Intelligent Database Systems

Book · January 2001

Source: DBLP

Elisa Bertino
Purdue University

Barbara Catania
Università degli Studi di Genova

Intelligent Database Systems

Barbara Catania

University of Genova (Italy)

IIWAS 2001 - Linz, Austria

Outline

- Introduction to Intelligent Database Systems (IDBs)
- Fundamental IDB approaches
- IDBs and their role in Web applications
- An IDB approach for metadata representation and retrieval
- Conclusions
- Bibliography

Introduction

What is an IDB?

DB technology:
- limited modeling capabilities
- new data management applications

AI techniques:
- often toy systems
- no persistent management of data

Late '80s/'90s

DB techniques can aid an AI system to deal with large amount of information

IDB Technology

AI techniques can provide semantic support to a DB system

Characteristics of IDBs

- Architecture based (at least implicitly) on an organization in the Expert Systems (ESs) style
  - Fact DataBase (FDB) + Rule Base (RLB)
- Use of AI techniques
  - Knowledge representation techniques
    - semantic data representation
  - Inference techniques
    - improved reasoning about data
  - Intelligent user interfaces
    - help users to make requests and receive replies
- persistency of the FDB

A traditional taxonomy of IDBs

Efforts originated in a DB context

Static extensions
extending the expressive power of traditional DB data models

Dynamic extensions
introducing some form of reasoning inside DBMSs

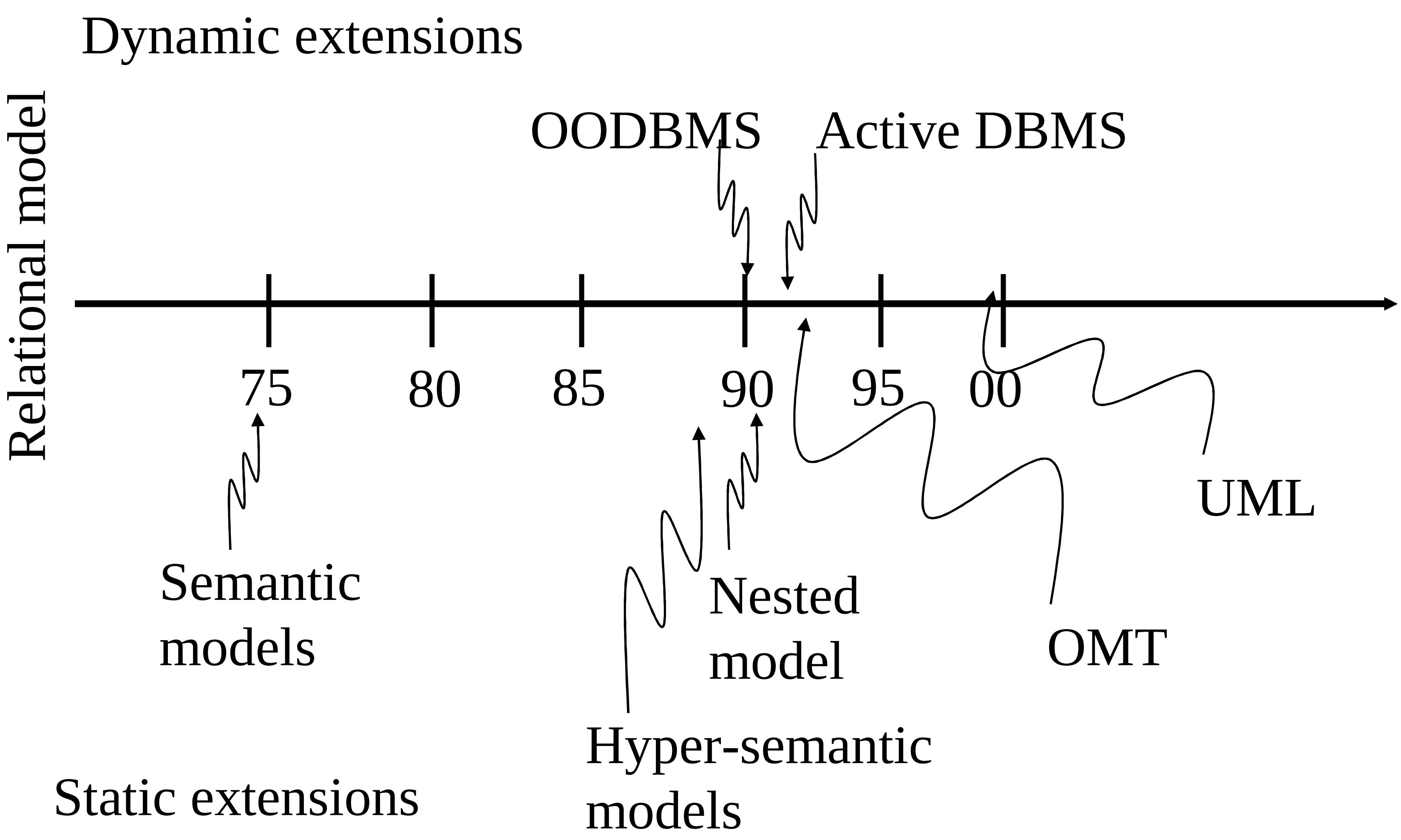
Efforts originated in a AI context

Basic solutions
coupling knowledge-based systems and DBMSs

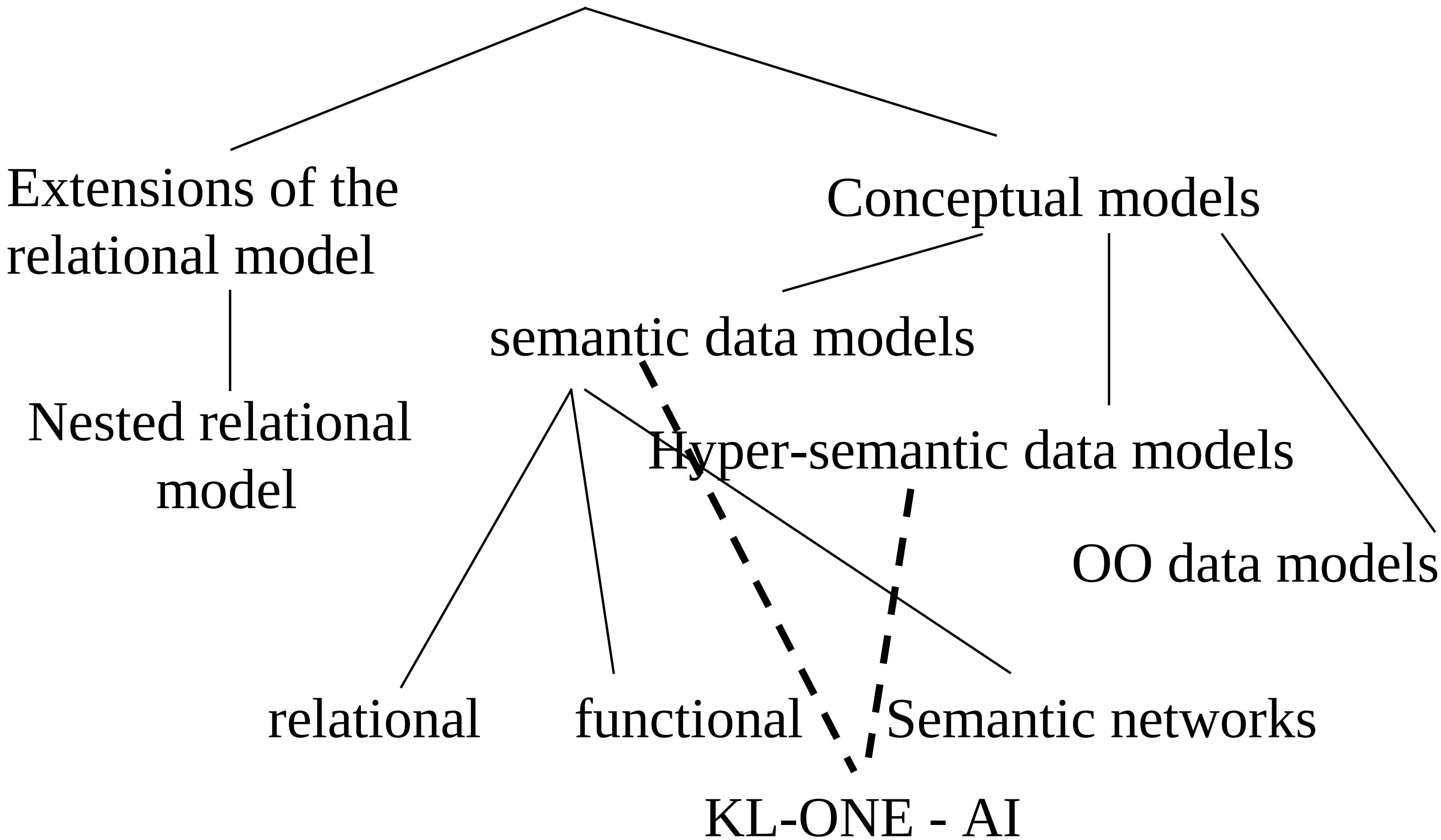
Advanced solutions
attempt to use AI systems to deal directly with large amount of information

