SMCA04-11-0366-R2

Enterprise Collaboration: On-Demand Information Exchange Using Enterprise Databases, Wireless Sensor Networks, and RFID Systems

Cheng Hsu[1]*, **David Levermore**[1], **Christopher Carothers**[2], and **Gilbert Babin**[3]
[1]Decision Sciences and Engineering Systems, RPI, Troy, NY hsuc|leverd@rpi.edu
[2]Computer Sciences, RPI, Troy, NY chrisc@cs.rpi.edu
[3]HEC Montréal, Montreal, Quebec, Canada gilbert.babin@hec.ca

Original: December, 2004
Revised June, 2005
Revised November, 2005
Accepted March, 2006

To appear in *IEEE Transactions on Systems, Man, and Cybernetics, Part A*.

• Please send correspondences to Cheng Hsu.

Abstract

New extended enterprise models such as supply chain integration and demand chain management require a new method of on-demand information exchange that extends the traditional results of Global Database Query. The new requirements stem from, first, the fact that the information exchange involves large number of enterprise databases which belong to large number of independent organizations; and second, these databases are increasingly overlapping with real-time data sources such as wireless sensor networks and RFID (radio frequency identification) systems. One example is the industrial push to install RFID-augmented systems to integrate enterprise information along the life cycle of a product. The new effort demands openness and scalability, and leads to a new paradigm of ***collaboration*** using all these data sources. The collaboration requires a metadata technology (for reconciling different data semantics) that works on thin computing environments (e.g., emerging sensor nodes and RFID chips) as well as on traditional databases. It also needs a new extended Global Query model that supports participants to offer/publish information as they see fit, not just request/subscribe what they want. This paper develops new results toward meeting these requirements.

Keywords: Internet collaboration, information integration, market-style resource allocation, global query, Metadatabase, common schema

I. The On-Demand Information Exchange Problem

On-demand information exchange is a core element of Enterprise Collaboration; which underlies many emerging extended enterprise models in business, homeland security, and other domains. Prime examples in the business space include Industrial Exchanges (e.g., covisint.com), on-demand business (e.g., IBM solutions), and RFID-augmented information systems that drill through supply chains (e.g., Wal-Mart, FedEx, and Cisco). These trends are also echoed in military and scientific applications, which often involve wireless sensor networks.

Take supply chain integration as an example. A well-publicized model practice is the CFAR project by Wal-Mart and Warner-Lambert (reported in, e.g., ComputerWorld, September 23, 1996). The project connected some of Wal-Mart's demand forecasting databases with Warner-Lambert's production scheduling systems to shorten the replenishment cycle time for certain products. This practice remains a high watermark for enterprise collaboration at the process level of supply chain integration. However, it faces some basic limits when examined from the perspective of on-demand information exchange. First, the mechanism by which the systems exchange information was not readily expansible to include other likely participants in the supply chain. Second, the mechanism hard-coded the contents of the information transfer rather than providing them on-demand. In theory, both of these problems can be facilitated by global database query results such as federated databases; however, these results run into organizational hurdles because they require the participating companies to surrender much of the control of their databases. Then comes an even more daunting problem; that is, even if the prime (e.g., Wal-Mart) could whip its supply chain into order, the tiered suppliers on the chain still have to reconcile this particular Wal-Mart standard and protocols with their other primes or buyers. Suppliers need to project different "personalities" of their databases onto different "federations" of databases (assuming one federation for one supply chain) for integration, since each chain (prime) may require different information contents and data semantics.

This example shows that supply chain integration is enterprise ***collaboration*** in nature, and available database integration results (e.g., federated databases) do not sufficiently support this new regime. Furthermore, we submit that on-demand information exchange must ultimately consider sensors and RFID objects that feed real-time data to enterprise databases. This point is illustrated in the next example.

An ongoing study (2003-2006) at the State of New York under the auspices of the U.S. Federal Highway Administration includes a vision of turning the I-87 corridor (from the border of Montreal, Canada to New York City, New York) into a "Smart Highway" that supports regional economical growth as well as improves highway maintenance and transportation. Among the possibilities considered, massive sensor networks could be deployed along the highway and exchange data with the vehicles and freight cargos that carry next generation RFID (capable of light data processing). When the real-time feed of data from the sensors are also connected to enterprise databases at the freight companies, government agencies, and all other concerned parties, the Smart Highway would become an integral part of a supply chain management system for any cargo owners or users; or an extended Just-in-Time system for any trucking companies; or a control mechanism for any intelligent transportation systems…; or, simply, it would become an integrated information system for many possible extended enterprises.

Both examples show that enterprise collaboration requires **collaborative and extended global query** beyond the previously available models, which focus on pervasive and constant control (synchronization) of databases – i.e., databases respond to queries 24/7 (not on-demand) without own control (cannot pick-and-choose what queries to answer), and the production data models are made known to users. Previous results also do not consider light database processing on board of the chip at a sensor node or an RFID object. From the perspective of enterprise collaboration, sensor networks and RFID systems could interact with enterprise databases in both directions: they receive information in order to adapt their on-board instructions, and they send information to update the databases. Therefore, they could be full-fledged participants of the information exchange, even in the way of issuing database queries to enhance their on-board data-triggered controls and analytics; especially when the "system-on-chip" technology matures.

Therefore, the **problem** of on-demand information exchange has **three aspects**: determining-assigning the task-oriented collaborators; globally querying the heterogeneous data sources; and efficiently computing the information exchange tasks on both thin platforms (RFID chips and sensor nodes) and traditional databases. The first aspect of the problem is characteristic of collaboration. In traditional global query, there is always just one fixed set of participants (C(n,n)); while there are many possible sets (all combinations of n) in collaboration and each set could contribute different performance (perceived value and technical efficiency) to the overall community. This difference shows the need for a new regime to provide control to the participants, as well as to match them and execute the global query in an information exchange. A derived requirement of control is the participating enterprise databases' ability to publish differential "personalities" to fit for different collaborations. The global query aspect of the problem entails an open and scalable metadata technology to deal with heterogeneity. Previous results employ common schema, federated registry, or other methods to reconcile information semantics when participants use different data models. Industrial standards and ontology are crucial to these efforts. Unfortunately, standards tend to be proprietary and numerous, and domain-specific ontology is hard to develop and evolve. The third aspect, thin computing, is interwoven with the progress of chip technology. However, it is possible to consider mainly a range of generic designs with a range of possible functionality without predicating on any particular technology. The significance will be the model itself – i.e., the explicit incorporation of sensors and RFID in the information exchange to inter-operate with traditional databases.

This paper presents a **Two-Stage Collaboration Model -** matching collaborators and executing the global query - to resolve the above problem. The model uses a market-style structure to match collaborators on their "export" databases (personalities) for on-demand tasks - the first aspect. It uses a metadata-based matching method and employs a scalable model-based common schema to support global query - the second aspect. It also uses an extended enterprise metadata language to include certain classes of sensors and chip-based RFID in the global query - the third aspect.

In the next section, Section II, we review the previous results and discuss the basic concepts of the Two-Stage Collaboration model. The core methods of the model are presented in Section III. They consist of the metadata language and the query database schema for query formulation (the first stage), and the extended global query processing method (the second stage). Section IV elaborates the Common Schema and discusses the computing services for sensor networks. Section V describes the verification of the model through laboratory prototyping and previous proofs. Section VI concludes the paper and comments on future work.

