The SEWASIE system: A multi-agent system for querying heterogeneous data sources with ontologies

Pablo R. Fillottrani
Faculty of Computer Science
Free University of Bozen/Bolzano
Piazza Domenicani, 3 39100 Bolzano, Italy

- SEWASIE (SEmantic Web and AgentS in Integrated Economies)

- SEWASIE (SEmantic Web and AgentS in Integrated Economies)
- shared-cost RTD project, within European IST programme

The Sewasie Project

- SEWASIE (SEmantic Web and AgentS in Integrated Economies)
- shared-cost RTD project, within European IST programme
- started on May 2002, and finished on May 2005

Objectives

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web
- relying in a distributed agent-based architecture

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web
- relying in a distributed agent-based architecture
- integrating semantic search and communication (semiautomatic negotiation) is based on community-specific multilingual ontologies

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web
- relying in a distributed agent-based architecture
- integrating semantic search and communication (semiautomatic negotiation) is based on community-specific multilingual ontologies
- a semantic enrichment process for knowledge-base extraction of meta-information of the heterogeneous data sources

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web
- relying in a distributed agent-based architecture
- integrating semantic search and communication (semiautomatic negotiation) is based on community-specific multilingual ontologies
- a semantic enrichment process for knowledge-base extraction of meta-information of the heterogeneous data sources
- efficient user-friendly interfaces for formulating queries

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web
- relying in a distributed agent-based architecture
- integrating semantic search and communication (semiautomatic negotiation) is based on community-specific multilingual ontologies
- a semantic enrichment process for knowledge-base extraction of meta-information of the heterogeneous data sources
- efficient user-friendly interfaces for formulating queries
- in well defined economic sectors where large numbers of SMEs need to cooperate

Objectives

- designing and implementing an advanced search engine enabling SMEs with intelligent access to heterogeneous data sources on the web
- relying in a distributed agent-based architecture
- integrating semantic search and communication (semiautomatic negotiation) is based on community-specific multilingual ontologies
- a semantic enrichment process for knowledge-base extraction of meta-information of the heterogeneous data sources
- efficient user-friendly interfaces for formulating queries
- in well defined economic sectors where large numbers of SMEs need to cooperate
- grounded on W3C standards

Partners

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)
- RWTH Aachen (D)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)
- RWTH Aachen (D)
- **Free University of Bozen/Bolzano** (I)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)
- RWTH Aachen (D)
- **Free University of Bozen/Bolzano** (I)
- CNA Servizi Modena s.c.a.r.l. (I)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)
- RWTH Aachen (D)
- **Free University of Bozen/Bolzano** (I)
- CNA Servizi Modena s.c.a.r.l. (I)
- Thinking Networks AG (D)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)
- RWTH Aachen (D)
- **Free University of Bozen/Bolzano** (I)
- CNA Servizi Modena s.c.a.r.l. (I)
- Thinking Networks AG (D)
- IBM Italia SpA (I)

Partners

- Universit di Modena e Reggio Emilia (I, coordinator)
- Universit degli Studi di Roma "La Sapienza" (I)
- RWTH Aachen (D)
- **Free University of Bozen/Bolzano** (I)
- CNA Servizi Modena s.c.a.r.l. (I)
- Thinking Networks AG (D)
- IBM Italia SpA (I)
- Fraunhofer-Gesellschaft/FIT (D)

Achievements

Achievements

- integration of core modules in an agent-based architecture

Achievements

- integration of core modules in an agent-based architecture
- integration of extended features modules (negotiation, OLAP systems, visualization components)

Achievements

- integration of core modules in an agent-based architecture
- integration of extended features modules (negotiation, OLAP systems, visualization components)
- high degree of automation in updating and integrating ontologies, and in adding new information sources

Achievements

- integration of core modules in an agent-based architecture
- integration of extended features modules (negotiation, OLAP systems, visualization components)
- high degree of automation in updating and integrating ontologies, and in adding new information sources
- extensive user testing and feedbacks

Achievements

- integration of core modules in an agent-based architecture
- integration of extended features modules (negotiation, OLAP systems, visualization components)
- high degree of automation in updating and integrating ontologies, and in adding new information sources
- extensive user testing and feedbacks
- several dissemination and explotation activities

Achievements

- integration of core modules in an agent-based architecture
- integration of extended features modules (negotiation, OLAP systems, visualization components)
- high degree of automation in updating and integrating ontologies, and in adding new information sources
- extensive user testing and feedbacks
- several dissemination and explotation activities

More info: `www.sewasie.org`

This Talk

- aims to describe the agent technology used in SEWASIE, especially in the query process implementation

- aims to describe the agent technology used in SEWASIE, especially in the query process implementation

