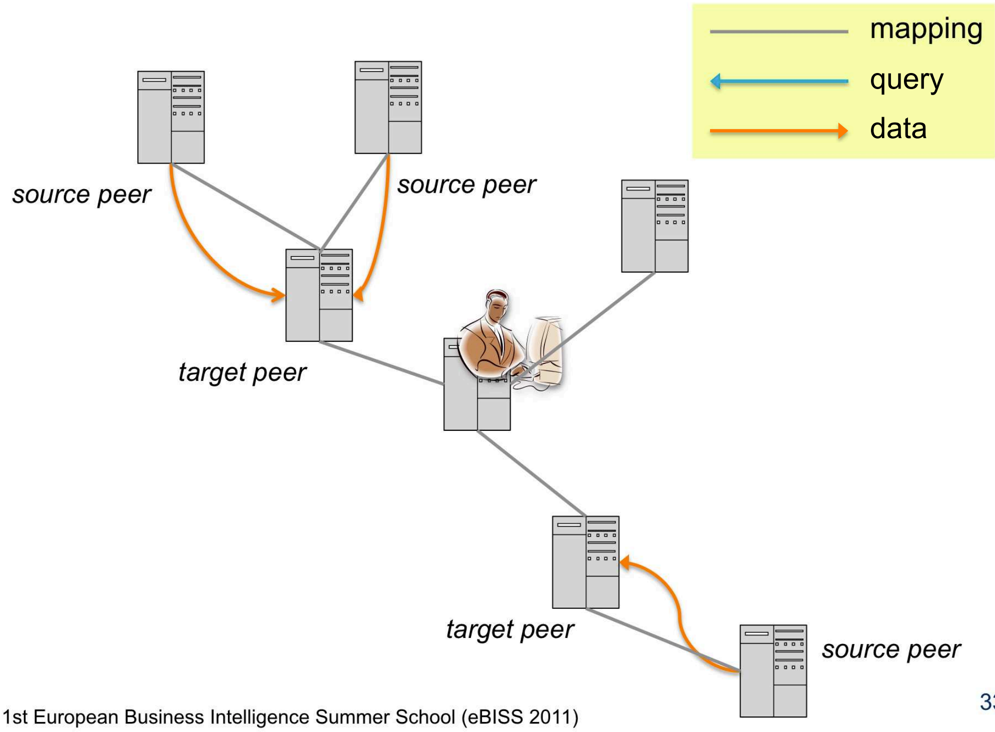
Query reformulation



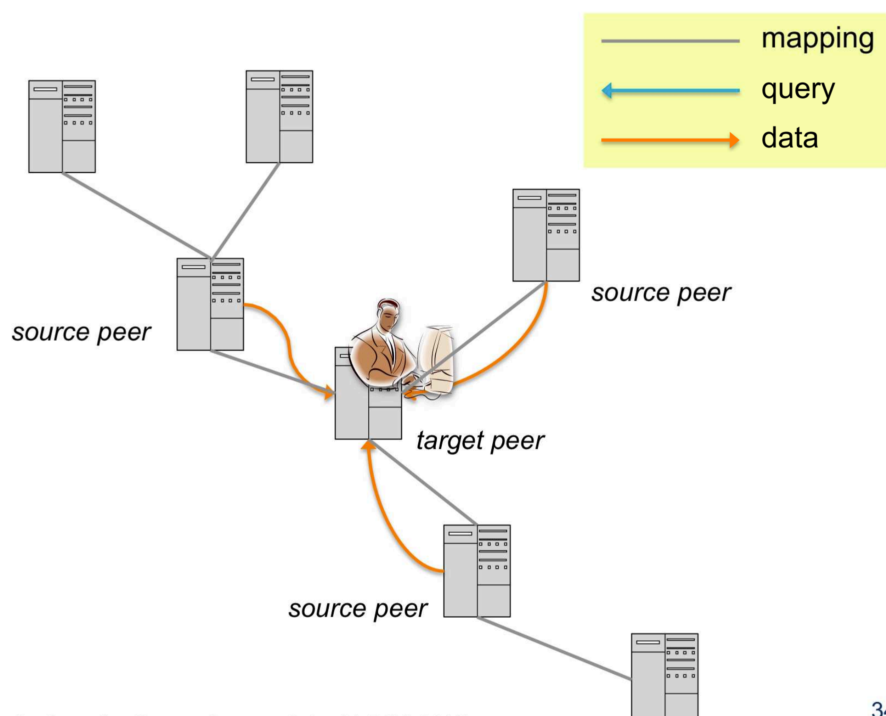
Query reformulation



Query reformulation



Query reformulation

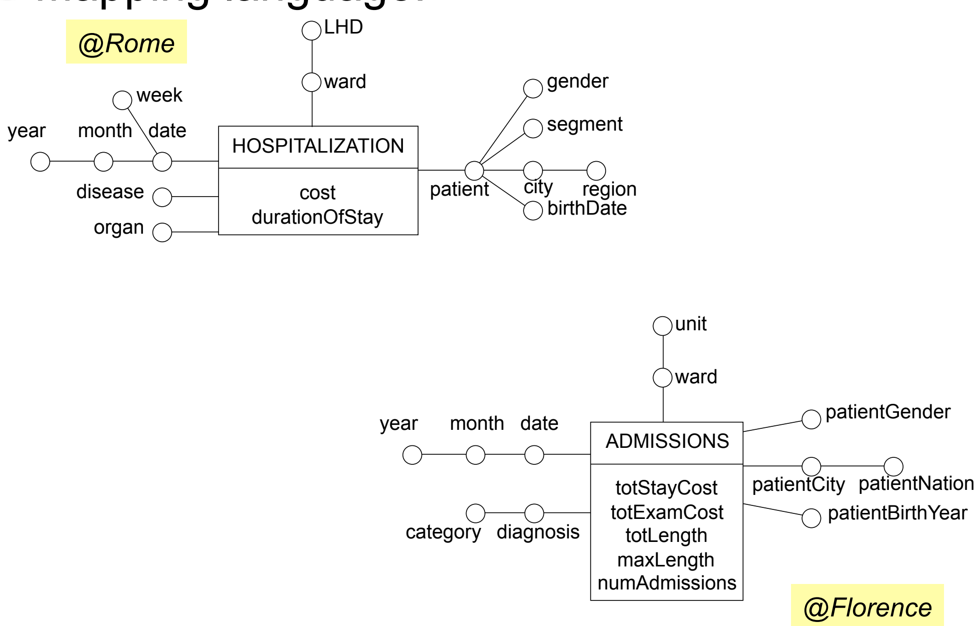


Semantic mappings

- They describe how the concepts in the multidimensional schema of one peer map onto those of another peer
- *Requirements*:
  - Handling the asymmetry between dimensions and measures