Fundamental IDB approaches

Efforts originating in a DB context



Static extensions



Dynamic extensions

Introducing programming languages constructs

OODBMS

Frame-based systems - AI semantic data models

Introducing active rules

Active DBMS

Production rules - AI

Efforts originating in an AI context

- Based on the notion of Knowledge Based system (KBS)
- KBSs typically contain:
  - explicitly represented rules RLB
  - simple facts FDB
  - components which can make inferences over the Knowledge Base KB = RLB + FDB
- the information dealt with by the KBS consists therefore of:
  - explicitly stored facts and rules
  - derived facts

Knowledge Based System types

- Pure rule-based representations supporting inference by resolution
  - systems developed in a logic programming context
  - ES shells based only on the production rule paradigm
- Pure frame- or object-based representations supporting inference by inheritance
  - frame systems
  - terminological (description logic) systems
  - KESE: hybrid systems, commercially available, supporting alternative inference methods and representation schemes (SPOKE, KEE)

An overall view

KBS type		Formalism	Reasoning
Logic programming systems		logical clauses	resolution principle
Expert Systems (ES)		production rules	procedural, logic
Inheritance-based systems		ontology of concepts, inheritance hierarchies	inheritance
	Frame-based systems	structured concepts	inheritance
	Terminological systems	terminological knowledge, assertional knowledge, based on description logic	classification
KESE		various approaches	various approaches

Tradeoff

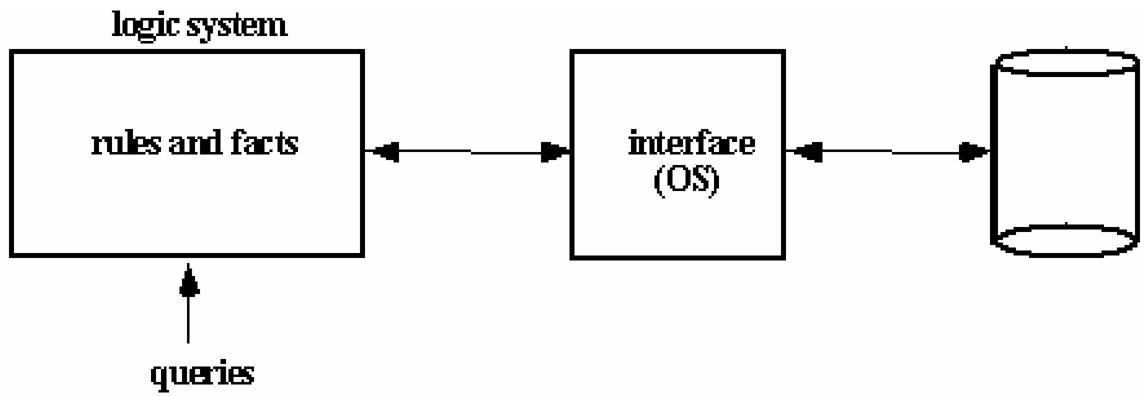
- Tradeoff between: computational complexity expressive power, and completeness
  - sound, complete, tractable but limited expressive power
    - KRYPTION, CLASSIC
  - sound, complete, intractable
    - KRISL
  - sound, higher expressive power, intractable and incomplete (more efficient)
    - BACK,NIKL, LOOM

KBS and DBMS

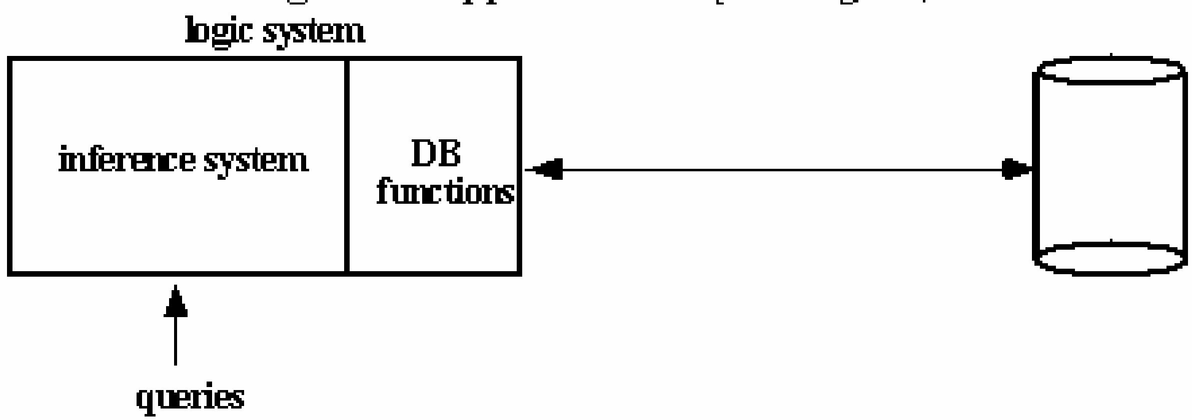
- Conventional KBSs are inadequate for supporting new data/knowledge-intensive applications
- Problems:
  - KBSs usually deal with knowledge bases of small size, in volatile memory
  - KBSs provide only limited DBMS services
- Possible solution: coupling KBSs with DBMSs
  - Coupling of logic programming systems with DBMSs
    - Deductive databases
  - coupling of ES shells and KESEs wih DBMSs
    - five classes of approaches

Deductive databases

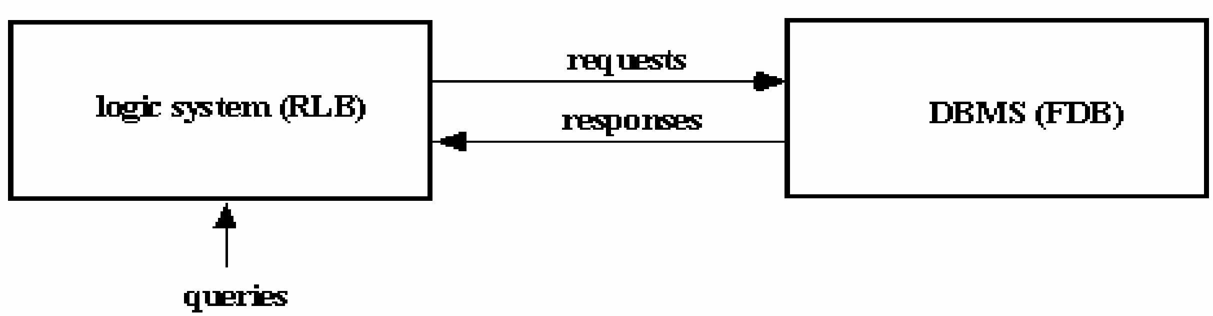
- Intensional database (IDB), containing logic formulas
  - IDB: parent(X,Y) <- father(X,Y)
    parent(X,Y) <- father(X,Z), parent(Z,Y)
- Extensional database (EDB), containing base relations
  - EDB: father(ann,john)father(john, mark)
- Through logic inference mechanisms, derive, from base relations, information not explicitly stored in the EDB
  - father(ann,john), father(john, mark),
  - parent(ann,john), parent(john, mark), parent(ann,mark)
- Language typically used for IDB:
  - Datalog (restriction of Prolog, set-oriented)
- formal theoretical foundation