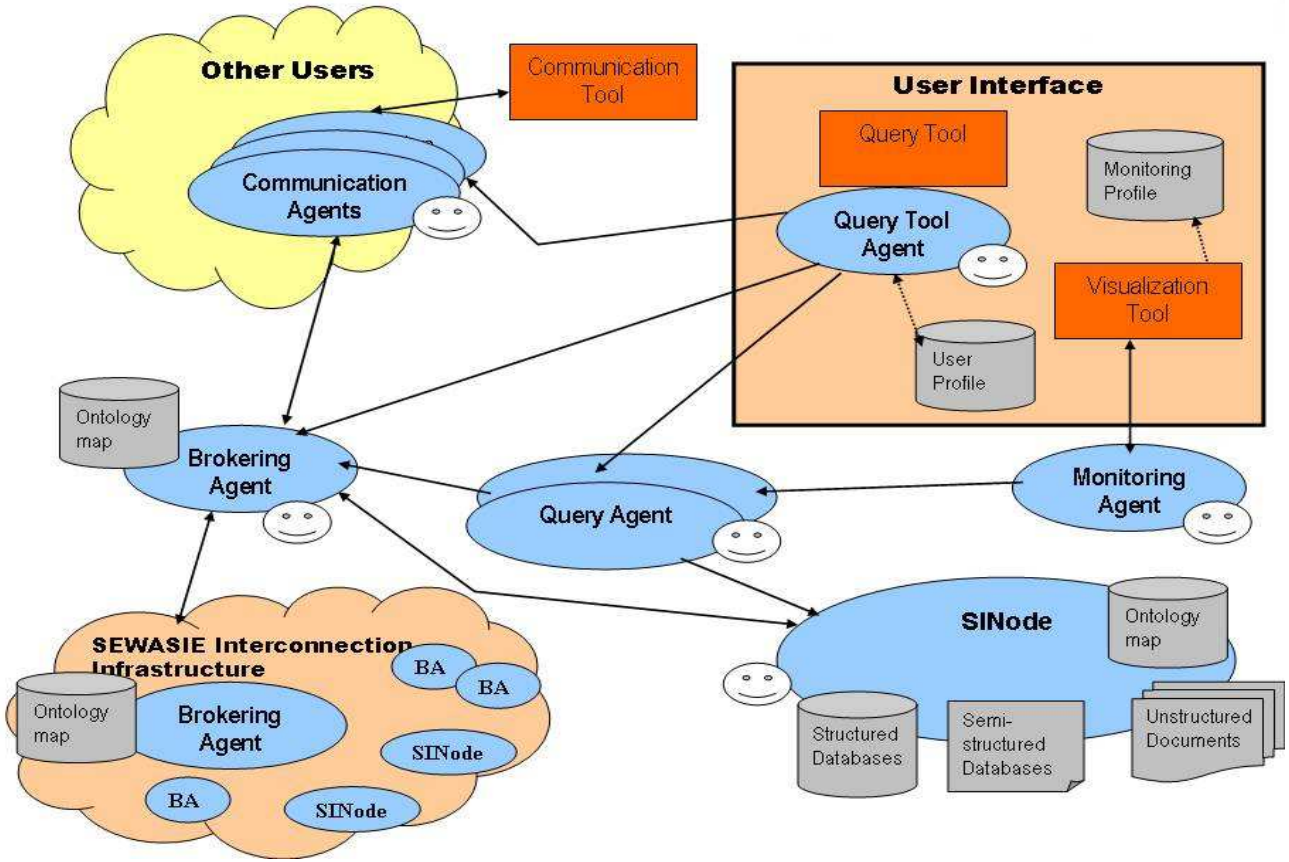
Talk Overview

1. general architecture
2. query tool
3. query process
4. system deployment
5. conclusions

Talk Overview

1. general architecture
2. query tool
3. query process
4. system deployment
5. conclusions

General Architecture



Main features:

Main features:

- two levels of data integration:

Main features:

- two levels of data integration:
  - Lower level: nodes with integrated information from communities with strong ties (SINodes)

Main features:

- two levels of data integration:
  - Lower level: nodes with integrated information from communities with strong ties (SINodes)
  - Higher level: weaker relationships between these integrated nodes (BA)

Main features:

- two levels of data integration:
  - Lower level: nodes with integrated information from communities with strong ties (SINodes)
  - Higher level: weaker relationships between these integrated nodes (BA)
- agent technology provides the unifying framework: core modules are fully agentized, extended features are integrated trhough proxy agents

Main features:

- two levels of data integration:
  - Lower level: nodes with integrated information from communities with strong ties (SINodes)
  - Higher level: weaker relationships between these integrated nodes (BA)
- agent technology provides the unifying framework: core modules are fully agentized, extended features are integrated trhough proxy agents
- extended features: query tool, visualization tool, communication tool

SINodes

- SINodes are mediator-based systems that manage heterogeneous information sources

SINodes

- SINodes are mediator-based systems that manage heterogeneous information sources
- represent the lower level of knowledge integration

SINodes

- SINodes are mediator-based systems that manage heterogeneous information sources
- represent the lower level of knowledge integration
- maintain a Global Virtual View (GVV) of the overall information they manage

SINodes

- SINodes are mediator-based systems that manage heterogeneous information sources
- represent the lower level of knowledge integration
- maintain a Global Virtual View (GVV) of the overall information they manage
- an ontology builder component helps in the process of semantic enrichment.

Brokering Agents

Brokering Agents

- implement the higher level of knowledge integration.