  - Specifying the relationship between two attributes of different multidimensional schemata in terms of their granularity
  - Considering aggregation operators to avoid the risk of inconsistent query reformulations
  - Expressing also mappings at the instance level to transcode data

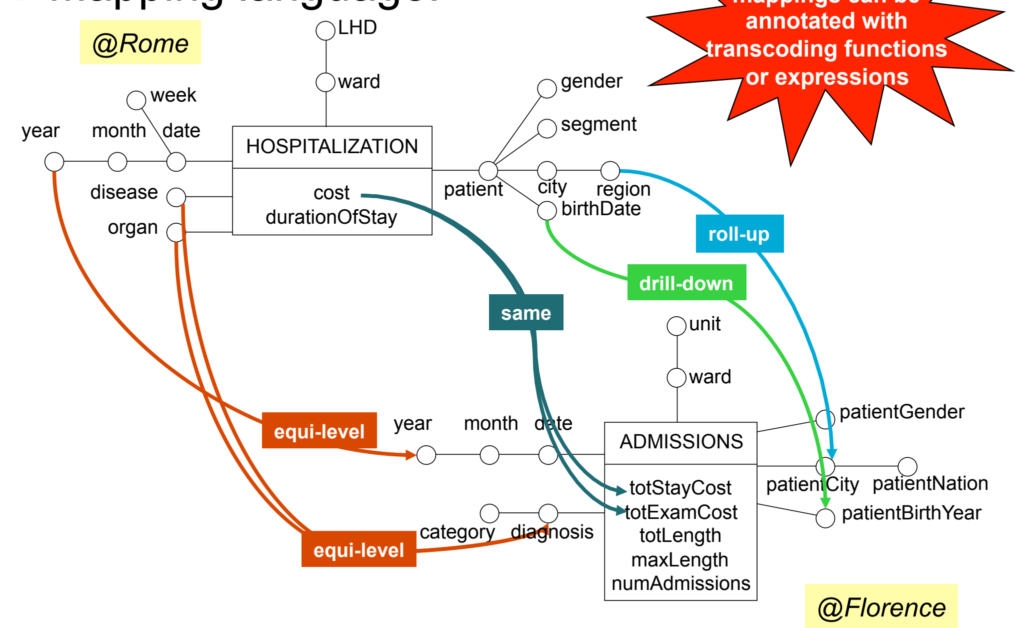
Semantic mappings

- Mapping language:



Semantic mappings

- Mapping language:



Example: mappings

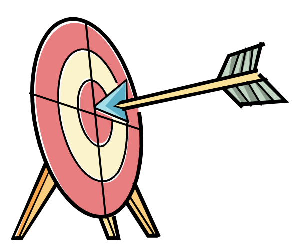
$\omega 1$	⟨ cost,sum ⟩ same { totStayCost, totExamCost }
$\omega 2$	⟨ cost,avg ⟩ same { totStayCost, totExamCost }
$\omega 3$	⟨ durationOfStay,sum ⟩ same { totLength }
$\omega 4$	⟨ durationOfStay,avg ⟩ same { totLength }
$\omega 5$	⟨ durationOfStay,max ⟩ same { maxLength }
$\omega 6$	{ LHD } roll-up { unit }
$\omega 7$	{ ward } equi-level { ward }
$\omega 8$	{ year } equi-level { year }
$\omega 9$	{ month } equi-level { month }
$\omega 10$	{ date } equi-level { date }
$\omega 11$	{ week } roll-up { date }
$\omega 12$	{ disease,organ } equi-level { diagnosis }
$\omega 13$	{ disease } drill-down { category }
$\omega 14$	{ patient } drill-down { patientGender,patientCity,patientBirthYear }
$\omega 15$	{ gender } equi-level { patientGender }
$\omega 16$	{ segment } related { patientGender,patientCity,patientBirthYear }
$\omega 17$	{ birthDate } drill-down { patientBirthYear }
$\omega 18$	{ city } equi-level { patientCity }
$\omega 19$	{ region } roll-up { patientCity }

Example: transcodings

$\omega 1$	cost = totStayCost+totExamCost, segment **in** { 'NH','EU' }
$\omega 2$	cost = totStayCost+totExamCost, segment **in** { 'NH','EU' }
$\omega 3$	durationOfStay = totLength, segment **in** { 'NH','EU' }
$\omega 4$	durationOfStay = totLength, segment **in** { 'NH','EU' }
$\omega 5$	durationOfStay = maxLength, segment **in** { 'NH','EU' }
$\omega 6$	LHD = 'LHD39 - Florence'
$\omega 7$	ward = ward
$\omega 8$	year = year
$\omega 9$	month = month
$\omega 10$	date = date
$\omega 11$	week = weekOf(date)
$\omega 12$	disease = substring(diagnosis, 1, 40), organ = substring(diagnosis, 41, 80)
$\omega 13$	categoryOf(disease) = category
$\omega 14$	—
$\omega 15$	gender = completeGender(patientGender)
$\omega 16$	—
$\omega 17$	yearOf(birthDate) = patientBirthYear
$\omega 18$	city = patientCity
$\omega 19$	region = regionOf(patientCity)

Mapping accuracy

- A mapping is exact when it is either an `equi-level` or a `roll-up` mapping and it has an associated transcoding, or it is a `same` mapping
- A mapping is loose when it is either a `drill-down` or a `related` mapping
- An attribute mapping is approximate when it has no associated transcoding

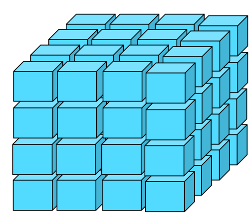


Mapping accuracy

- The accuracy of a reformulation of *q* on peer *s* depends on the accuracy of the mappings involved
  - When (i) for each attribute mentioned in *q* there is an exact mapping from *s*, and (ii) for each metric required by q there is a same mapping from *s*, there exists a *compatible* reformulation of *q* on *s*, i.e., one that fully preserves the semantics of *q*
    - when a compatible reformulation is used, the results returned by *s* do exactly match with *q* so they can be seamlessly integrated with those returned by *t*
  - In all the other cases, the results returned by *s* match *q* with some approximation
    - value mismatch
    - granularity mismatch
    - no reformulation