Homogeneous approach : a *pure logic system*



Homogeneous approach : an *enhanced system*

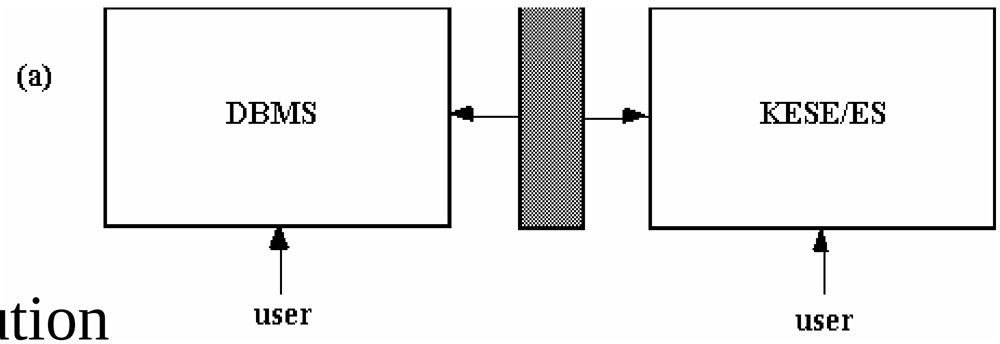


Heterogeneous approach

Coupling ES shells and KESEs with DBMSs

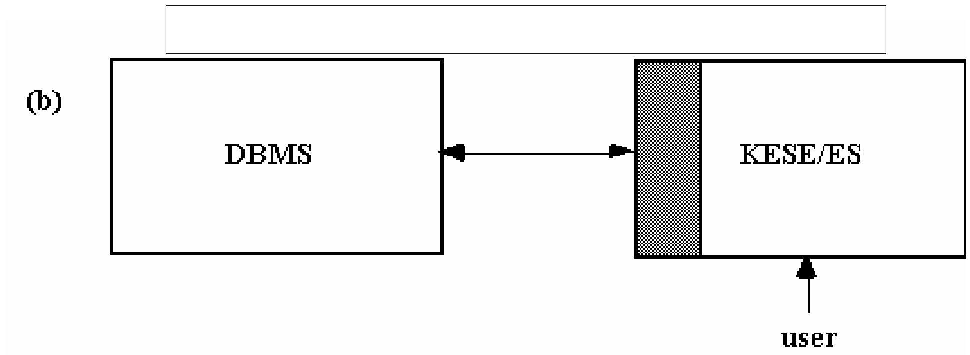
- No theoretical foundations
- mismatch between ES shell/KESEs and DBMS
    - semantic, impedence, and granularity mismatch
- most proposals for KESEs does not give rise to real IDBs
    - useful for hystorical motivations
    - they represent the basic approaches of IDB architecture
- DB used to store AI objects:
    - AI objects are translated into and out of DB objects
    - AI objects are stored in their native format in the DB (for example, as LOB)

Coupling ES shells and KESEs with DBMSs



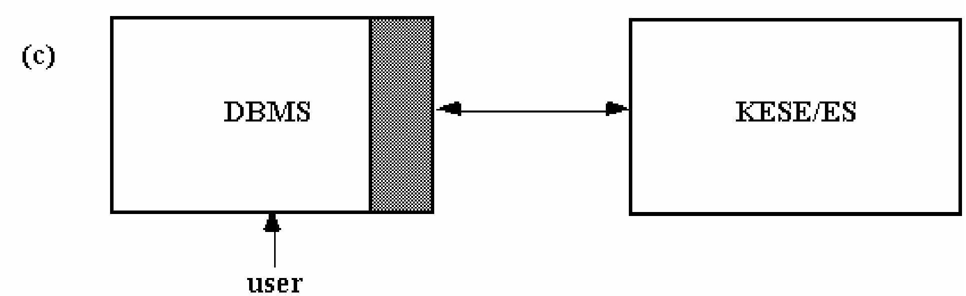
- Full-bridge solution
- often flat file as intermediate medium
- the control of the interactions and the processing can be located on the central bridge or distributed
- Such architecture does not scale up well
- Examples:
  - DIFEAD (ESs, rel. DBMSs, intermediate data dictionary)
  - KADBASE (ESs, rel. DBMSs, distributed)
  - Europe-Bruke approach (BACK, rel. DBMSs)

Coupling ES shells and KESEs with DBMSs



- Extension of a KB with components proper to a DBMS
- used mainly for KBs based on the logical approach
- adopted by the vendors of the main ES tools to provide their systems with some elementary possibilities of extracting information from a database
- Examples: ROCK, KBMS, SPOKE
- no standard approach exists for realizing the access functions

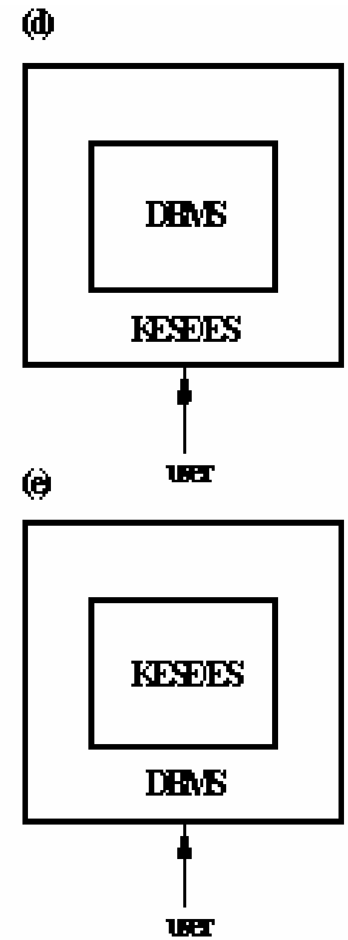
Coupling ES shells and KESEs with DBMSs



- Extension of a DBMS with components proper to a KBS
- Two possible interactions:
  - explicit access procedure: an explicit call to the KBS is inserted in the application program
  - implicit access procedure: the access to the inference engine is through the same query interface used to access data
- Similar to rule based systems and OODBMSs

Coupling ES shells and KESEs with DBMSs

- DB and KBS systems are strongly integrated
  - only one environment
  - no semantic mismatch problems
- Architecture d):
  - construction of a DB system after (or during) the set up of the KBS
  - integral approaches
- Architecture e):
  - the DBMS technology is more stable and mature than the KBS technology, and the installed base of DBs is definitely larger than the KBSs base
  - DBs are probably a better place for incorporating ES functionalities than vice versa
  - Examples: ARCHES, KBase



Integral approach

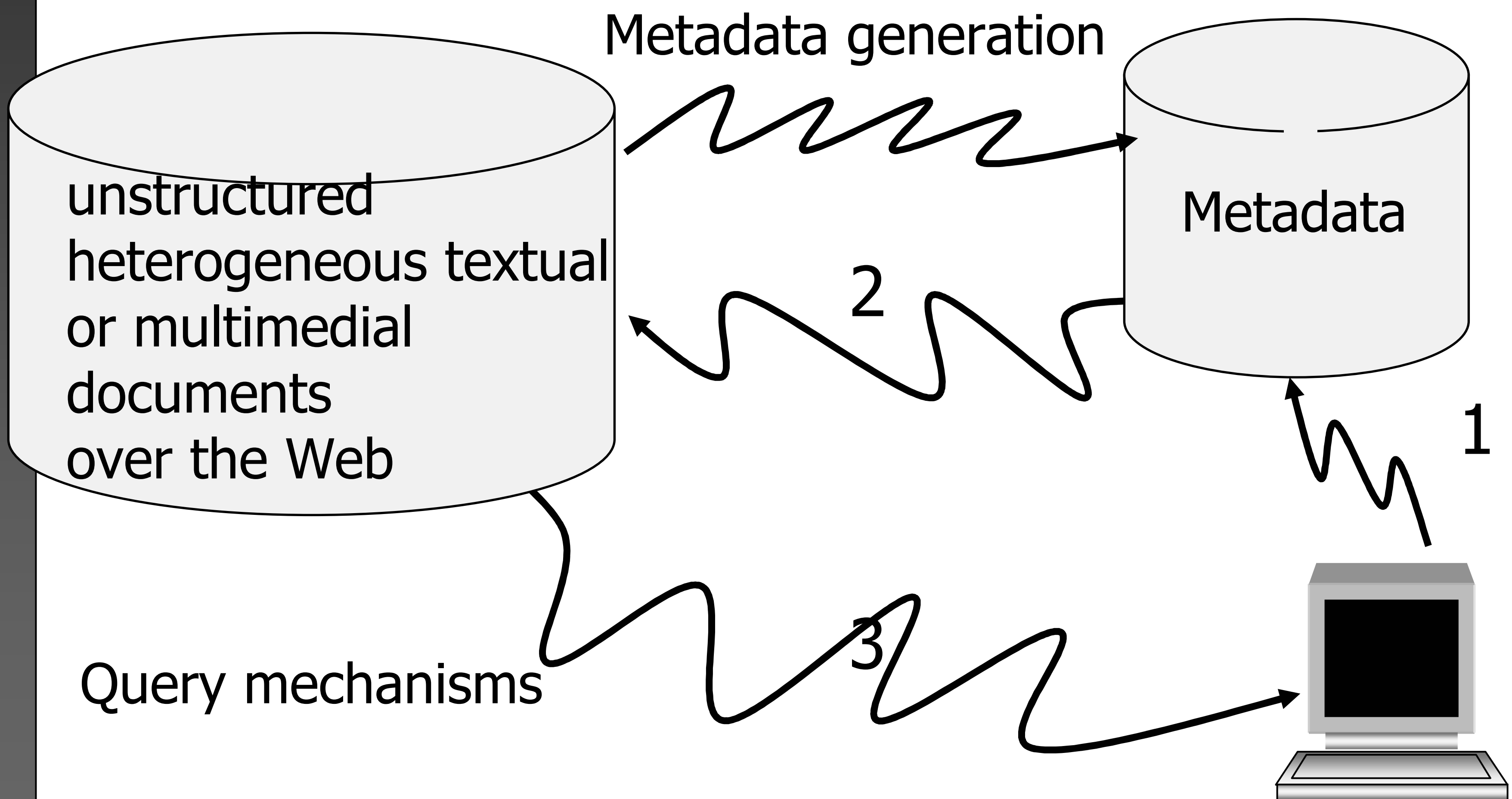
- Example of type d) architecture:
  - only one pure KBS environment
  - data model is some sort of AI knowledge representation language
  - all sort of inference techniques are used
  - KBS environment should be able to support DBMS services
- attempt to use some kind of AI system to deal directly, in a DBMS style, with large quantities of persistent information
- no theoretical foundation
- Example: TELOS, CYC, NKRL, lexical approaches (WordNet)
- Limitations:
  - great variety of knowledge representation models
  - complexity of the used formalisms
  - lack in supporting DBMS functionalities