II. The Concepts of the Two-Stage Collaboration Model

The first basic concept has to do with the matching of information requests with information offers for the collaboration. This is a task of allocating information resources to jobs; which has the analytic nature of real-time scheduling – an NP hard problem [15, 16, 23]. Research has shown that market-style self-scheduling is a promising approach to solving this problem when closed-form solutions are unattainable - see [3, 4, 11, 14, 22, 37, 38, 39, 44, 49]. Researchers have also applied this market approach to managing distributed (homogeneous) databases [43].

The market approach is often credited by these works to reduce the waiting time built in the command chain and increase the flexibility in setting the control criteria online to allow for dynamic utilization of resources. We would further argue that for on-demand information exchange a market mechanism promises to promote maximum participation of all possible information resources in the community for the common good. However, to make the market approach work for on-demand information exchange, the field needs additional results to respond to the unique challenges of the problem; such as the dual-nature of participants (an information provider can also be an information user in the same time), heterogeneous information models, and real-time online performance measurement and adjustment [28, 29]. Thus, existing e-business models (e.g., industrial exchanges [20] and application services providers [45]) have not yet provided global query capabilities. We embrace the market approach and fill in the gap.

Second, the field of Global Database Query has provided many metadata methods to help reconcile different data models; including global schema integration, federated registry, conceptual dictionary, and repository of enterprise metadata [7, 12, 13, 17, 30, 34, 41, 46]. All of these results are one kind or another of "loose" common schema to accommodate heterogeneous data semantics when no one authority can impose a single, comprehensive data standard on all participants (which would be a strict common schema) [18, 21, 43]. Not all of these results offer sufficient simplicity, openness, and scalability to work effectively in collaborative environments over the Internet. In fact, the experiences in the field (e.g., the E-Engineering effort at GE and the ebXML Registry Information Model at Oasis) suggest that an open and scalable common schema would not be feasible unless it is based on some sound ontology. Sound ontology; unfortunately, is also evasive. Common practices in the field tend to base ontology on the domain knowledge; that is, the ontology would enumerate all possible "concepts" of the application domain. The on-going ebXML project coordinated at the United Nations represents a state of the art [48]. In academia, a popular approach is to use linguistics, such as the Conceptual Models [42].

An alternative that has been shown to be feasible is to develop the ontology from the basic concepts and constructs of an information model that application systems can use to represent their information contents. In this approach, these generic elements will directly give rise to the basic structure of a repository of enterprise metadata, and the repository will become a common schema. The Information Resources Dictionary System by NIST represents such an approach, and the Metadatabase model for global query of multiple databases is another, proven example [2, 5, 6, 12, 24, 25, 26, 27]. We employ and further extend the Metadatabase Model to develop the open and scalable common schema required for on-demand information exchange.

Third, wireless sensor networks and RFID belong to the same class of mobile data sensing technology. Sensors perform both transponder and transceiver roles, while RFID at present serves as transponders. However, both technologies perform chip-based data processing. The chip technology gives them the ***potential*** to become ever more significant enterprise data sources. At present, the sensor nodes of wireless sensor networks can perform such computing as beam forming, aggregation of related data, and routing. The central nodes or gateways of these networks can provide enterprise level computing capacity to supplement and integrate the distributed processing at sensor nodes [1, 19, 30, 34, 35, 40, 50]. Generally, the sensor node computing must be energy-efficient due to limited power supply (batteries). However, computation is inexpensive relative to communication. Thus, it is advantageous to embed some database capability into on-board computing for sensor nodes to make them self-sufficient in control. The same can be argued for chip-based RFID, at least from a technology perspective; that is, RFID chips could function as mobile data processing and transmitting nodes.

Furthermore, we envision a two-way inter-operation between enterprise databases and sensors and RFID in the new model: direct feeding of real time data into enterprise databases, and adaptive control of sensor networks and RFID-carrying objects based on data from enterprise databases. With sufficient, efficient computing results [8, 9, 10, 32, 36, 47] to support the heightened on-board computing, we embrace these nodes as new enterprise data sources.

With the above concepts, we now define the **Two-Stage Collaboration Model (TSCM)**.

Definition of Collaboration:

Participant: a single or a cluster of data sources that controls its own participation in the Community and the information contents with which it participates; responsible for its own data models; and can independently issue information requests and/or information offers. (This definition is stronger than the usual notion of autonomy in the database literature.)

Community: a collection of participants which joins through a Community-sanctioned registration process and subscribes to a Community-sanctioned protocol of collaboration.

On-Demand: The self-controlled initiation of a request and/or an offer (the publication and subscription for information exchange) by a participant; which can start at any time, last for any duration, and change at any time.

Collaborators: the matched information requests and information offers (or the participants who initiated these matched requests/offers).

The basic logic of the TSCM:

The *objective function*: the maximization of collaboration.

The *decision variables:* information offers and information requests.

The *constraint*s: the transaction requirements and data semantics of tasks.

A *feasible solution*: a match of information offers with requests.

The *solution*: The execution of the matched offers and requests for the collaborators.

Stage 1: match requests with offers to find the right information collaborators; i.e., determine the optimal alignment of participants for a global query processing task.

Stage 2: execute the global query task; i.e., distribute the task to participants and process it at the local sites, and then assemble the query results for the user.

The above definition allows for peer-to-peer collaboration as well as using a global administrator. Nonetheless, a minimum (passive) global site that implements a registration process is required. The definition is consistent to that of the Enterprise Resources Model (ERM) that we presented in [29]. However, in this paper, the TSCM concept is further elaborated with new particular methods to implement the model for on-demand information exchange. Specifically, Stage Two uses exMQL, Query Database, and exMGQS (see below) to extend the previous Metadatabase Model, while the matching method for Stage One is completely new. These particular methods are amenable to coupling with industrial standards such as ebXML and XQuery.

This design leads to simplified computation. Unlike most other market models, which separate matching and global query as two un-related methods, the new TSCM design features a unified Metadata Query Language, called exMQL, to perform both matching and global query. Towards this end, the Metadatabase model provides the basic representation method for both the novel query database used in the matching and the community data sources in the global query, and thereby integrates the query language, query processing, and matching into a simplified regime.

Both information requests and information offers are formulated in exMQL as database queries and saved in a Query Database whose schema mimics the basic semantics of the exMQL. With this design, matching is but a regular database query processing against the query database. After the optimal participants are determined from the matching, the language also executes the requests at the local sites of the matched offers – i.e., global query processing. The query database schema and the metadata language both include rules as well as data, to support not only the data resources in an information offer/request but also their constraints. The inclusion of rules is an extension to the previous matching methods and previous query languages.

The design reduces the usually custom-developed Blackboard to a standard database management system (DBMS) augmented with certain SQL-based procedures. The contributions of the design also include the open and scalable common schema and the efficient computing onboard the sensors that the Metadatabase facilitates. We elaborate the design below.

III. The Core Methods of the Two-Stage Collaboration Model

The architecture of the Two-Stage Collaboration model is presented first; then, within this context the new matching method and the ensuing global query processing are discussed.

A. The Global Site and the Participant Site

Elements of the new matching method are shown in Figure 1, which depicts a global (or market) site with full functionality. A reduced set global site is possible if the role of the Metadatabase is reduced from one that provides global data semantics reconciliation to any lesser designs.