Brokering Agents

- implement the higher level of knowledge integration.
- one BA maintains a view of all the knowledge handled in the network, but this view is not global

Brokering Agents

- implement the higher level of knowledge integration.
- one BA maintains a view of all the knowledge handled in the network, but this view is not global
- this knowledge is in the form of an ontology

Brokering Agents

- implement the higher level of knowledge integration.
- one BA maintains a view of all the knowledge handled in the network, but this view is not global
- this knowledge is in the form of an ontology
- typically, there is one BA for each business domain

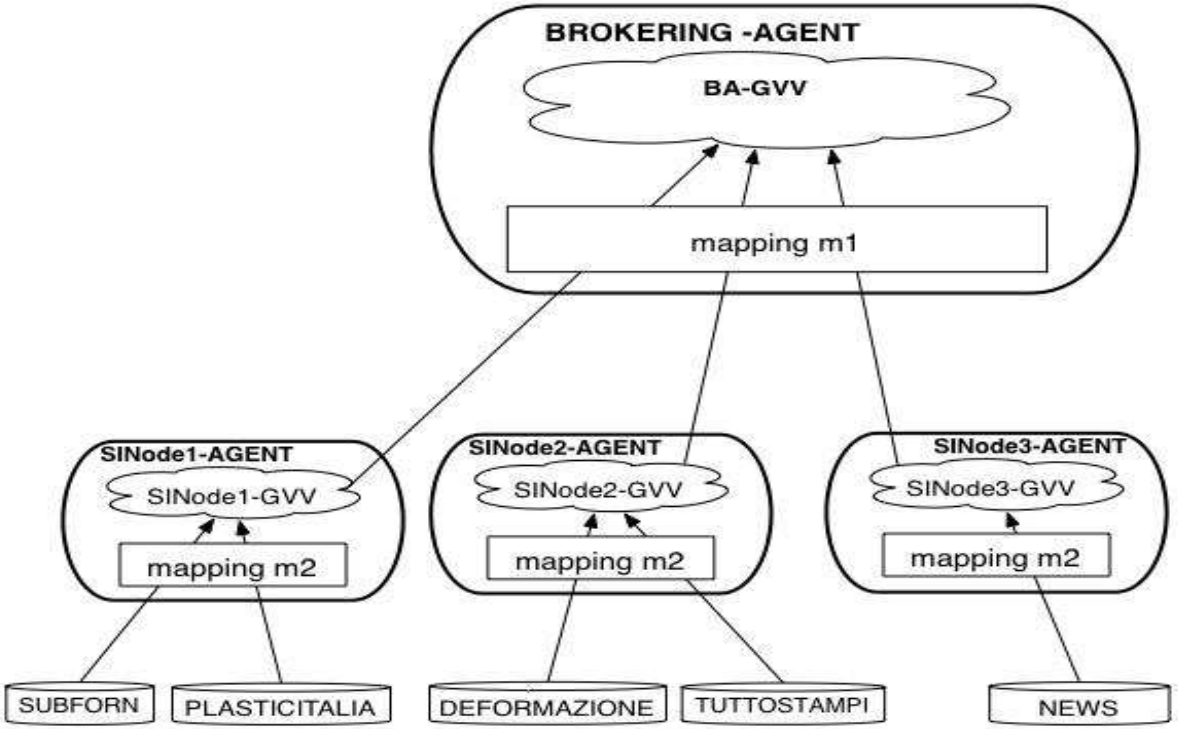
Brokering Agents

- implement the higher level of knowledge integration.
- one BA maintains a view of all the knowledge handled in the network, but this view is not global
- this knowledge is in the form of an ontology
- typically, there is one BA for each business domain
- they maintain a direct control of several SINodes, and are aware of other BAs

Brokering Agents

- implement the higher level of knowledge integration.
- one BA maintains a view of all the knowledge handled in the network, but this view is not global
- this knowledge is in the form of an ontology
- typically, there is one BA for each business domain
- they maintain a direct control of several SINodes, and are aware of other BAs
- they are the entry points to the system, routing the queries to relevant information nodes

Example: the mechanical sector BA



Query Agents

Query Agents

- carry the user query from the user interface to the SINodes

Query Agents

- carry the user query from the user interface to the SINodes
- they solve the query by interacting with BAs first, and then with SINodes

Query Agents

- carry the user query from the user interface to the SINodes
- they solve the query by interacting with BAs first, and then with SINodes
- BAs return both SINodes and other BAs references relevant to the query

Query Agents

- carry the user query from the user interface to the SINodes
- they solve the query by interacting with BAs first, and then with SINodes
- BAs return both SINodes and other BAs references relevant to the query
- before contacting a SINode, the QA must translate the query to its GVV, following the BA mappings

Query Agents

- carry the user query from the user interface to the SINodes
- they solve the query by interacting with BAs first, and then with SINodes
- BAs return both SINodes and other BAs references relevant to the query
- before contacting a SINode, the QA must translate the query to its GVV, following the BA mappings
- can be also created by other agents

Query Tool:

Query Tool:

- integrated user interaction with the semantic search system: guides the user in the process of creating queries, composed by terms from the ontology

Query Tool:

- integrated user interaction with the semantic search system: guides the user in the process of creating queries, composed by terms from the ontology
- supports the user in formulating a precise query -which best captures her/his information needs- even in case of complete ignorance of the vocabulary of the underlying information systems

Query Tool:

- integrated user interaction with the semantic search system: guides the user in the process of creating queries, composed by terms from the ontology
- supports the user in formulating a precise query -which best captures her/his information needs- even in case of complete ignorance of the vocabulary of the underlying information systems
- also include an ontology design tool

Query Tool:

- integrated user interaction with the semantic search system: guides the user in the process of creating queries, composed by terms from the ontology
- supports the user in formulating a precise query -which best captures her/his information needs- even in case of complete ignorance of the vocabulary of the underlying information systems
- also include an ontology design tool
- each instance includes a Query Tool Agent that is responsible to carry out communications with other agents

Query Tool:

- integrated user interaction with the semantic search system: guides the user in the process of creating queries, composed by terms from the ontology
- supports the user in formulating a precise query -which best captures her/his information needs- even in case of complete ignorance of the vocabulary of the underlying information systems
- also include an ontology design tool
- each instance includes a Query Tool Agent that is responsible to carry out communications with other agents
- the final purpose of the tool is to generate a conjunctive query ready to be executed by the evaluation engine associated to the information system

Visualization Tool:

Visualization Tool:

- allows the visual presentation of results, combining internal and external business perspectives

Visualization Tool:

- allows the visual presentation of results, combining internal and external business perspectives
- includes the results coming from the process of monitoring information sources

Visualization Tool:

- allows the visual presentation of results, combining internal and external business perspectives
- includes the results coming from the process of monitoring information sources
- a Monitoring Agent is created for each topic of interest, collecting iformation, filtering query results and filling monitoring repositories.

Visualization Tool:

- allows the visual presentation of results, combining internal and external business perspectives
- includes the results coming from the process of monitoring information sources
- a Monitoring Agent is created for each topic of interest, collecting iformation, filtering query results and filling monitoring repositories.
- this data is later linked to OLAP reports, with the generation of automatic annotations

Communication Tool:

Communication Tool:

- supports negotiations between the user and other parties

Communication Tool:

- supports negotiations between the user and other parties
- the starting point is a query results including contact information, and the initial ontology

Communication Tool:

- supports negotiations between the user and other parties
- the starting point is a query results including contact information, and the initial ontology
- an Initiation Agent is created for starting this process

Communication Tool:

- supports negotiations between the user and other parties
- the starting point is a query results including contact information, and the initial ontology
- an Initiation Agent is created for starting this process
- the semi-automatic negotiation is also carried out by agents: Initiation agents, Negotiation agents, Filtering and Ranking agents, and Resource Management agents

Talk Overview

1. general architecture
2. query tools
3. query process
4. system deployment
5. conclusions

- the intelligence of the interface is driven by an ontology describing the domain of the data in the information system

Query Tool

- the intelligence of the interface is driven by an ontology describing the domain of the data in the information system
- the ontology defines a vocabulary that is richer than the logical schema of the underlying data, and is meant to be closer to the user's rich vocabulary

Query Tool

- the intelligence of the interface is driven by an ontology describing the domain of the data in the information system
- the ontology defines a vocabulary that is richer than the logical schema of the underlying data, and is meant to be closer to the user's rich vocabulary
- the query supports intensional navigation, helping the unskilled user by overcoming problems related to the lack of schema comprehension

Query Tool

- the intelligence of the interface is driven by an ontology describing the domain of the data in the information system
- the ontology defines a vocabulary that is richer than the logical schema of the underlying data, and is meant to be closer to the user's rich vocabulary
- the query supports intensional navigation, helping the unskilled user by overcoming problems related to the lack of schema comprehension
- refinement process: users may specify their requests using generic terms, refine some terms of the query, or introducing new terms; and then iterating the process

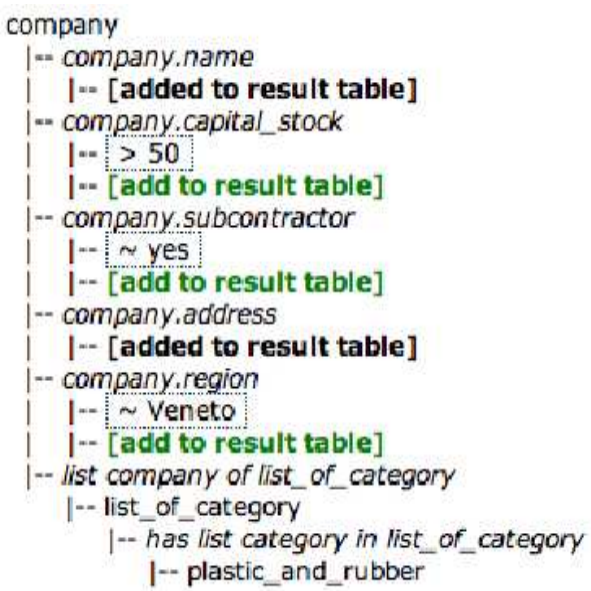
Query Tool

- the intelligence of the interface is driven by an ontology describing the domain of the data in the information system
- the ontology defines a vocabulary that is richer than the logical schema of the underlying data, and is meant to be closer to the user's rich vocabulary
- the query supports intensional navigation, helping the unskilled user by overcoming problems related to the lack of schema comprehension
- refinement process: users may specify their requests using generic terms, refine some terms of the query, or introducing new terms; and then iterating the process
- the paradigm is explore and discover information about the domain, by getting an explicit meaning to a query and to its subparts through classification