IDBs and their role in Web applications

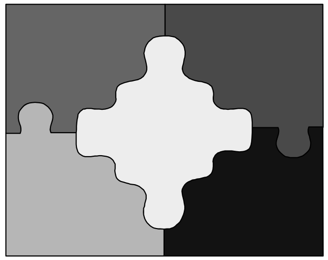
Some applications

- Metadata representation
- Integration of heterogeneous sources
- Web application design

Metadata representation: problem



Metadata representation: an IDB approach



Unstructured (possibly multimedia) document

Natural Language (NL) caption

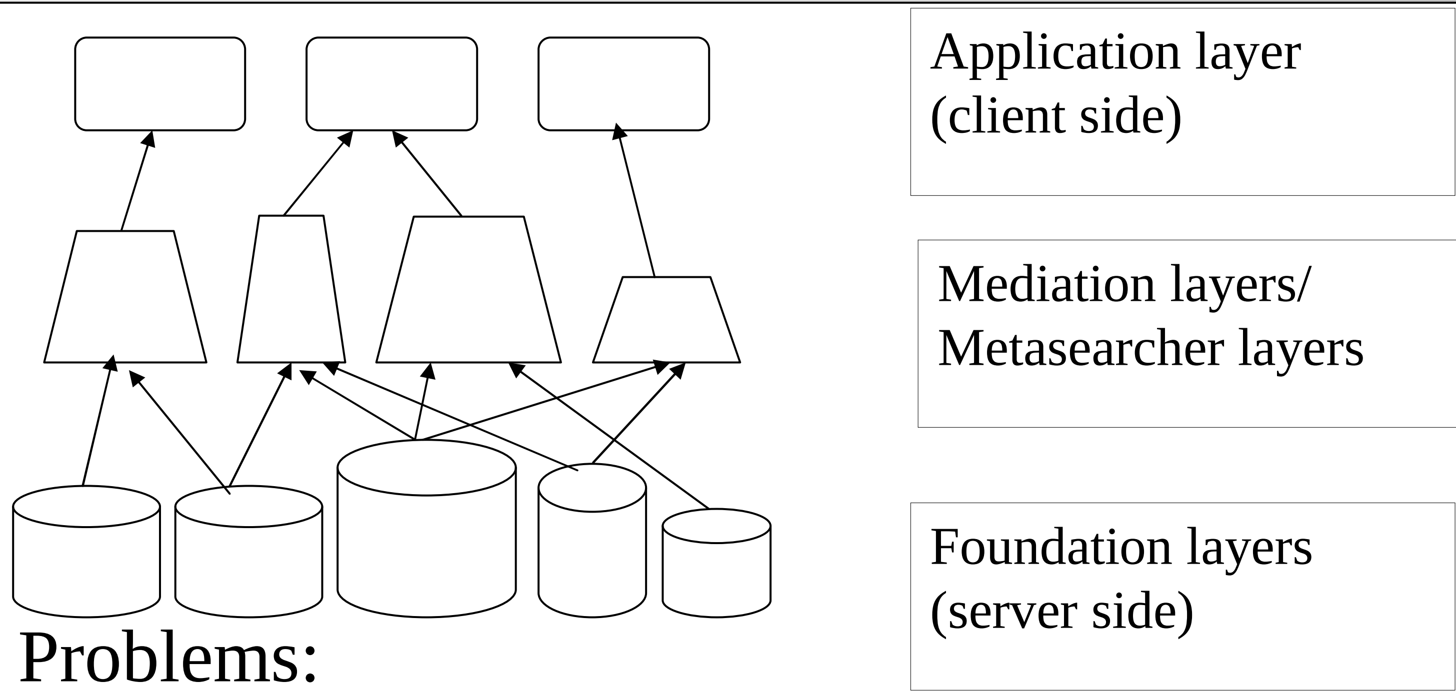
Annotation in a Knowledge representation language

- General solution for the mixed media access problem
  - texts
  - images
  - pictures
  - …
- support similarity-based indexing
  - similar caption = similar documents

Metadata representation: examples

- Solutions based on the illustrated approach have been proposed, among the others, in:
  - CYC
  - NKRL (see later)
- a solution based on TELOS has also been proposed to construct and manage an API for a metadata repository

Integration: problem



- Problems:
  - how is it possible to represent a global domain model?
  - how is it possible to represent the local knowledge?
  - how is it possible to map global queries into local queries and merging results?

Integration: an IDB approach

- Usage of knowledge representation languages for representing:
  - domain model
  - heterogeneous sources
  - query mapping
- important role played by ontologies
- advantages:
  - clear formal and declarative foundation
  - powerful reasoning facilities

Integration: examples

- Carnot project (MCC):
  - integration of heterogeneous sources using a set of articulation axioms that describe how to map SQL queries and domain concepts
  - articulation axioms built in CYC
- SIMS (University of Southern California):
  - LOOM is used both to represent the global domain model and the local heterogeneous sources characteristics
- TSIMMIS (Stanford University):
  - inheritance-based language (OEM) to describe sources
  - a logic OO-language is used to specify mediators as views upon OEM sources (LOREL)
- Garlic (IBM)
  - ODMG as model for sources and programming interfaces

Web application design: problem

- Web applications are characterized by three main design dimensions:
  – structure
  – navigation
  – presentation
- Problems:
  – which models can be used to support the development of Web applications in all the lifecycle steps?

Web application design: approach

- Conceptual level:
  - Structural modeling:
    - semantic/hypersemantic data models
    - OO models
  - Navigation:
    - techniques proposed for the more general problem of human-computer interaction specification
      - first-order logic, Petri Nets, finite state machines, ...
  - Presentation:
    - software tools and formal methods
- Design level:
  - structured or semi-structured data models

Web application design: examples

- WebML [Politecnic of Milano, Italy]
- Araneus [University of Rome, Italy]
- Strudel [At&T, INRIA, Univ. Washington]

An IDB approach for metadata representation and retrieval

Joint work with E. Bertino and G.P. Zarri

To better explain ...

- A specific problem concerning Web applications
- a concrete approach
  - example of an integral approach (NKRL)
  - example of KESE (CONCERTO)
  - example of type b) architecture (Knowledge Manager)
  - important problems (standardization, DBMS facilities)

Metadata

- Machine-understandable knowledge that describes the properties and the relationships of Internet resources
- To be used to get information about the structure and the contents of these resources
- Different classes:
  - Structure-based metadata: external characteristics of the support (color, shape, texture, motion, etc.)
  - Content-specific metadata: representing the meaning of documents
    - keywords
    - conceptual annotations

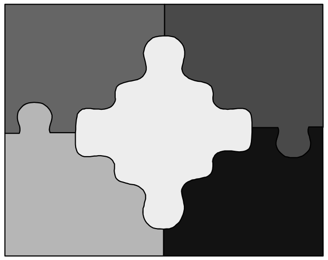
Conceptual annotations

- Structured information, describing in depth the semantic meaning of a document
- several proposals:
  - UNTANGLE, MIHMA, Information Manifold, Ontobroker
- often based on description logic
- limitations:
  - often unable to describe complex events
  - not always adequate to describe actions, facts, events
  - automatic extraction quite difficult
- alternative approach: NKRL [Zarri, ‘94-’00]

Why NKRL ?