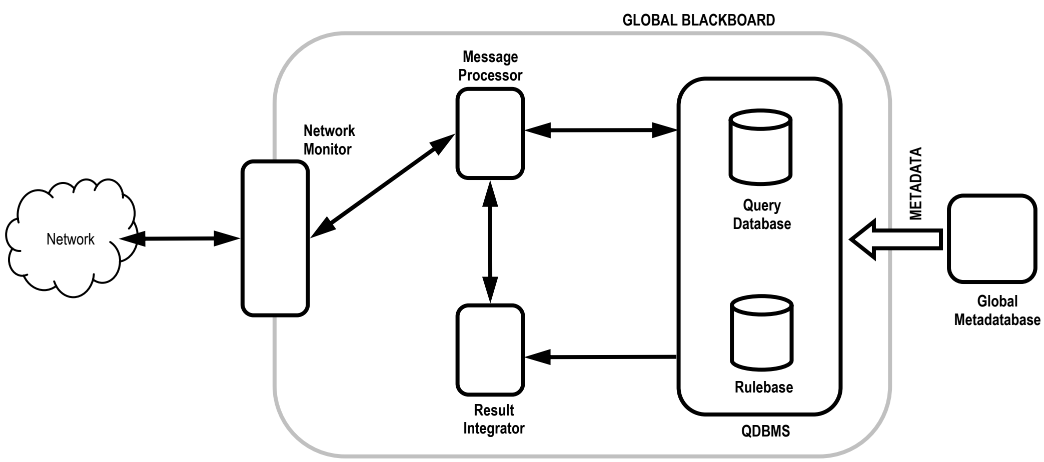


Figure 1: The Architecture of the (Full) Global/Market Site.

The Global Site has two major parts: the Blackboard and the Metadatabase. The blackboard performs the matching and global query functions while the Metadatabase is essentially the (open and scalable) schema of the collaborating community. As discussed in Section II, the Blackboard is a collection of processing methods augmented to a DBMS, where the exMQL is processed. The exMQL is the neutral, global query language the participants use to define their information requests and information offers as queries; and the results are saved into the Query Database against which the matching is performed. The Rulebase is the repository of the rules included in the exMQL expressions, and is also structured as a database. The schema of the Rulebase is a subset of that of the Query Database; both of which are managed under the same QDBMS (Query Database Management System) as a part of the Blackboard. The Message Processor, Result Integrator, and Network Monitor interoperate within the Global Site, which connects with the Local Sites for the execution of the Global Query.

The Metadatabase (a repository of metadata structured as a database) is created to include and integrate local data models ("export databases" and views) as metadata tuples in metadata tables. Metadata entries (data models) are added, deleted, updated, and retrieved as ordinary database operations against the Metadatabase without interrupting the continuous operation of the market. The Metadatabase is also implemented using a standard DBMS. A formal registration process with varying degrees of central coordination is required of all participants. In the maximum registration, the Metadatabase integrates all published local data models; while in a minimum regime, the Metadatabase contains only global equivalence information. The local sites are responsible for maintaining one or more export database(s) as the proxy database(s) for their enterprise databases in collaborations. These export databases could just be some database views in the minimum regime. In any case, the participants could opt to register a large, stable data model for the export database within which they publish their smaller, ad hoc offerings as often as they wish; or, they could frequently register different, small local data models that represent their published views (export databases) and change them on the fly. The most demanding part of the registration is the establishment of a list of global data items across local sites, and the maintenance of the data equivalence among them. We stress, however, that this is a focus of ongoing research in the field, not a problem that the new model uniquely faces.

However, we wish to point out that the new model actually facilitates these pervasive problems. The Metadatabase method affords a minimum common schema to minimize the demand of registration - a list that is automatically derived from all registrations, which can reveal any peer-to-peer correspondence of local data items. More significantly, ***registering a proxy-export database tends to be much easier and more practical to participants than registering an enterprise database***. That is, a company could subject an external-use-oriented proxy (see below) to certain community protocols where it could not a mission-critical production system.

The Local Sites are connected to the Global Site and to each other through a Proxy Database Server – see Figure 2. A local site is constructed and maintained according to a global design, but is otherwise administered by the participant. The Proxy Database Server in the full version replicates the global blackboard, so that it can administer any **peer-to-peer information exchange** when the Global Site is not involved. In this case, the Proxy Server has the ability to initiate a virtual circle of matching among peers and serve as the "global" site of the circle during the life of the match. In this way, the global blackboard co-exists with many concurrent, distributed local blackboards and their virtual sub-communities of information exchange. This design promises to reduce the computing bottlenecks in the community, enhance overall system reliability (against system or network failures), and support maximum flexibility of the collaboration regime. The Proxy Server serves as the surrogate of the Global Site, as the agent of the Participant, and as the firewall between the global and local sites.

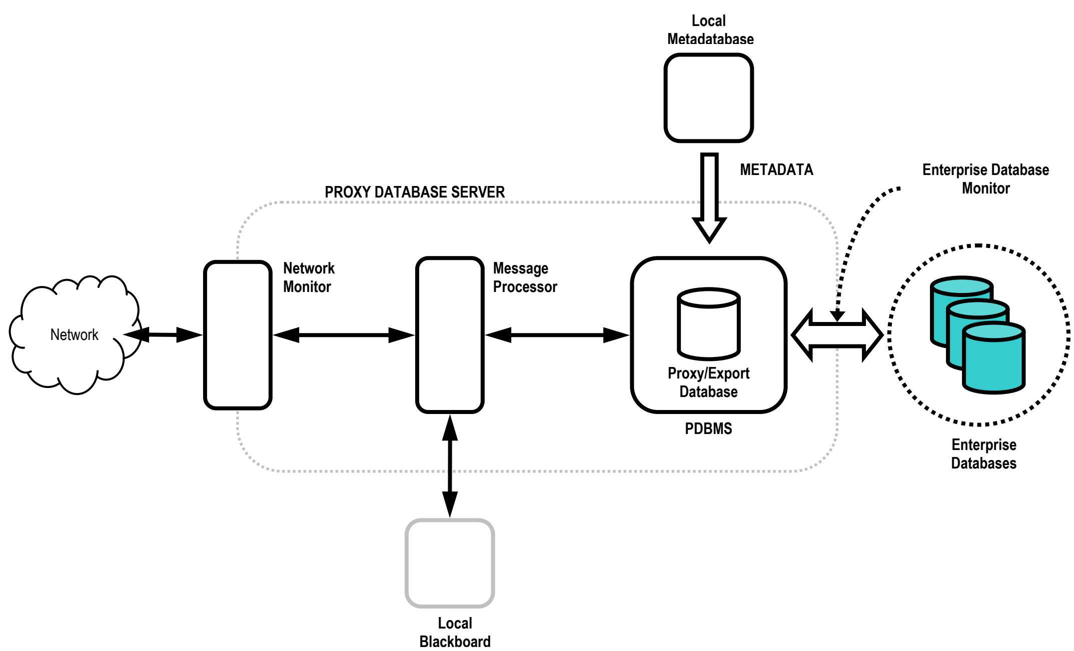


Figure 2: The Architecture of the (Full) Local/Participant Site.

The Proxy Server may reduce its requirement on a full Metadatabase when peer-to-peer metadata interfacing or interchanging is included; and hence becomes a reduced version. Similar to the minimum regime of registration, a minimum set of metadata at proxy servers includes only the list of global data equivalents. A partial Metadatabase ranges between the minimum and the maximum contents. The cost for maintaining the distributed copies of global equivalents represents an upper bound to the real-time performance of peer-to-peer collaboration.