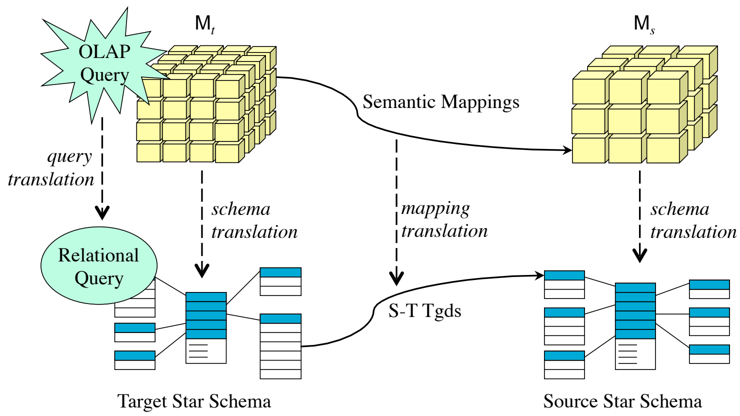
Expressiveness

- BIN query:
  - a group-by clause
  - an (optional) selection conjunctive predicate
  - a numerical expression involving measures to be computed
  - an aggregation operator to be used for each measure

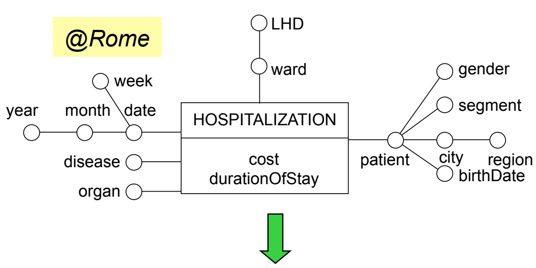


Query reformulation

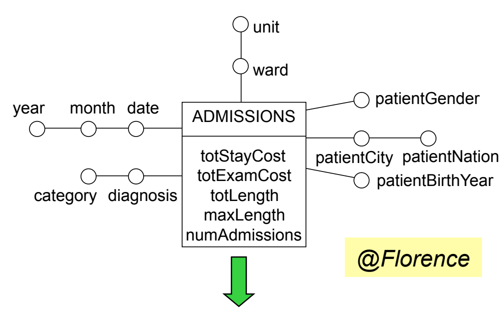
- Framework:
  - To translate semantic mappings we use a logical formalism called source-to-target tuple generating dependencies (ten Cate & Kolaitis, 2010), asserting that if a pattern of facts appears in the source, then another pattern of facts must appear in the target



Example: Schema translation



HospFT(<u>organ</u>,<u>disease</u>,<u>date</u>,<u>ward</u>,<u>patient</u>,cost,durationOfStay)
OrganDT(<u>organ</u>)
DiseaseDT(<u>disease</u>)
DateDT(<u>date</u>,week,month,year)
WardDT(<u>ward</u>,LHD)
PatientDT(<u>patient</u>,birthDate,city,region,segment,gender)



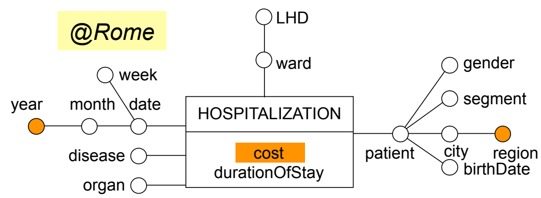
AdmFT(<u>diagnosis</u>,<u>date</u>,<u>ward</u>,<u>patientCity</u>,<u>patientBirthYear</u>,<u>patientGender</u>,
totStayCost,totExamCost,totLength,maxLength,numAdmissions)
DiagnosisDT(<u>diagnosis</u>,category)
DateDT(<u>date</u>,month,year)
WardDT(<u>ward</u>,unit)
PatientCityDT(<u>patientCity</u>,patientNation)
PatientBirthYearDT(<u>patientBirthYear</u>)
PatientGenderDT(<u>patientGender</u>)

Example: Query translation

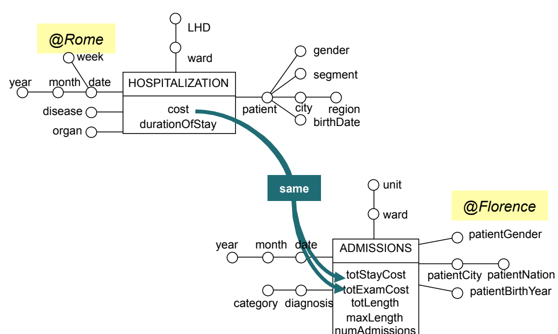
- Total hospitalization costs for region and year

$$\pi_{region,year,\mathtt{SUM}(cost)}\ (HospFT \bowtie DateDT \bowtie PatientDT)$$

$$q(R,Y,\mathtt{SUM}(C)) \leftarrow \mathrm{HospFT}(\_,\_,D,\_,P,C,\_),\ \mathrm{DateDT}(D,\_,\_,Y),\ \mathrm{PatientDT}(P,\_,\_,R,\_,\_)$$