- NKRL: Narrative Knowledge Representation Language
- Ability to represent, through ontologies, both:
  - the important notions of a given application domains (concepts)
  - mutual relationships between concepts (facts, events)
- ability to (partially) automatically extract conceptual annotations in NKRL by using tools developed in two European projects:
  - NOMOS (Esprit P5330)
  - COBALT (LRE P61011)
- the proposed solution fluctuates between:
  - very simple, low-level rule-based techniques making use of elementary semantic categories like those included in WordNet
  - very complex inference-intensive applications of CYC

The proposed approach



Unstructured
(possibly multimedia)
document

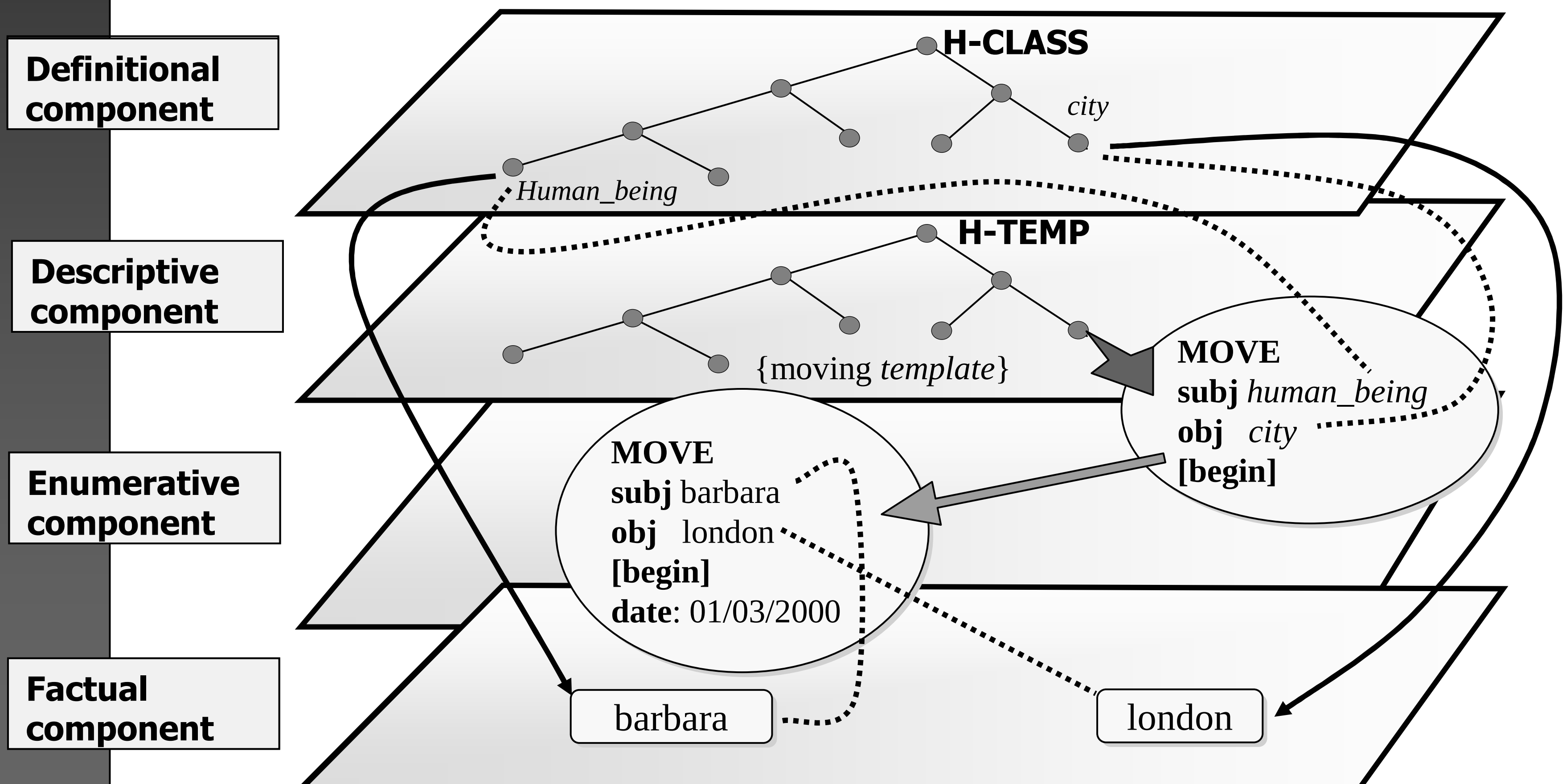
Natural Language
(NL) caption

Annotation in NKRL

- General solution for the mixed media access problem
  – texts
  – images
  – pictures
  – …
- support similarity-based indexing
  – similar caption = similar documents

NKRL

On March 1st, 2000, Barbara will go to London



Definitional component

- Supplies the tools for representing the important notions (concepts) of a given domain
- a concept is a frame-based structure composed of
  - OID
  - symbolic label like *physical_entity*, *human_being*, *city*, etc.
  - a set of characteristics features
- concepts are represented by using an ontology of terms, called HCLASS
- general concepts belonging to the upper levels of are represented inside a catalogue and are assumed to be invariable
- similarities with terminological languages

Enumerative component

- It is composed of all the instances of sortal concepts, called individuals
- non sortal concepts does not admit direct instances
- similarly to concepts, individuals are represented as frame based structures
- Example:
  - chair27
  - paris_
  - lucy_

Descriptive component

- It contains the description of the events proper to a given domain
- supplies the tools used to produce the formal representations (predicative templates) of general classes of narrative events, like 'moving a generic object', 'formulate a need', 'be present somewhere'
- Templates are structured into an inheritance hierarchy, HTEMP, corresponding to a taxonomy of events
- Basic templates (more than 150) are described in a catalogue
- By means of proper specialization operations, it is possible to obtain from the basic templates the derived templates needed to implement a particular application

Descriptive component

- Templates are characterized by a threefold format:

  $(P_i\ (R_1\ a_1\ )(R_2\ a_2\ )\ \ldots\ (R_n\ a_n\ ))$

  – P_i denotes the symbolic label identifying the template (class of events)

  – R_k , k = 1,…, n, denote generic roles

  – a_k , k = 1,…, n, denote the arguments associated with the roles (concepts, instances, pred. occ.)

- Predicates: BEHAVE, EXIST, EXPERIENCE, MOVE, OWN, PRODUCE, RECEIVE
- Roles: SUBJ(ect), OBJ(ect), SOURCE, DEST(ination), MODAL(ity), TOPIC, CONTEXT

Factual component

- Concerns the instances (predicative occurrences) of the predicative templates
  – representation of single, specific events

- Examples:
  – Tomorrow, I will move the wardrobe
  – Lucy was looking for a taxi

Example

Milan, October 15, 1993. The financial daily Il Sole 24 Ore reported Mediobanca had called a special board meeting concerning plans for capital increase.

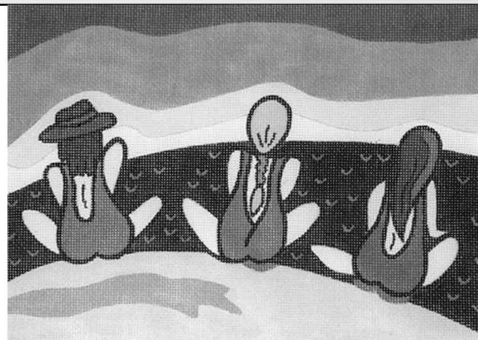
c1) MOVE SUBJ (SPECIF sole_24_ore *financial_daily*): (milan_)
OBJ #c2
date-1: 15_october_93
date-2:

c2) PRODUCE SUBJ mediobanca_
OBJ (SPECIF *summoning_*
(SPECIF board_meeting_1 mediobanca_ *special_*))
TOPIC (SPECIF plan_1 (SPECIF *cardinality_ several_*)
capital_increase_1)
date-1: circa_15_october_93
date-2:

Advanced representation facilities

- Structured arguments built up making use of a specialized sublanguage (AECS), including four expansion operators:
  - disjunctive (ALTERNative = A)
  - distributive (ENUMeration = E)
  - collective (COORDination = C)
  - attributive (SPECIFication = S)
- ability to bind predicative occurrences together
  - binding occurrences

Application to multimedia documents



Three nice girls are lying on the beach

NL caption

C1) EXIST SUBJ (SPECIF girl_1 *nice_*
(SPECIF *cardinality_* 3)): (beach_1)
MODAL *lying_position*

[girl_1
InstanceOf: *girl_*
HasMember: 3]