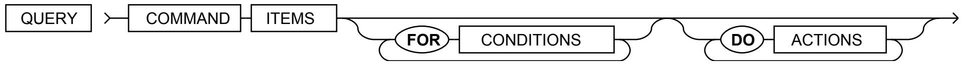
Participants formulate their information requests and information offers through the Proxy Server, which provides user-friendly interface to the Query Language exMQL. The Export (Proxy) Database is the image of the local enterprise databases that the participant publishes to the collaborating community. The Proxy DBMS manages the image as a real database for the global query processing. The extent to which the backend enterprise databases are taking part, directly, in the global query can be controlled through the controls of the PDBMS. We wish to note that the export database could simplify the complexity of dealing with intricate enterprise systems since multiple enterprise databases can have their views consolidated at the export database. Conversely, a participant can maintain multiple export databases for the same production systems, to subscribe to multiple standards and protocols imposed by, for instance, multiple supply chains. Both make the collaborative model more practical than traditional Global Query designs. The Message Processor and Network Monitor are the modules that pertain to the global query processing and inter-operate with their counter-parts at the Global Site. Together, these components at the Local Sites and the Global Site constitute a distributed processing environment called ROPE in the literature [2, 12, 24].

B. The Metadata Query Language: exMQL

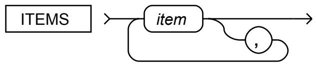
The Metadata Query Language is extended from the previous MQL results in [12] to include rules in a uniform query format for both information publication and subscription in the Two-Stage Collaboration. The structure and basic semantics are based on the TSER representation method [26] and the GIRD metadata model [25]. The full specification of exMQL is provided in Figures 3 using an alternative BNF (Backus-Naur Form) syntax. Each query operation is performed via the extended Metadatabase Global Query System (exMGQS) which is the combination of the Global Blackboard and the Proxy Database Servers at local sites.

The exMQL automatically retrieves metadata from the Metadatabase to facilitate query formulation, and hence provides certain interesting design features. It supports joins of data from different participants that have different definitions (i.e., differing names, formats, units, and/or value coding). such as relief the users the burden to know such technical details as the physical locations, local names, and implicit join conditions. Users can use familiar names in queries to qualify data elements in different databases, since the Metadatabase utilizes an internal identifier for data item resolution, while associates these with “friendly” user-defined names.

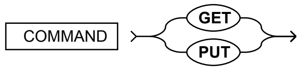
<QUERY> ::= <COMMAND> <ITEMS> *['FOR' <CONDITIONS>]* *['DO' <ACTIONS>]* ;



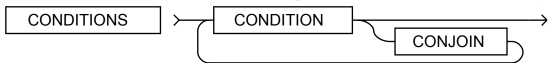
<ITEMS> ::= /[item || ',']/ ;



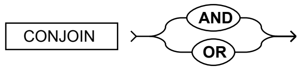
<COMMAND> ::= 'GET' | 'PUT' ;



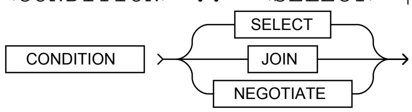
<CONDITIONS> ::= /[<CONDITION> || <CONJOIN>]/ ;



<CONJOIN> ::= 'AND' | 'OR' ;



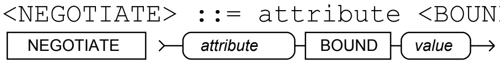
<CONDITION> ::= <SELECT> | <JOIN> | <NEGOTIATE> ;



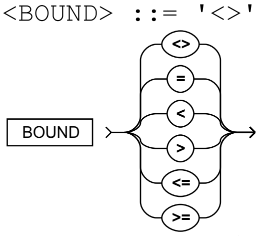
<SELECTION> ::= item <BOUND> value ;

<JOIN> ::= item <BOUND> item ;

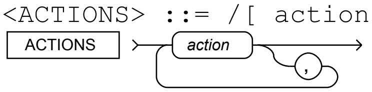
<NEGOTIATE> ::= attribute <BOUND> value ;



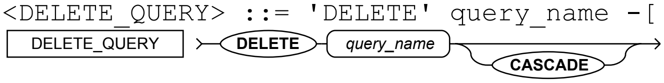
<BOUND> ::= '<>' | '=' | '<' | '>' | '<=' | '>=' ;



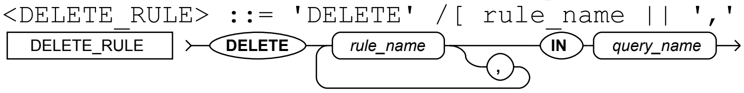
<ACTIONS> ::= /[action || ',']/ ;



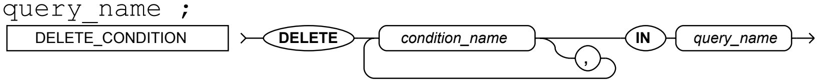
<DELETE_QUERY> ::= 'DELETE' query_name -['CASCADE']- ;



<DELETE_RULE> ::= 'DELETE' /[rule_name || ',']/ 'IN' query_name ;



<DELETE_CONDITION> ::= 'DELETE' /[condition_name || ',']/ 'IN'
query_name ;



<UPDATE_QUERY> ::= 'UPDATE' <ITEMS> *['FOR' <CONDITIONS>]* *['DO'
<ACTIONS>]* 'IN' query_name ;

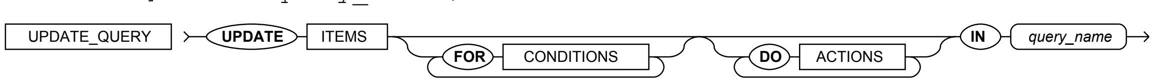


Figure 3: Extended Metadatabase Query Language Syntax Diagrams

In Figure 3 the GET and PUT commands specify a subscription query (information request) and publication query (information offer), respectively. Both commands are followed by a comma-delimited list of data items for retrieval (subscription) or sharing (publication), as represented by the ITEMS category. At least one data item must be provided in a query, which must also contain a GET or PUT command. These are the minimum requirements for a global query in the TSCM.

The FOR command specifies the retrieval conditions. Three classes of retrieval conditions are considered: selection conditions (SC), join conditions (JC) and negotiation conditions (NC). These conditions serve two functions: (1) to be used in the evaluation of a match, and (2) used in a manner analogous to the WHERE command in traditional SQL. Multiple conditions are conjoined by the logical operators, AND and OR. A selection condition is defined as a data item bound to a literal value, i.e. a string or numeric value. A join condition is a data item bound to another data item, and a negotiation condition is a system-supported negotiation attribute bound to a literal value. The specification of conditions in a query is optional. The DO command is used to specify the actions of a query. An action can be associated with a particular condition, and accordingly will be executed if the condition evaluates to true. Also, an action can be associated with a query in general, and so will be executed on the successful match of a query. The specification of actions in a query is optional.

The exMQL provides functions that change queries. The DELETE command removes a query from the Blackboard. If the optional command CASCADE is specified, then all queries related to the particular proxy database server will be removed. If it is necessary to delete a rule of a query, the DELETE RULE command is used. This command deletes all conditions associated with a rule. More than one rule can be specified for deletion in each delete rule command. The DELETE CONDITION command removes a condition from an existing rule, and multiple conditions can be specified on a comma-delimited list. A query can be completely revised by using the UPDATE QUERY command. The new lists of data items and conditions in the update query command will replace the existing ones. The changes will cascade to the rulebase.