Query Tool

- the query is structured as a tree-shaped conjunctive query

- the query is structured as a tree-shaped conjunctive query
- example: *Give me the subcontracting companies in Veneto with a big capital stock in the plastic and rubber sector*



- manipulation of the query is always restricted to a well defined, and visually delimited, subpart of the query: the focus

Query Tool

- manipulation of the query is always restricted to a well defined, and visually delimited, subpart of the query: the focus
- the system allows only the operations which are "compatible" with the focus in the context of the current query expression

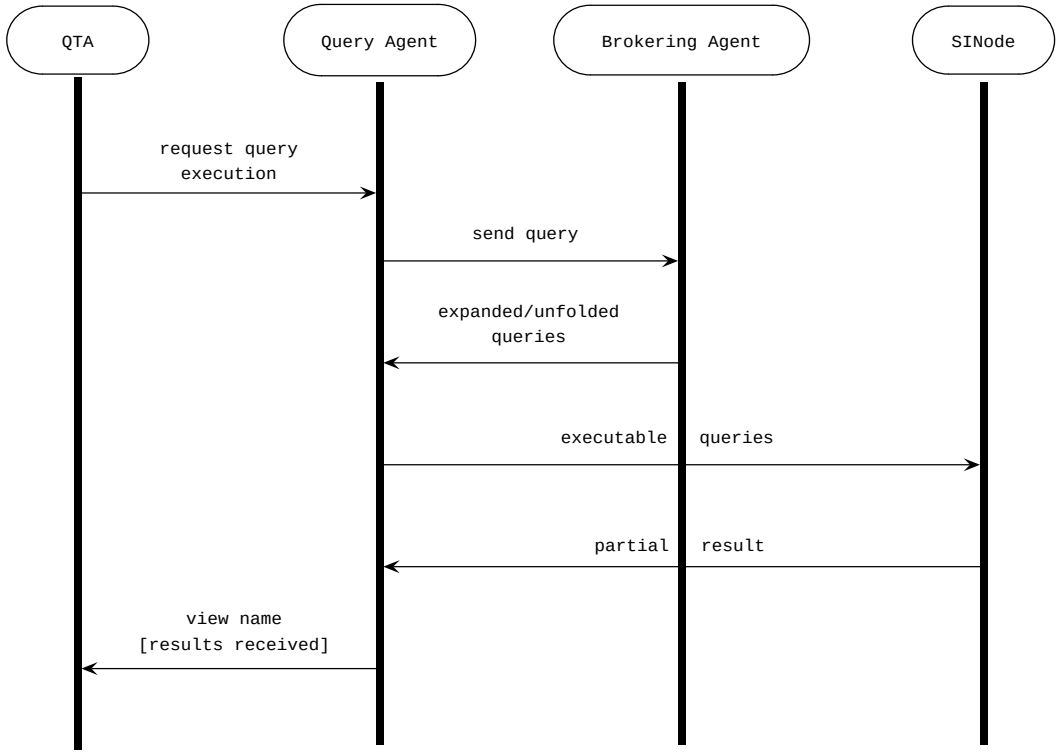
Query Tool

- manipulation of the query is always restricted to a well defined, and visually delimited, subpart of the query: the focus
- the system allows only the operations which are "compatible" with the focus in the context of the current query expression
- a description logic reasoner is called each time the query expression is modified, in order to get "compatible" expressions