NKRL annotation

Queries in NKRL

- Query are expressed through search patterns
- It must be possible to specify:
  - perfect match (identical structure)
  - perfect match apart from cardinality (identical structure apart from the cardinality of AECS lists)
  - subsumed match (information globally congruent from a semantic point of view - e.g., additional SPEFIC lists are possible-)
- automatic transformation of queries into similar queries

Example

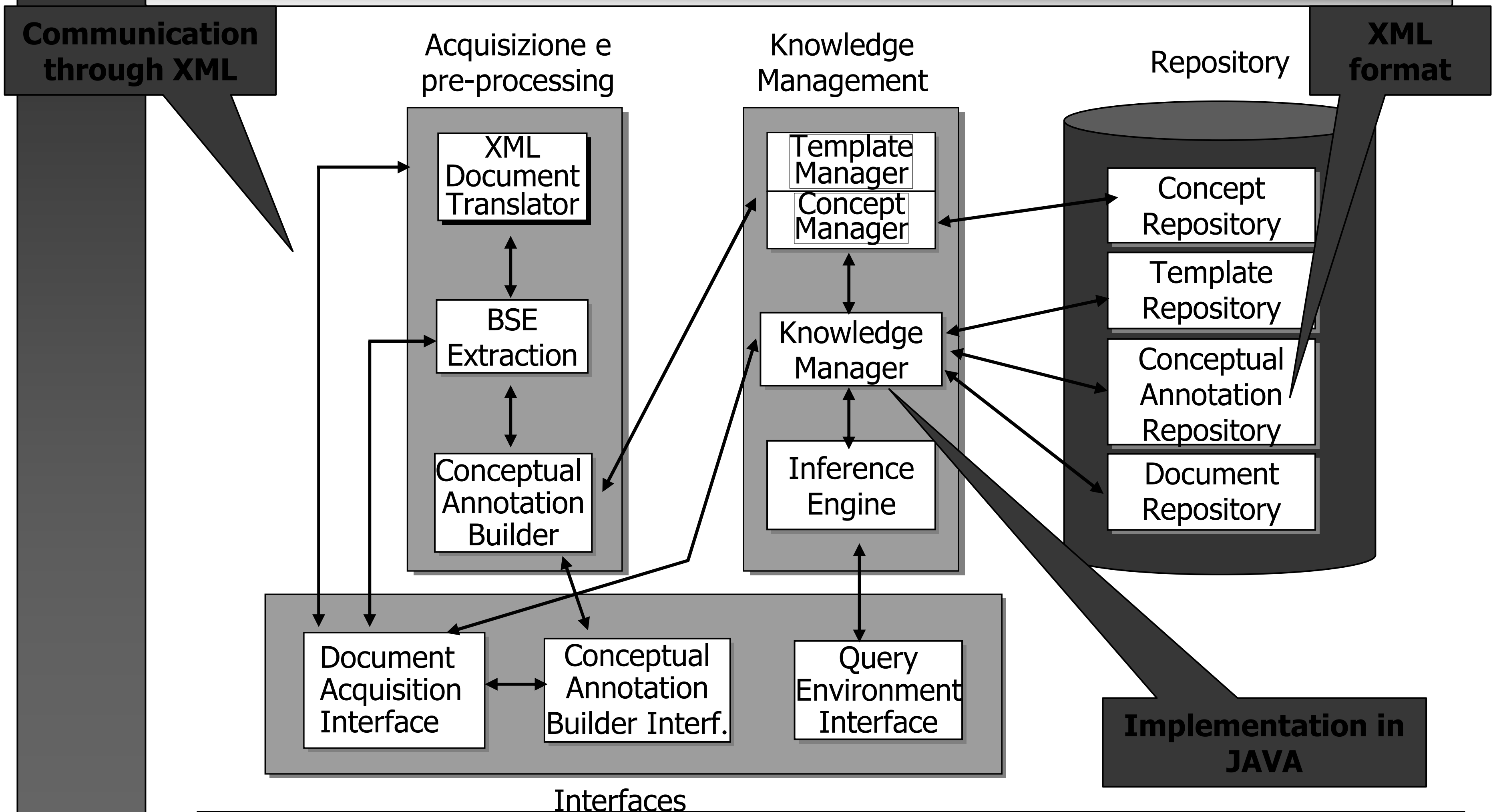
Which was the theme of the recent board meeting called out by Mediobanca ?

((?w IS-OCCURRENCE
    :predicate PRODUCE
    :SUBJ mediobanca_
    :OBJ (SPECIF ?x (SPECIF ?y mediobanca_))
    :TOPIC ?z)
    (1_october_93, 20_october_93)
    ((?x IS-A (:OR *assembly_ adjournment_ dissolution_*))
    (?y IS-A *board_meeting*)
    (?z IS-A *planning_activity*)))

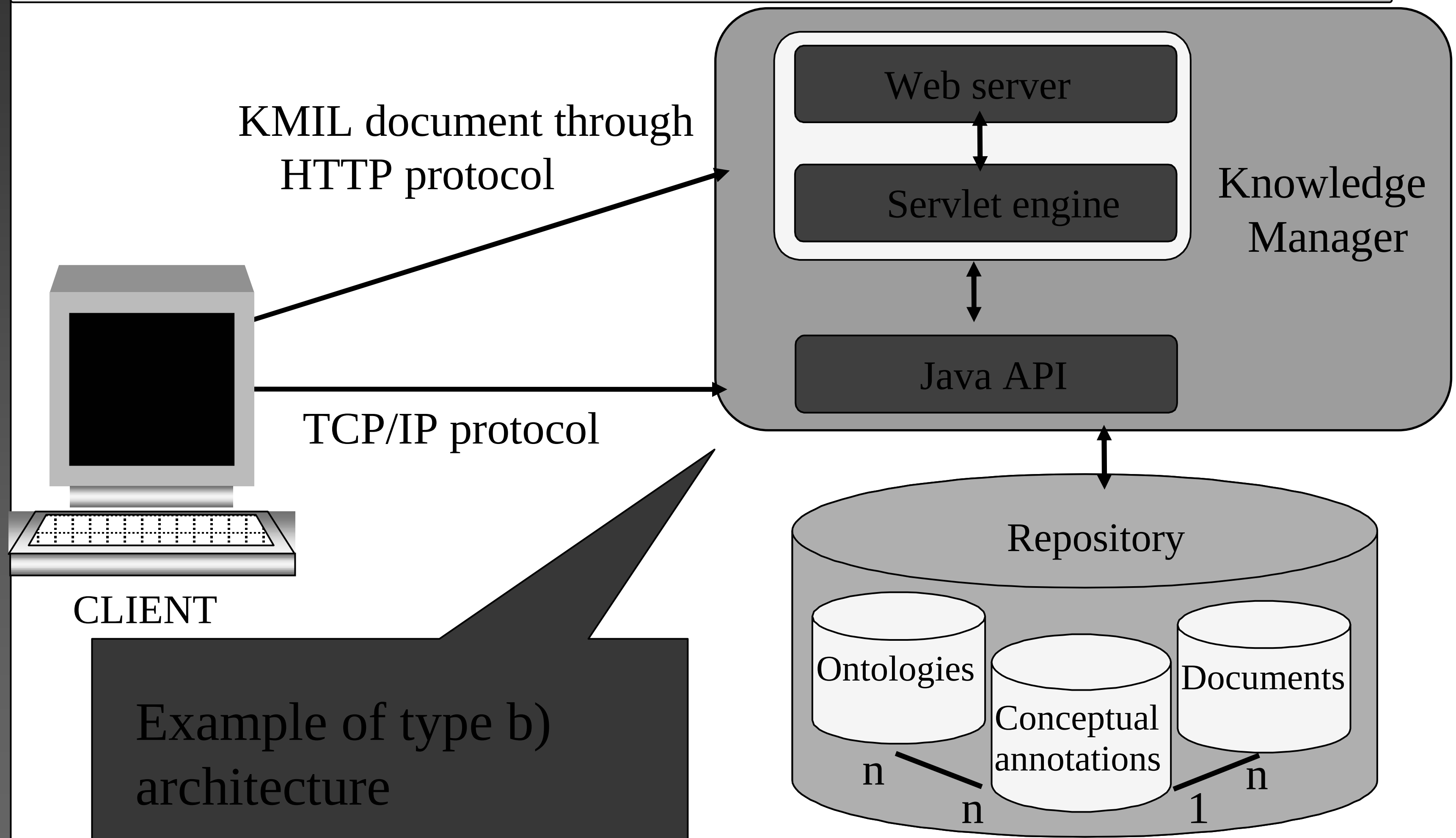
The CONCERTO Esprit Project

- The previous ideas have been implemented in the context of the CONCERTO Esprit Project
- only textual, possibly semi-structured (HTML, XML) documents
- the architecture can be extended to deal with multimedia documents