Example: Mapping translation

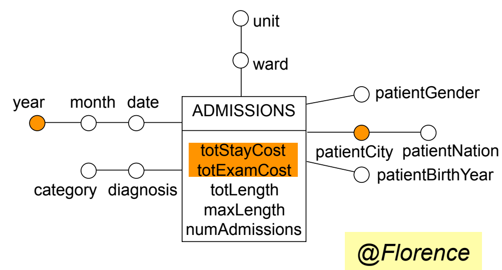


$$\forall S,E,C\ (\mathrm{AdmFT}(\_,\ldots,S,E,\_,\_),\ C=S+E \rightarrow \mathrm{HospFT}(\_,\ldots,C,\_)$$

Example: Reformulation

- The group-by is reformulated using the roll-up mapping from *region* to *patientCity* and the equi-level mapping from *year* to *year*, while measure *cost* is derived using the same mapping

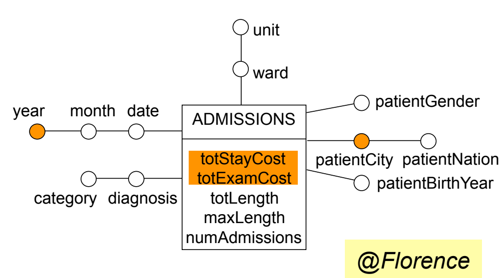
$$\pi_{\text{year,patientCity,SUM(totStayCost+totExamCost)}}\ (\text{AdmFT} \bowtie \text{DateDT} \bowtie \text{PatientCityDT})$$



Example: Reformulation

- In presence of a transcoding from *region* to *patientCity*, a compatible reformulation can be done

$$\pi_{\text{year,regionOf(patientCity),SUM(totStayCost+totExamCost)}}\ (\text{AdmFT} \bowtie \text{DateDT} \bowtie \text{PatientCityDT})$$



Theoretical results

- The BIN query language is *closed under reformulation*
  - Our query reformulation algorithm can be used by each peer in a BIN to implement chains of reformulations. In this way, any query formulated over a peer schema can be safely distributed across the network, and answers can come from any other peer in the network which is connected to the queried peer through a chain of semantic mappings
- Our reformulation algorithm is proved to be *sound and complete* with respect to the semantics of query answering, that in data sharing settings is usually given in terms of certain answers

Implementation issues

- How to bridge the language gap between the query handler and the local multidimensional engine?
  - A BIN query cannot be directly executed on the peer local multidimensional engine
  - Intra-peer reformulation must deal with the presence of transcodings in the query group-by set, and must properly manage non-distributive aggregation operators

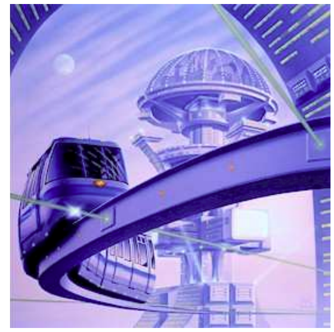
Implementation issues

- How to share transcodings among peers?
  - *Public transcodings* are standard database functions that are shared by all peers
  - *Protected transcodings* are owned by a peer, that will make them available to its neighboring peers by attaching them to query messages
    - If protected transcodings are expressed as procedures, a shared programming language must be available in the BIN
    - Otherwise, transcodings can be expressed as look-up tables to be applied by a relational engine; in this case, an obvious drawback is the quantity of information to be transmitted over the network

Summary

- We have surveyed the basic approaches to collaborative BI
- We have outlined a peer-to-peer architecture for supporting distributed and collaborative decision-making scenarios
- We have shown how an OLAP query formulated on one peer can be reformulated on a different peer, based on a set of inter-peer semantic mappings

Open issues



- Devising multidimensional-aware object fusion techniques for integrating data returned by different peers
- Finding smart algorithms for routing queries to the most "promising" peers in the BI network
- Designing smart user interfaces for emphasizing the differences and relationships between the returned data
- Using preferences to rank the returned data depending on how compliant they are with the original local query
- Studying mechanisms for controlling data provenance and quality to provide users with reliable information
- Devising advanced approaches for security, especially data sharing policies that depend on the degree of trust between participants