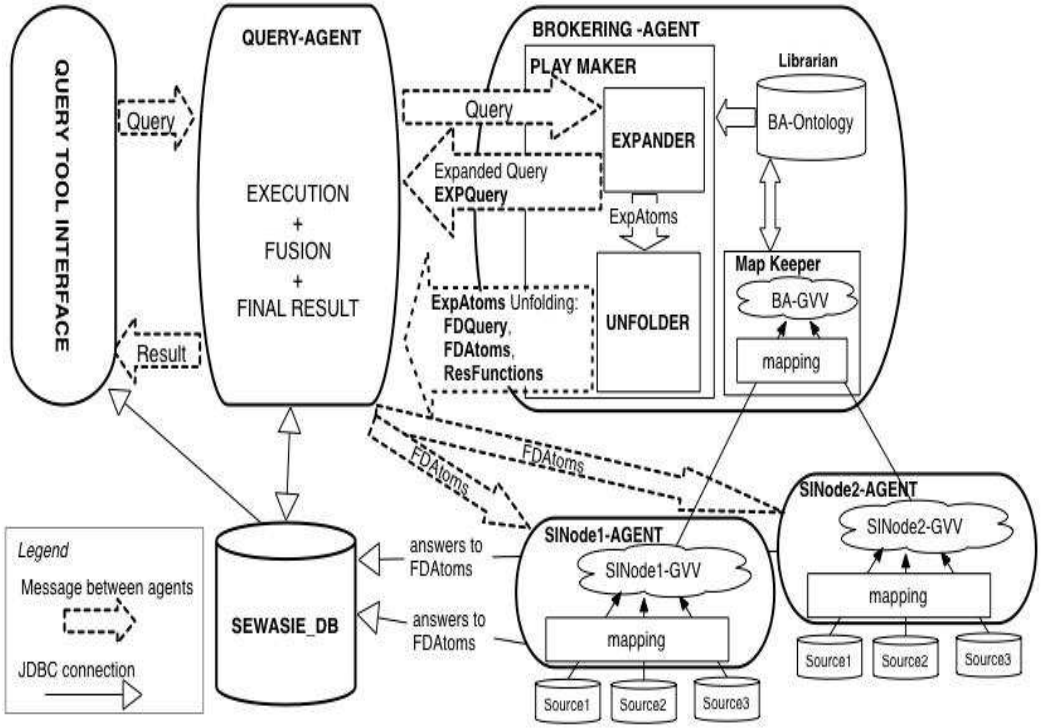
Talk Overview

1. general architecture
2. query tool
3. query process
4. system deployment
5. conclusions

General Description

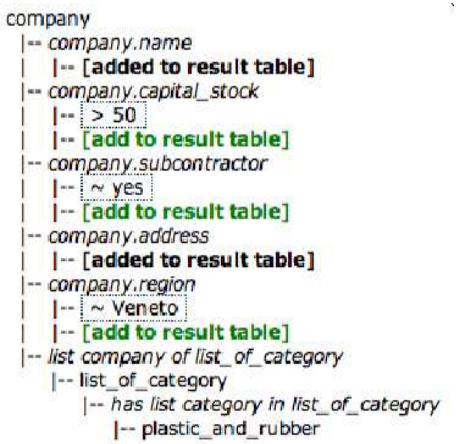


A More Detailed View



- Query Expansion: the query is expanded by taking into account the constraints (key constraints, foreign key constraints) in the BA-GVV: all constraints in the ontology are "compiled in" the expansion (EXPQuery), so that it can be processed ignoring constraints

- Query Expansion: the query is expanded by taking into account the constraints (key constraints, foreign key constraints) in the BA-GVV: all constraints in the ontology are "compiled in" the expansion (EXPQuery), so that it can be processed ignoring constraints
- example:



EXPQuery:
SELECT $r2.NAME, r2.ADDRESS, r2.NATION$
FROM $scq1\ r1, scq2\ r2, scq3\ r3$
WHERE $r1.CATEGORY\_ID = r3.CATEGORY\_ID$
AND $r2.COMPANY\_ID = r3.COMPANY\_ID$
UNION
SELECT $r2.NAME, r2.ADDRESS, r2.NATION$
FROM $scq4\ r1, scq2\ r2, scq3\ r3$
WHERE $\cdots$
UNION $\cdots$

- also in this step, relevant subqueries (EXPAtoms) are extracted from the expanded query. An EXPAtom is a Single Class Query, i.e., a query on a single Global Class of the BA-GVV.

- also in this step, relevant subqueries (EXPAtoms) are extracted from the expanded query. An EXPAtom is a Single Class Query, i.e., a query on a single Global Class of the BA-GVV.
- Query Unfolding: each EXPAtom is unfolded by taking into account the mappings in the BA ontology, so that it is rewritten w.r.t. the SINode GVVs

- also in this step, relevant subqueries (EXPAtoms) are extracted from the expanded query. An EXPAtom is a Single Class Query, i.e., a query on a single Global Class of the BA-GVV.
- Query Unfolding: each EXPAtom is unfolded by taking into account the mappings in the BA ontology, so that it is rewritten w.r.t. the SINode GVVs
- example:

scq1:
SELECT $CATEGORY\_ID$
FROM $Mould\_Making$
scq2:
SELECT $NAME, COMPANY_ID, CAPITAL_STOCK,$
$REGION, SUBCONTRACTOR, ADDRESS$
FROM $company$
WHERE $CAPITAL\_STOCK > 50$
AND $REGION\ LIKE\ VENETO$ AND $SUBCONTRACTOR\ LIKE\ yes$ scq3:
...

- the output to the QA contains: a SQL query (FDQuery) which computes object fusion, its atoms (FDAtoms) are Single Class Queries over the SINode GVV; the expanded atoms queries; and some residual functions

- the output to the QA contains: a SQL query (FDQuery) which computes object fusion, its atoms (FDAtoms) are Single Class Queries over the SINode GVV; the expanded atoms queries; and some residual functions
- example:

FDQuery:
SELECT $*$
FROM $FDAtom1$ OUTER JOIN $FDAtom2$
ON $(FDAtom1.COMPANY_ID = FDAtom2.COMPANY_ID)$

- at the QA, the Execution steps involves sending messages to SINodes for solving FDAtoms; results are stored in a DBMS

- at the QA, the Execution steps involves sending messages to SINodes for solving FDAtoms; results are stored in a DBMS
- then the Fusion steps executes the FDQuery overy these tables; results are stored as a view representing each EXPAtom

- at the QA, the Execution steps involves sending messages to SINodes for solving FDAtoms; results are stored in a DBMS
- then the Fusion steps executes the FDQuery overy these tables; results are stored as a view representing each EXPAtom
- finally, the Final Result is obtained by executing EXPQuery over these views