C. The Conceptual Schema of the Blackboard

The conceptual structure of the Query Database and the Rule-Base – see Figure 4 - are based on the GIRD representation [25]. The interesting point is the recognition in this research that ***this representation unifies the formulation of information requests and information offers, the matching for collaborators, and the processing of the global query for the collaboration***.

The Logical Model of the Query Database and the Rulebase

The query database and the rulebase have a unified logical representation in a single model, as illustrated in Figure 4. The SYSTEM, QUERY, and VIEW meta-entities in Figure 4 replace the APPLICATION, SUBJECT and ENTREL meta-entities in the GIRD, respectively. The remaining meta-entities, and meta-relationships that connect them, retain their original definitions as outlined in [6, 25]. The salient changes are described below:

- The SYSTEM meta-entity identifies the participants/export databases that are currently participating in active queries, and so represents a dynamic view of the TSCM. Each proxy database server is defined by a unique identifier, which is determined at design-time when the local data model is integrated into the Metadatabase; however this is not made available to the global Blackboard unless a query has been submitted by the proxy database server.
- The QUERY meta-entity identifies the queries submitted by proxy database servers. Each query submitted to the Blackboard is associated with a unique identifier that is assigned at run-time. A timestamp attribute is generated automatically when the query is received at the Blackboard. The timestamp is primarily used for breaking ties, but is additionally used to remove queries from the query database after a user-defined or system-enforced expiry data has passed. The related **components** meta-MR associates queries with a particular proxy database server (SYSTEM), and upholds existence and dependency constraints by deleting these queries if the proxy database server is removed entirely from the TSCM.
- The VIEW meta-entity allows a single query to provide multiple interfaces, analogous to a database table that has multiple views.
- The ITEM meta-entity represents the data items specified in each query. The related **belongto** meta-PR associates data items to a specific VIEW, and the **describes** meta-PR specifies the data items that belong to each QUERY.

The rulebase inherits the rule modeling paradigm of the GIRD and is derived from the core concepts and implementation provided in [6]. The RULE, CONDITION, ACTION, and FACT meta-entities, and their associated meta-PR's, meta-FR's and meta-MR's retain their original functionality; however they now must provide for the management of data objects as opposed to metadata objects. The general syntax of a rule is based on the Event-Condition-Action (ECA) grammar described in [2]. In the TSCM however, actions do not have to be associated with a condition, although the action is still associated with the event. In this case the conditions implicitly evaluate to true when the event is triggered.

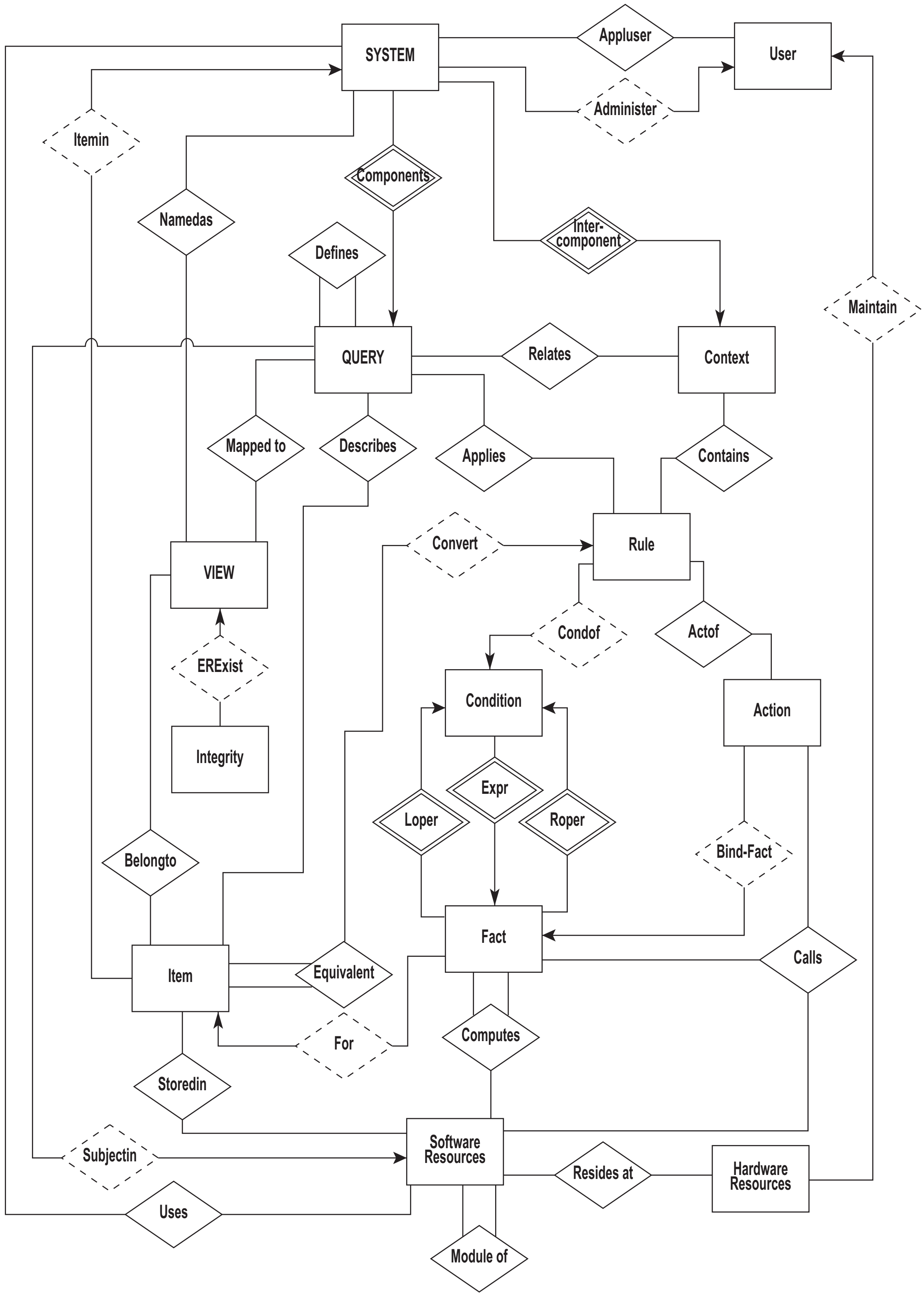


Figure 4: Conceptual Structure of the Blackboard.

D. The Extended Global Query System: solution for the second stage of the collaboration

The extended Global Query System (exMGQS) is assisted not only by the Metadatabase [2, 12], but also by the new Query Database. We depict the basic control and information flows of exMGQS in Figure 5, using the concept of the ROPE shell [2] to simplify the representation. The ROPE shell (the double circle icon in the figure) signifies the inter-operating elements of the Blackboard and the Proxy Database Server. At the second stage of the TSCM, the matched queries are converted into local-bound sub-queries for the set of participants determined at the first stage, by the Global Blackboard using the Metadatabase. The sub-queries are then transmitted to participants. The sub-queries are processed by the ROPE shells at the Proxy Database Servers and the results assembled at the Global Blackboard. This procedure is similar to the one described in [12]. For peer-to-peer collaboration, the Local Blackboard at the Proxy Database Server will play the role of the Global Blackboard. We assume a protocol of distributed copies of the full Metadatabase for the community. Since the Metadatabase contains only data models with fixed meta-tables, its size (number of rows) is a linear function of the number of local sites. It is also possible to include different processing regimes for different classes of participants, such as using the Decoupled Query Methods to optimize the processing for all sites that have homogeneous data semantics, including the sensors.