The CONCERTO KESE architecture



The KM architecture



Technological choices

- How to represent conceptual annotations
- How to implement the repositories
- How to communicate with the Knowledge Manager

Ontologies and Conc. Ann. Representation

- Ontologies:
  - linearization of the hierarchies in a set of tables
- Conceptual annotations:
  - Traditional implementation: three-layered approach:
    - Common Lisp + a frame/object oriented environment + NKRL
  - To increase the standardization:
    - Java + RDF (Resource Description Format)
    - implemented in XML

RDF

- RDF (W3C): proposal for defining and processing WWW metadata
- model based on directed labelled graphs
  - nodes represent Web resources
  - described by using attributes
  - edges represent relationships between resources
- no predefined vocabulary (ontologies, keywords,…) exists
- model implemented in XML

Problems mapping NKRL in RDF

- RDF structures: dyadic
  - two resouces are linked by a binary conceptual relation under the form of a property
- NKRL structures: threefold relationship
  - symbolic label
  - predicate
  - one or more roles and fillers
- NKRL structures have been transformed in dyadic structures and mapped in RDF

An example of RDF representation

```
<?xml version=1.0 ?>
<!DOCTYPE DOCUMENTS SYSTEM CA_RDF.dtd>
<CONCEPTUAL_ANNOTATION>
  <rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
           xmlns:ca=http://projects.pira.co.uk/concerto#>
    <rdf:Description about=occ11824>
    <rdf:type resource=ca:Occurrence/>
       <ca:instanceOf>Template43</ca:instanceOf>
       <ca:predicateName>Move</ca:predicateName>
       <ca:subject rdf:ID=Subj43 rdf:parseType=Resource>
          <ca:filler>barbara_</ca:filler>
       </ca:subject>
       <ca:object rdf:ID=Obj43 rdf:parseType=Resource>
          <ca:filler>london<ca:filler>
       </ca:object>
       <ca:listOfModulators>
          <rdf:Seq><rdf:li>begin</rdf:li></rdf:Seq>
       </ca:listOfModulators>
       <ca:date1>01/03/2000</ca:date1>
       </rdf:Description>
   </rdf:RDF>
  </CONCEPTUAL_ANNOTATION>
```

Repository implementation

- Conceptual annotations are represented in XML
- Two possible usages of XML documents:
  - Data Centric: such documents represent the tool by which traditional data are transferred over the Web
    - XML as a vehicle for data transfer
    - Example:sales orders, flights scheduling,...
  - Document Centric: the information is represented by the document itself
    - XML as a model for data representation
    - Example: books, textual documents, metadata
- In CONCERTO:
  - document centric XML documents

XML and DBMS

- Two categories of DBMS:
  - XML-Native DBMS: architecture designed for totally supporting management of XML documents
    - not yet very robust
    - useful for Document Centric documents
    - Example: eXcelon (Object Design Inc.)
  - XML-Enabled DBMS: all DBMS that extend their architecture with functionalities proper to the management of XML documents
    - Object-Relational (DB2, Oracle8i,…), relational (Microsoft SQL Server)
    - useful for Data Centric and partially for Document Centric documents

Communication protocol

- Standard communication protocol
  - Knowledge Manager Interface Language (KMIL) for interacting with the KM
  - implemented in XML

- The Knowledge Manager can be hosted on a generic machine, becoming independent from the other modules of the architecture

An example of KMIL input

```
<?xml version=1.0?>
 <!DOCTYPE KMIL-SESSION SYSTEM KmilIn.dtd>
  <KMIL-SESSION>
   <KMIL-ACTION serial_number=1>
     <KMIL-INSERT-PredOcc IdPO=occ11824 Doc=doc132>
         <TEXT> RDF Text </TEXT>
     </KMIL-INSERT-PredOcc>
   </KMIL-ACTION>
   <KMIL-ACTION serial_number=2>
     <KMIL-INSERT-PredOcc IdPO=occ11845 Doc=doc133>
         <TEXT> RDF Text </TEXT>
     </KMIL-INSERT-PredOcc>
   </KMIL-ACTION>
  </KMIL-SESSION>
```

An example of KMIL output

```
<?xml version=1.0?>
 <!DOCTYPE KMIL-SESSION SYSTEM KmilOut.dtd>
  <KMIL-SESSION>
     <KMIL-ACTION-OUTPUT serial_number=1
                         action_status = OK>
     </KMIL-ACTION>

     <KMIL-ACTION serial_number=2
                  action_status = ERROR>
          <ERROR-CODE code = KMIL-ERR-08/>
     </KMIL-ACTION>
  </KMIL-SESSION>
```

On-going work

- Efficient management of conceptual annotations on persistent storage
  - clustering
  - optimization/indexing
  - security
- Strongly related to XML document management
  - initial work on clustering and caching

Conclusions

Ideal IDBs

- Like a DBMS:
  - persistent storage management
  - support of query and update languages
  - support of indexing and query optimization techniques
  - concurrency control and recovery
  - security
- Like advanced data models:
  - like nested models: non-atomic attribute values
  - like semantic data models and OODBMS: abstraction, inheritance
  - like hyper-semantic data models: no real distinction between data and knowledge
  - like active DB: reaction

Ideal IDBs

- Like more advanced KESEs:
  - support of various inference techniques: deductive, abductive, nonmonotonic, probabilistic, analogical
- Like any state-of-the-art DBMS, ES shell or KESE:
  - sophisticated user interfaces as well as knowledge and application engineering tools
- Like (some) advanced systems in the AI style:
  - uniform, high-level type of representation (frames, objects, semantic networks, hybrid representation schemata …) in both the Rule Base and the Fact DataBase

Recent trends

- The techniques analyzed before are now at the basis of several research directions
- macroscopic directions:
  - Advanced data models
  - advanced reasoning
  - advanced architectures
  - advanced index and retrieval techniques

An extended taxonomy of IDBs

- Advanced data models
  - ...
  - temporal DBMS
  - semistructured and unstructured data representation
  - Ontologies
- advanced reasoning
  - ...
  - temporal DBMS
  - query languages for semistructured and unstructured data
  - data mining
- advanced architectures
  - heterogeneous systems
  - cooperative systems (multi-agent systems)
- advanced indexing techniques
  - indexing and retrieving advanced data
  - Internet indexing and retrieval techniques

Bibliography

Efforts in the DB context

BERTINO, E., MARTINO, L. (1993). Object-Oriented Database Systems - Concepts and Architectures, Addison-Wesley.

BOOCH, G., RUMBAUGH, J., AND JACOBSON, I. (1998). The Unified Modeling Language User Guide. Addison-Wesley, 1998.

BUNEMAN, P. (1997). Semi-Structured Data. Tutorial at ACM SIGART-SIGACT-SIGMOD Int. Conf. on Principles of Database Theory, 1997.

CERI, S., GOTTLOB, G., AND TANCA, L. (1990) Logic Programming and Databases, Springer-Verlag, Berlin (Germany), 1990.

CHEN, M.S., HAN, J., AND YU, P.S. (1996). Data Mining: An Overview from a Database Perspective. IEEE Transactions on Knowledge and Data Engineering, 8(6):866-883, 1996.

GYSSENS, M. AND GUCHT, D.V. (1991). A Comparison between Algebraic Query Languages for Flat and Nested Databases. Theoretical Computer Science, 87:263-286, 1991.

Efforts in the DB context

HULL, R. AND KING, R. (1987) Semantic Database Modeling: Survey, Applications, and Research Issues, ACM Computing Surveys, 19, 201-260.

PECKHAM, J. AND MARYANSKI, F. (1988). Semantic Data Models. ACM Computing Surveys, 20(3):153-189, 1988.

POTTER, W.D., TRUEBLOOD, R.P., AND EASTMAN, C.M. (1989). Hyper-Semantic Data Modelling. Data & Knowledge Engineering, 4:69-90, 1989.

POTTER, W.D., and TRUEBLOOD, R.P. (1988) Traditional, Semantic and Hyper-semantic Approaches to Data Modeling, IEEE Computer, 21, n° 6, 53-64.

RUMBAUGH, J. ET AL. (1991). Object Oriented Modeling and Design, Englewood Cliffs, N.J., Prentice Hall, 1991.

WIDOM, J. AND CERI, S. (1996). Introduction to Active Database Systems. In Active Database Systems: Triggers and Rules for Advanced Systems (J.Widom, S. Ceri, eds.), Morgan Kaufmann Publishers, San Francisco (Calif.).