- at the QA, the Execution steps involves sending messages to SINodes for solving FDAtoms; results are stored in a DBMS
- then the Fusion steps executes the FDQuery overy these tables; results are stored as a view representing each EXPAtom
- finally, the Final Result is obtained by executing EXPQuery over these views
- by virtue of the separation between query expansion and query rewriting and evaluation, query processing is polynomial time in data complexity (i.e., with respect to the size of the data at the sources)

Talk Overview

1. general architecture
2. query tool
3. query process
4. system deployment
5. conclusions

- JADE was chosen as the underlying agent platform

- JADE was chosen as the underlying agent platform
  - FIPA compliant

- JADE was chosen as the underlying agent platform
  - FIPA compliant
  - Java based

- JADE was chosen as the underlying agent platform
  - FIPA compliant
  - Java based
  - free available under LGPL

- JADE was chosen as the underlying agent platform
  - FIPA compliant
  - Java based
  - free available under LGPL
  - well documented and supported

- JADE was chosen as the underlying agent platform
  - FIPA compliant
  - Java based
  - free available under LGPL
  - well documented and supported
- JADE agents consists of a set of cooperative non pre-emptive behaviors

- JADE was chosen as the underlying agent platform
  - FIPA compliant
  - Java based
  - free available under LGPL
  - well documented and supported
- JADE agents consists of a set of cooperative non pre-emptive behaviors
- common behavior scheduling is implemented in the libraries (cyclic, sequential, timed, etc)

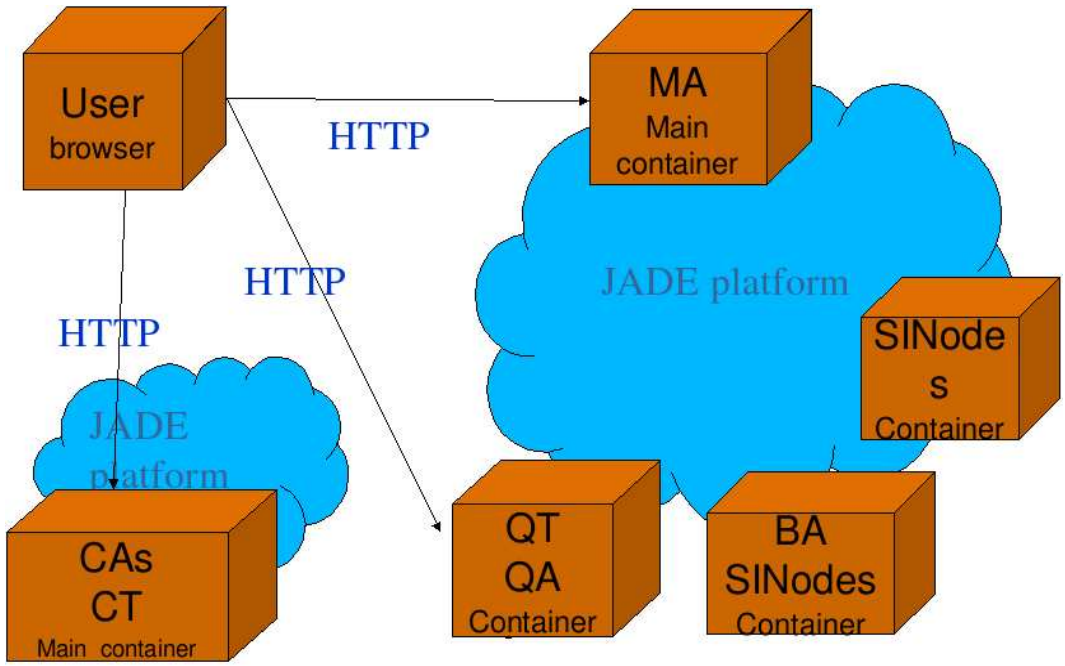
- JADE was chosen as the underlying agent platform
  - FIPA compliant
  - Java based
  - free available under LGPL
  - well documented and supported
- JADE agents consists of a set of cooperative non pre-emptive behaviors
- common behavior scheduling is implemented in the libraries (cyclic, sequential, timed, etc)
- communication between agents is done following standard FIPA interaction protocols

- a JADE multi-agent system is composed of a set of platforms; each platforms holds highly related agents

- a JADE multi-agent system is composed of a set of platforms; each platforms holds highly related agents
- a platform can be distributed over several hosts, each one running a container

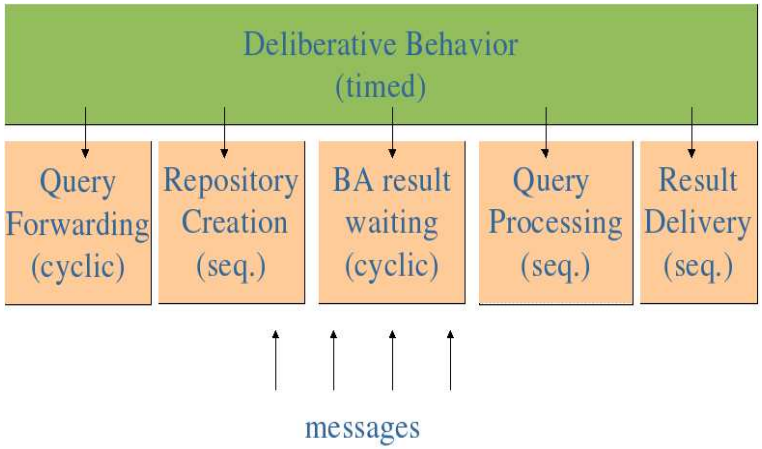
- a JADE multi-agent system is composed of a set of platforms; each platforms holds highly related agents
- a platform can be distributed over several hosts, each one running a container
- there must be a main container for each platform, where the agent services run