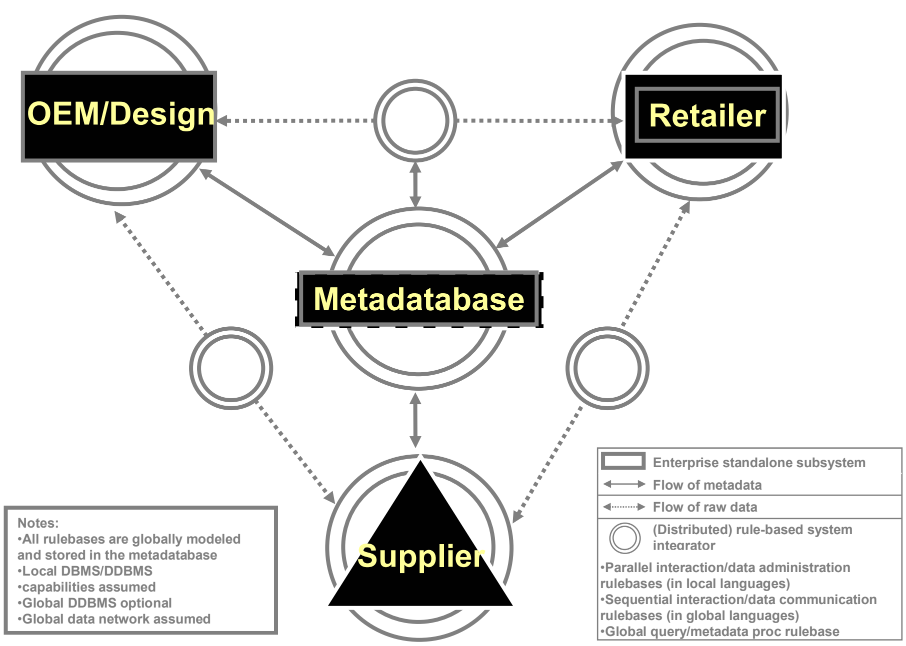


Figure 5: The Global Query Information Flow.

IV. Additional Methods: Common Schema, Efficient Computing, and Light Data Sources

The above core methods integrate the new Matching model with the previous Metadatabase Model. We now address three feasibility topics: common schema, efficient computing, and light databases on sensor nodes and RFID chips.

A. Common Schema: Openness, Scalability, and Ontology

Common Schema is a daunting requirement for all distributed systems. Its intellectual nature is simply the generalization of the well-know three-schema database model from dealing with a single database to a large collection of databases; however, the task has been proven to be anything but simple. As discussed in Section 1, a common schema needs some ontology to provide it with structural stability and a definitive scope of contents. The Metadatabase uses an ontology that comes directly from a class of generic information modeling concepts that the field uses to design database. Therefore, the ontology does not presume a comprehensive linguistics at one hand, nor a universal set of meta-knowledge about all applications at the other; both of which are hard to acquire and hard to maintain. The field offers a number of such modeling concepts, including Entity-Relationship-Attribute and Object-Classification. To the extent that these concepts are applicable to all participants and that a general methodology exists to reverse-engineer the participants' proprietary data models into these concepts, a set of common TYPES of metadata is readily at hand. This set is open and scalable as long as these conditions hold, and it is as simple as the concepts themselves. This is the approach the Metadatabase follows.

The conceptual schema of the Metadatabase is shown in Figure 6. It is based on the ontology of an extended entity-relationship-attribute model called TSER, which encompasses the ERA and the O-O methods with an attendant translation methodology [26]. Each icon in the figure represents either a table of metadata or a particular type of integrity control rule. All metadata are categorized into four inter-related groups: User-Application, Database Model, Contextual Knowledge, and Software and Hardware Resources. The Database Model metadata types include Subject (comparable to Object and View), Entity-Relationship, and Data Item; while Contextual Knowledge is structured in terms of a Rulebase Model (the representation of Rules). User-Application is defined to support multiple enterprises, application families, and user interface (including natural language input [5]). The Software and Hardware Resources represents, originally, the networking middleware and the local databases involved in Global Query. The Equivalent meta-relationship cross-references data items from one model to their equivalents in others. Together with the conversion routines represented in Software and Hardware Resources, they achieve data semantics reconciliation for the community. This conceptual schema defines meta-tables (of metadata) that constitute the Metadatabase; which is implemented as a standard database and serves as the common schema of the community.

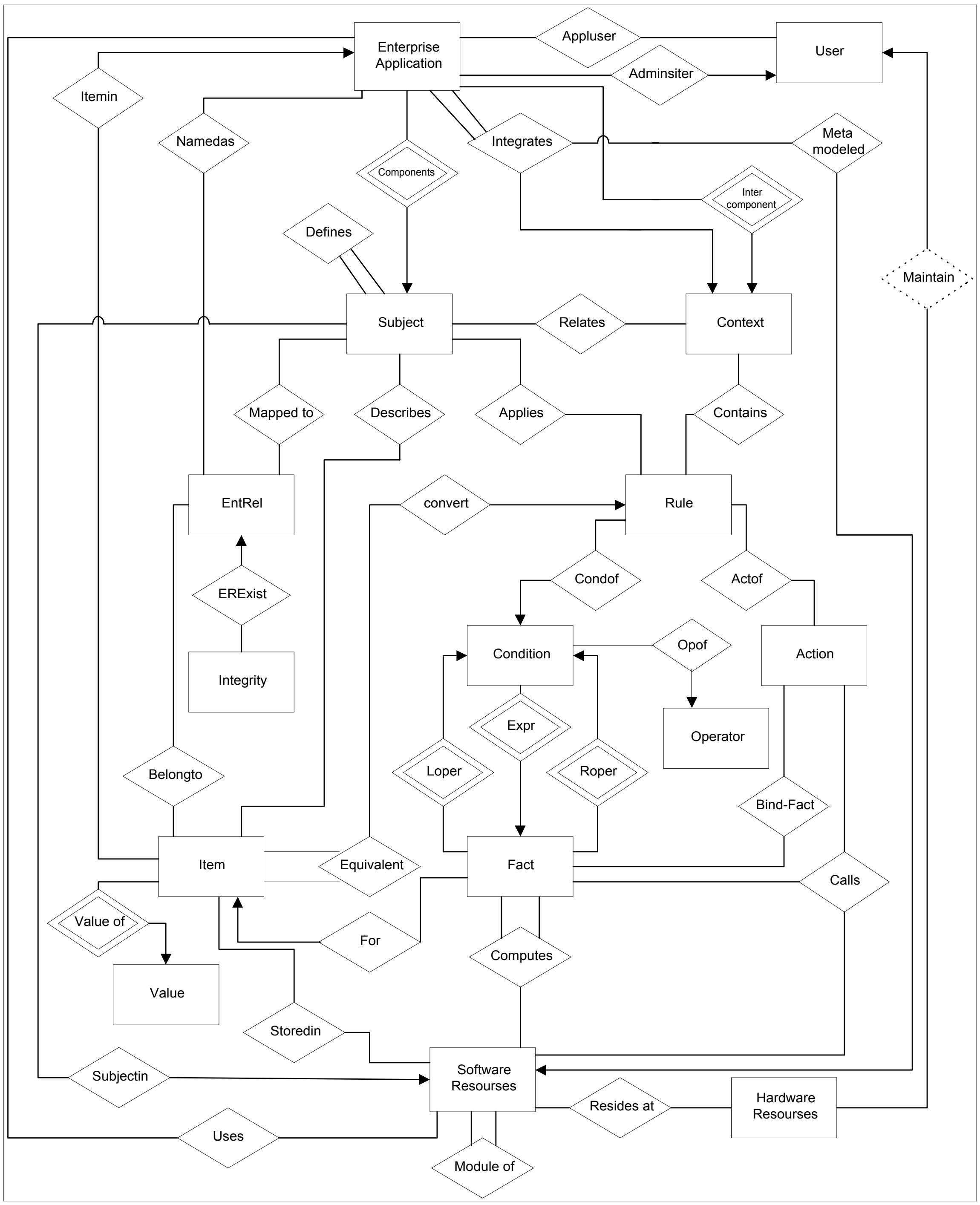


Figure 6: The Structure of the Common Schema - the Metadatabase.