Efforts in the AI context

BRACHMAN, R.J., AND LEVESQUE, H.J. (1985) A Fundamental Tradeoff in Knowledge Representation and Reasoning, in Readings in Knowledge Representation and Reasoning, Brachman, R.J., and Levesque, H.J., eds. San Francisco: Morgan Kaufmann

BREWKA, G. (1987) The Logic of Inheritance in Frame Systems, in Proceedings of the Tenth International Joint Conference on Artificial Intelligence - IJCAI/87. San Francisco: Morgan Kaufmann.

CHANDRASEKARAN, B., JOSEPHSON, J.R., AND BENJAMINS., V.R. (1999) Ontologies: What are they? Why do we need them? IEEE Intelligent Systems and Their Applications, 14(1):20-26, 1999. Special Issue on Ontologies.

ECAI. Proc. of the European Conference on Artificial Intelligence.

FORGY, C.L. (1995) The OPS Languages : An Historical Overview, PC AI 9(5), 16-21.

Efforts in the AI context

FRIDMAN NOY, N., AND HAFNER, C.D. (1997) The State of the Art in Ontology Design - A Survey and Comparative Review, AI Magazine, 18(3), 53-74.

KRAMER, B.M., CHAUDHRI, V.K., KOUBARAKIS, M., TOPALOGLOU, T., WANG, H., AND MYLOPOULOS, J. (1991) Implementing TELOS, ACM SIGART Bulletin, 2(3), 77-83.

LENAT, D.B., GUHA, R.V., PITTMAN, K., PRATT, D., AND SHEPHERD, M. (1990) CYC : Toward Programs With Common Sense, Communications of the ACM, 33(8), 30-49.

MACGREGOR, R. (1991). Inside the LOOM Classifier. SIGART Bulletin, 2(3):70-76, 1991.

MILLER, G.A. (1995) WordNet : A Lexical Database for English, Communications of the ACM, 38(11), 39-41.

MUGNIER, M.-L., AND CHEIN, M., eds. (1998) Conceptual Structures : Theory, Tools, and Applications (Lecture Notes in Artificial Intelligence 1453). Berlin: Springer-Verlag.

Efforts in the AI context

MYLOPOULOS, J., BORGIDA, A., JARKE, M., AND KOUBARAKIS, M. (1990) TELOS: Representing Knowledge About Information Systems, ACM Transactions on Information Systems, 8, 325-362.

NEBEL, B., AND VON LUCK, K. (1988) Hybrid Reasoning in BACK, in Methodologies for Intelligent Systems 3, Ras, Z.W., and Saitta, L., eds. Amsterdam: North-Holland

NEBEL, B. (1990) Reasoning and Revision in Hybrid Representation Systems (Lectures Notes in Artificial Intelligence, vol. 422). Berlin: Springer-Verlag.

PÉREZAND, A.G. AND BENJAMINS, V.R. (1999). Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods. In Proc. of the IJCAI'99 Workshop on Ontologies and Problem-Solving Methods, 1999.

Coupling DB and AI efforts

AL-ZOBAIDIE, A., AND GRIMSON, J.B. (1987) Expert Systems and Database Systems: How Can They Serve Each Other ?, Expert Systems, 4(1), 30-37.

BERTINO, E., CATANIA, B., AND ZARRI. G.P. (2001) Intelligent Database Systems, Addison Wesley, 2001

BORGIDA, A. (1995) Description Logics in Data Management, IEEE Transactions on, Knowledge and Data Engineering, 7, 671-682.

BRESCIANI, P. (1994) Uniformly Querying Knowledge Bases and Data Bases, in Working Notes of the KRDB'94 Workshop, Reasoning about Structured Objects : Knowledge Representation Meets Databases (Document D-94-11), Saarbrücken: Deutsches Forschungszentrum für Künstliche Intelligenz GmbH.

BRODIE, M.L. (1988) Future Intelligent Information Systems: AI and Database Technologies Working Together, in Readings in Artificial Intelligence and Databases, Brodie, M.L., ed. San Francisco Morgan Kaufmann.

Coupling DB and AI efforts

BRODIE, M.L. AND MYLOPOULOS, J., eds. (1986) On Knowledge Base Management Systems. New York: Springer-Verlag.

COHEN, B. (1989) Merging Expert Systems and Databases, AI Expert, 4(2), 22-31.

JARKE, M. AND VASSILIOU, Y. (1984) Coupling Expert Systems with Database Management Systems, in Artificial Intelligence Applications For Business, Reitman, W., ed. Norwood (NJ): Ablex.

KRDB WORKSHOPS. Proc. of the Workshop on Knowledge Representation Meets Databases.

EXPERT DATABASE SYSTEMS CONFERENCE. (1986,1987,1988,1989) Menlo Park: Benjamin/Cummings.

NORRIE, M.C., REIMER, U., LIPPUNER, P., RYS, M., AND SCHEK, H.-J. (1994) Frames, Objects and Relations: Three Semantic Levels for Knowledge Base Systems, in Working Notes of the KRDB'94 Workshop, Reasoning about Structured Objects : Knowledge Representation Meets Databases.

Coupling DB and AI efforts

RUPAREL, B. (1991) Designing and Implementing an Intelligent Database Application : A Case Study, Expert Systems with Applications, 3, 411-430

SELLIS, T., LIN, C.-C., AND RASCHID, L. (1993) Coupling Production Systems and Database Systems : A Homogeneous Approach, IEEE Transactions on Knowledge and Data Engineering 5, 240-256.

ZARRI, G.P. (1990) Expert Databases : The State of the Art, in Proceedings of the Second International Conference on Expert Systems Applications - EXPERSYS'90, Gournay-sur-Marne: IITT-International.

IDBs and Web

ARENS, Y., CHEE, C.Y., HSU, C., AND KNOBLOCK, C.A. (1993) Retrieving and Integrating Data from Multiple Information Sources. International Journal of Intelligent and Cooperative Information Systems. Vol. 2, No. 2. Pp. 127-158, 1993.

CERI, S., FRATERNALI, P, AND BONGIO, A. (2000) Web Modeling Language (WebML): a Modeling Language for Designing Web Sites. In Computer Networks, 33, 137-157.

FERNANDEZ, M.F., FLORESCU, D., LEVY, A., AND SUCIU, D. (1998) Web-Site Management: The Strudel Approach. Data Engineering Bulletin, 21(2): 14-20.

FRATERNALI, P. (1999) Tools and Approaches for Developing Data-Intensive Web Applications: a Survey. In ACM Computing Surveys, 31(3), 227-263.

IDBs and Web

GRAVANO, L. AND PAPAKONSTANTINOU, Y. (1998). Mediating and Metasearching on the Internet. In Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 1998.

GUHA, R.V., AND LENAT, D.B. (1994) Enabling Agents to Work Together, Communications of the ACM, 37(7), 127-142.

MARTIN, P., POWLEY, W., AND ZION, P. (1998) A Metadata Repository API, in Proceedings of the KRDB'98 Workshop, Innovative Application Programming and Query Interfaces (Swiss Life Technical Report).

MECCA, G., MERIALDO, P., ATZENI, P., and CRESCENZI, V. (1999) The (Short) Araneus Guide to Web-Site Development - Second Intern. Workshop on the Web and Databases (WebDB'99) .

WIEDERHOLD, G. (1994). Interoperation, Mediation, and Ontologies. In Proc. of the Int. Symp. on Fifth Generation Computer Systems (FGCS'94), Workshop on Heterogeneous Cooperative Knowledge-Bases, Vol. W3, pages 33-48, ICOT, Japan, 1994.

NKRL for semistructured data representation

BERTINO, E., CATANIA, B., LARADI, D., MARIN, B., AND ZARRI, G.P. (2000) Repository Management in an Intelligent Indexing Approach for Multimedia Digital Libraries, ISMIS'2000. Berlin: Springer-Verlag.

BERTINO, E., CATANIA, B., AND ZARRI, G.P. (2000) A Conceptual Annotation Approach to Indexing in a Web-Based Information System Economic Research and Electronic Networking Journal (NETNOMICS), 2(3):247-264, 2000.

ZARRI, G.P. (1997) NKRL, a Knowledge Representation Tool for Encoding the 'Meaning' of Complex Narrative Texts, Natural Language Engineering — Special Issue on Knowledge Representation for Natural Language Processing in Implemented Systems, 3, 231-253.

ZARRI, G.P: ET AL. (1999) CONCERTO, An Environment for the "Intelligent" Indexing, Querying and Retrieval of Digital Documents. In LNAI 1609: Foundations of Intelligent Systems, Proc. of the 11th Int. Symp. ISMIS'99, pages 226--234, Warsaw, Poland, June 1999.