- the SEWASIE system is deployed as a JADE platform with the following configuration:

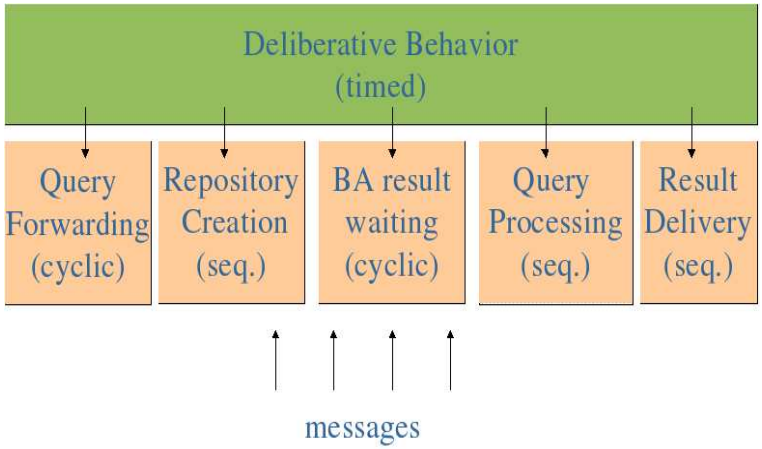


- a simple hybrid reactive-deliberative architecture is used in each agent

- a simple hybrid reactive-deliberative architecture is used in each agent

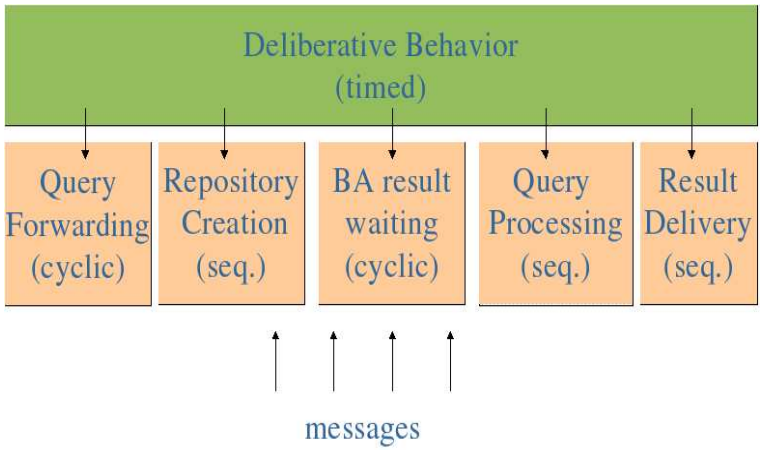


- a simple hybrid reactive-deliberative architecture is used in each agent



- thus, agent consist of a set of reactive behaviors for receiving and processing messages; these behaviours maps exactly the FIPA standard protocols the agent supports

- a simple hybrid reactive-deliberative architecture is used in each agent



- thus, agent consist of a set of reactive behaviors for receiving and processing messages; these behaviours maps exactly the FIPA standard protocols the agent supports
- and a single deliberative behavior that checks the agent state, and decides which protocol must be started, finished or aborted behaviors.

1. general architecture
2. query tool
3. query process
4. system deployment
5. conclusions

SEWASIE main features are:

SEWASIE main features are:

- a novel query formation paradigm, in which the ontology plays a fundamental role

SEWASIE main features are:

- a novel query formation paradigm, in which the ontology plays a fundamental role
- a technique for two-level data integration system, where the semantics of the schema (BA GVV, and SINode GVV, respectively) takes into account a novel technique (query expansion)

SEWASIE main features are:

- a novel query formation paradigm, in which the ontology plays a fundamental role
- a technique for two-level data integration system, where the semantics of the schema (BA GVV, and SINode GVV, respectively) takes into account a novel technique (query expansion)
- sound and complete, efficient algorithm

Conclusions

SEWASIE main features are:

- a novel query formation paradigm, in which the ontology plays a fundamental role
- a technique for two-level data integration system, where the semantics of the schema (BA GVV, and SINode GVV, respectively) takes into account a novel technique (query expansion)
- sound and complete, efficient algorithm
- agent technology used for distributed query processing, and expanded features integration

Conclusions

SEWASIE main features are:

- a novel query formation paradigm, in which the ontology plays a fundamental role
- a technique for two-level data integration system, where the semantics of the schema (BA GVV, and SINode GVV, respectively) takes into account a novel technique (query expansion)
- sound and complete, efficient algorithm
- agent technology used for distributed query processing, and expanded features integration
- play with it! `sewasie.ing.unimo.it:8080/sewasie/index.html`

1. general architecture
2. query tool
3. query process
4. system deployment
5. conclusions

Thank you!