A participant information model would be represented (reverse-engineered) in terms of these concepts and saved in the Metadatabase as entries to the meta-tables. Thus, a local model would have a constant number of entries in the Metadatabase, and the average of these numbers for all models is a constant. When a new participant is added to the community, a new Proxy Database Server will be fine-tuned for the participant to create a new Local Site. The Local Site will register itself and create a TSER representation of its Export Database(s) as new metadata entries to the Metadatabase. This process is amenable to using a CASE tool. The Metadatabase does not need to shut down at any time during the update, since only ordinary database operations are involved; and hence the whole community will not be disrupted, either. Ongoing update to existing local models uses a similar process. In this sense, the Common Schema is open and scalable, as required by the on-demand information exchange for enterprises collaboration.

A minimum Metadatabase design is attainable from Figure 6, to facilitate Efficient Computing on sensors and RFID chips, as well as to accommodate the peer-to-peer possibilities discussed in Section II. That is, depending on the specific technology used for the sensor nodes, the transceivers, and the transponders, the global Equivalent meta-table can be used as the core of a minimum Metadatabase; which, in turn, can become the Local Metadatabase on the Local Site. It is also possible to decompose the Software and Hardware Resources meta-tables of Figure 6 and include them in the Local Metadatabase to support the onboard analytics and light databases on sensors and RFID chips. The actual design has to be based on the implementation technology and be fine-tuned to save computing energy.

B. Efficient Computing for Sensor Networks and RFID Chips

The Full Local Site design is appropriate for the central nodes of sensor networks and the master nodes of transceivers of the RFID systems, as well. However, we need new designs to accommodate the particular requirements of sensor nodes and RFID chips if they are to become autonomous enterprise data resources. The above sections discussed the reduced designs for the Local Metadatabase. We now discuss the Efficient Computing needed for the Local Blackboard.

We first review the requirements. In the new vision, sensor nodes and RFID chips are participants of information integration for the enterprise. So, they need to be searchable by global queries initiated at other sites; and they need to be the subjects issuing global queries to obtain necessary data from other sites. The second capacity allows sensor networks and RFID chips to update their control regimes and analytics based on conditions published elsewhere, using, e.g., data-triggered control and decision rules. These rules could initiate global queries (publish or subscribe) based on data sensed or received and the time elapsed. The Blackboard, therefore, needs to process at least simple rules and possess minimum ROPE functionalities. Not all sensor nodes and RFID chips in the literature possess such computing capacity; but some do. More importantly, the future sensors and RFID will possess these capabilities as the chip technology advances. Therefore, we poise the research for the collaboration in the next phase of progress.

Given the physical characteristics of sensor networks and RFID chips, we submit that the Erlang (www.erlang.org) technology affords the collaboration model a tool to satisfy these requirements. Erlang is the language used to implement the routing software in one of Ericsson's high-speed ATM switches. Erlang is a functional programming language where data are passed as arguments to functions and all variables are single assignment – i.e., a variable can only be written once. Thus, it provides a number of features that increase system availability and robustness, such as ultra-light weight threads for concurrency, error detection methods for recovery and robustness, software real-time scheduling, predictable incremental garbage collection, "hot swappable" and "incrementally loaded" code modules, highly efficient code representation, and external interfaces to other third-party software. It also embeds a database management system capable of providing an overall set of services to support (a reduced version of) the Metadatabase and the Blackboard. We envision in the future research the development of a set of sensor computing services that go beyond the current sensor operating systems (e.g., TinyOS), using the above functions as the conceptual roadmap and Erlang as the implementation tool.

C. The Local Site Strategies for Light Databases on Sensor Nodes and RFID Chips

We now propose some strategies for implementing the TSCM architecture on sensor networks and RFID systems. Since we have to speculate on the emerging technologies, we shall restrict the discussion to a general and conceptual level.

Strategy I (Conservative): We assume that the central nodes and gateways of sensor networks and the transceivers of the RFID systems possess PC-class computing capability or more, and can be connected to the networks on which regular enterprise databases reside. In this case, the implementation strategy considers each (complete) sensor network and each (complete) RFID system as a participant (enterprise data source), and the Local Site architecture will be implemented at the central sites of these networks and systems. In fact, most sensor networks and RFID systems already manage their data resources as databases; it is just that they need a new model such as the TSCM to treat them as enterprise-level data sources and integrate them with the traditional enterprise databases. Strategy I can be implemented today to facilitate the feeding of real-time enterprise data into enterprise databases from these real-time data sources.

Strategy II (Moderate): We assume that sensor nodes and transceivers are widely distributed, and each could possess light databases such as the TinyDB technology for sensor nodes and the transceiver nodes at the Toll Booths of the EZ-PASS technology. The implementation strategy is two-fold in this case. First, we still implement the Local Site at the central sites and consider each complete network or system as a participant; but second, we also represent the data models of the sensor nodes and the distributed transceivers into the Local Metadatabase. That is, these data models, which would most likely be homogeneous within each network or system, along with their on-board instructions that use these local data will be represented as metadata and processed as such (see the above section). This way, the participant can include the distributed light databases in its queries, and to possibly update their data-triggered rules and instructions there.

Strategy III (Aggressive): We assume the existence of “system-on-chip” capability for sensor nodes, distributed transceivers, and RFID chips. We further assume that their system design will embrace a PC-class DBMS component on the chip, along with its communication capabilities. Finally, we assume that industrial standard exists to allow the RFID chips function as mobile sensor nodes, and the sensor nodes transceivers. In this case, we will have the flexibility to consider each sensor node, each distributed transceiver, and even each RFID chip a participant and apply the Local Site architecture to it. A combination of this strategy and the above two will also become possible, especially when the number and heterogeneity of the myriad sensor networks and RFID systems involved increases. This combination helps assure scalability, too.

V. The Laboratory Verification and Analysis

The core methods in Section III employ the previously established results of the Metadatabase Model (in particular the common schema and the global query processing using ROPE/MGQS) to complement the new Matching Model of the first stage. Therefore, the verification of the TSCM results is focused on establishing the feasibility of the new Matching Model, the design of the Global Site and the Local Site, and the Efficient Computing required for sensor nodes and RFID chips. It might be relevant to add that the MGQS results that the TSCM adopts have been tested satisfactorily for industrial scenarios provided by Digital (now a part of Oracle), General Motors, General Electrical, IBM, and Samsung Electronics during 1990's (see [24]) at Rensselaer and some company sites at Warren, Michigan; Kingston, New York; and Suwan City, Korea.