Related readings

- Abiteboul, S. Managing an XML warehouse in a P2P context. In *Proc. CAISE*, 2003
- Albrecht, J., & Lehner, W. On-line analytical processing in distributed data warehouses. In *Proc. IDEAS*, 1998
- Akinde, M.O., Bohlen, M.H., Johnson, T., Lakshmanan, L.V.S., & Srivastava, D. Efficient OLAP query processing in distributed data warehouses. *Inf. Syst*. 28(1- 2), 2003
- Banek, M., Vrdoljak, V., Min Tjoa, A., & Skocir, Z. Automated integration of heterogeneous data warehouse schemata. *IJDWM, 4*(4), 2008
- Berger, S., & Schrefl, M. From federated databases to a federated data warehouse system. In *Proc. HICSS*, 2008
- Chang, K.C., Garcia-Molina, H.: Mind your vocabulary: Query mapping across heterogeneous information sources. In *Proc. SIGMOD,* 1999
- Dubois, D., & Prade, H. On the use of aggregation operations in information fusion processes. *International Journal on Fuzzy Sets and Systems, 142*(1), 2004
- Georgiadis, P., Kapantaidakis, I., Christophides, V., Nguer, E. M., & Spyratos, N. Efficient rewriting algorithms for preference queries. In *Proc. ICDE*, 2008
- Golfarelli, M., Mandreoli, F., Penzo, W., Rizzi, S., & Turricchia, E. BIN: Business intelligence networks. In *Business Intelligence Applications and the Web: Models, Systems and Technologies*, IGI Global, 2011 (to appear).
- Golfarelli, M., Mandreoli, F., Penzo, W., Rizzi, S., & Turricchia, E. OLAP Query Reformulation in Peer-to-Peer Data Warehousing. *Information Systems*, 2011 (to appear)
- Halevy, A. Y., Ives, Z. G., Madhavan, J., Mork, P., Suciu, D., & Tatarinov, I. The Piazza Peer Data Management System. *IEEE TKDE, 16*(7), 2004
- Hoang, T. A. D., & Binh Nguyen, T. State of the art and emerging rule-driven perspectives towards service-based business process interoperability. In *Proc. Int. Conf. on Computing and Communication Technologie*, 2009

Related readings

- Kalnis, P., Siong Ng, W., Chin Ooi, B., Papadias, D., & Tan, K.-L. An adaptive peer-to-peer network for distributed caching of OLAP results. In *Proc. SIGMOD Conference*, 2002
- Kehlenbeck, M., & Breitner, M. H. Ontology-based exchange and immediate application of business calculation definitions for online analytical processing. In *Proc. DAWAK*, 2009
- Kießling, W. Foundations of preferences in database systems. In *Proc. VLDB*, 2002
- Mandreoli, F., Martoglia, R., Penzo, W., & Sassatelli S. SRI: exploiting semantic information for effective query routing in a PDMS. In *Proc. ACM Int. Workshop on Web Information and Data Management*, 2006
- Mecca, G., Papotti, P., & Raunich, S. Core Schema Mappings. In *Proc. ACM SIGMOD Int. Conf. on Management of Data*, 2009
- Papakonstantinou, Y., Abiteboul, S., & Garcia-Molina, H. Object fusion in mediator systems. In *Proc. VLDB*, 1996
- Schneider, M. Integrated vision of federated data warehouses. In *Proc. DISWEB*, 2006
- Sung, S., Liu, Y., Xiong, H., & Ng, P. Privacy preservation for data cubes. *Knowledge and Information Systems*, *9*(1), 2006
- Tatarinov, I. & Halevy, A.Y. Efficient Query Reformulation in Peer-Data Management Systems. In *Proc. ACM SIGMOD Int. Conf. on Management of Data*, 2004
- ten Cate, B. & Kolaitis, P. G. Structural characterizations of schema-mapping languages. *Comm. ACM*, 53 (1), 2010
- Torlone, R. Two approaches to the integration of heterogeneous data warehouses. *Int. Journ. on Distributed and Parallel Databases, 23*(1), 2008
- Vaisman, A., Espil, M.M., & Paradela, M. P2P OLAP: Data model, implementation and case study. *Information Systems,* 34(2), 2009

Thank you for you attention

Questions?