A. The Matching Model and the TSCM Architecture

A laboratory prototype of the Global Blackboard has been created in a laboratory environment to verify the Global Site and the Local Site design, to prove that the Match Model works, and to show that the exMQL and the Blackboard can all be implemented in open source technology and/or standard commercial products - see [33] for details. The prototype utilized the Fedora Linux Core 2 operating system running on a dual processor Dell workstation with Pentium 3 CPU, 900 MHz, and PostgreSQL Version 7.4.7, Apache and PHP. The Blackboard used PL/pgSQL and other programming facility provided to implement the procedures. Two database schemas were created for the Metadatabase and the Query Database (including the Rulebase) to contain both within one database and take advantage of their logical consistence. The prototype also included a Web-based user interface for the exMQL, implemented in PHP and Apache, to assist the users formulate publication and subscription queries.

According to a comprehensive analysis provided in [33], which is beyond the space of this paper, the basic matching algorithm at the first stage has linear complexity, *O*(*n*), in the size of the query database. The query cost as measured in page-reading and other operations were also shown to be consistent with the theoretical behavior illustrated in Figure 7 below:

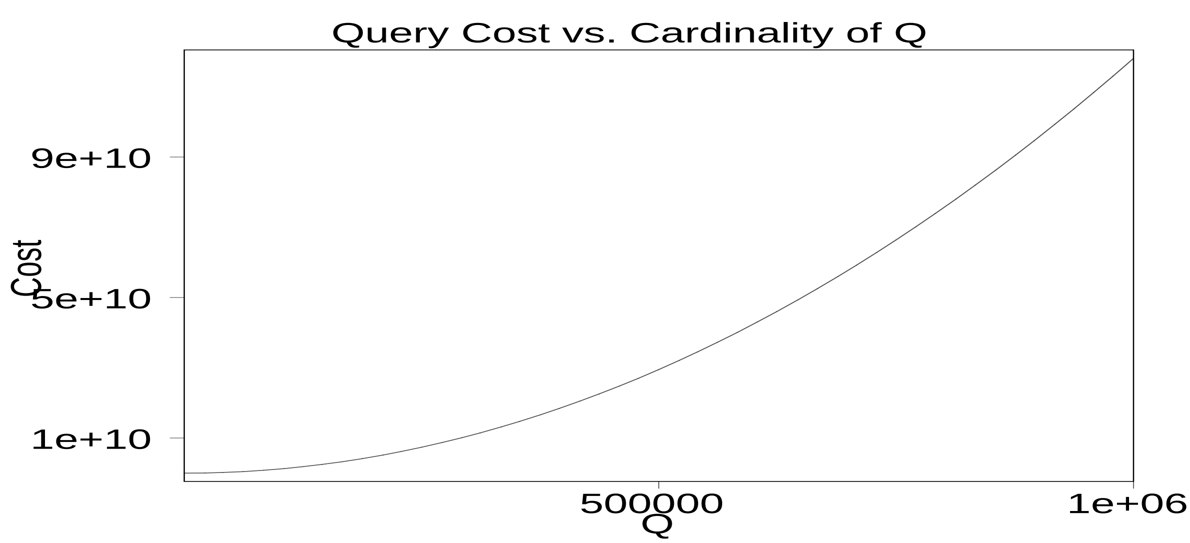


Figure 7: Blackboard Cost (CPU time) versus the Size of the Query Database.

These results suggest that the Match Model at the first stage is scalable in theory. Next, the TSCM architecture was verified in the laboratory environment to work as a complete solution.

The Global Site architecture was prototyped for a supply chain, and the Local Site was implemented for two enterprises - a retailer (the prime) and a supplier. The supply chain scenario was not limited to just one retailer and one supplier, but the verification of the ***architecture*** required only two participants to give a conceptually complete collaboration at the Global Site. The collaboration scenario showed that the Local Site design has worked to allow the retailer publishing select forecasting and inventory data from its enterprise databases for its suppliers to use as demand data needed for the latter's production scheduling. It also showed that the supplier, in a similar way, published its production data and delivery data that it wished to share with its buyers (the retailers) for use in the latter's replenishment processes.

Finally, the laboratory prototyping proved the implementation feasibility of the TSCM architecture. That is, the exMQL, the TSCM architecture, and the other elements of the Global Blackboard and the Proxy Database Server have all been implemented by using only the open source technology and standard off-the-shelf products. The Query Database and the matching required only ordinary database processing. The interaction of the Query Database, the Rulebase, and the ROPE elements at the Global Site required only ordinary database application processing. This implementation has worked correctly as designed [33]. Therefore, the claim that the Two-Stage Collaboration model can be implemented on an ordinary DBMS was verified. This claim indicates that the Local Site can be implemented for virtually any enterprise, any central nodes of sensor network, and any central node transceivers of RFID systems.

B. Efficient Computing

We also tested the limit of concurrent matching when massive participants join simultaneously, and reported the results in [29]. To go beyond the known upper bound that DBMS technology provides, the test used a specially created match engine to perform on a reduced set Query Database (with reduced semantics). The Erlang software technology was chosen to be the implementation tool, for the same reasons as we discussed above in Section IV.B. This test documented that the special match engine would scale up to10,000 concurrent matching jobs under concurrency control, and up to a million without it. We also discovered that using an external DBMS such as Oracle could be able to scale up to 100,000 concurrent jobs per machine.

The previous test revealed a few interesting observations. First, it confirmed the feasibility of designing Efficient Computing for Local Sites implemented on sensor nodes and RFID chips. Since they typically do not require concurrency control, the more relaxed upper bound would apply and hence allow for their massively distributed processing. Second, for the Global Site, we expect the upper bound of the concurrent matching jobs of the new Blackboard to be in the range of the DBMS-assisted special match engine, since the added overhead due to fuller semantics tends to be offset by the more efficient processing of DBMS query operations.

VI. An Assessment and Future Research

The Two-Stage Collaboration Model extends the previous Global Database Query model in two major ways: It allows the participants to offer or request data resources on demand and at will; and it supports not only enterprise database participants, but also wireless sensor networks and RFID systems. The new model integrates the previous Metadatabase Model (as the second stage) with the new Matching model (as the first stage), and thereby affords a solution to the on-demand information exchange problem. The model itself is an original contribution – it allows a database to belong to multiple "federations" of databases. The newly developed elements are original contributions, too: the exMQL contributes rules capability to matching, the Query Database contributes simplified and unified matching and querying, and the exMGQS contributes a feasible, model-based common schema to collaboration. The efficient computing supports linking sensors and RFID to databases. These results are developed in Sections II, III, and IV. The literature analysis established the claims of newness, while the laboratory prototyping substantiated the claims of feasibility (Section V).

The new model responds specifically to the supply chain integration problem and the Smart Highway vision discussed in Section I. It is now possible for a prime to employ global database query across the entire supply chain, and even for a large number of primes and suppliers to jointly form a community of on-demand information exchange. The suppliers could publish different (simultaneous) export databases from their enterprise databases (which are hidden from the community) to satisfy different protocols, standards, and requirements. They maintain full control of their databases but allow the community full access to the export databases and their data models. The global query protocols, including exMQL, exMGQS, and the TSCM architecture can be implemented in open source technology and/or standard commercial technology. The feasibility of registration and other related issues could be facilitated by the Export Database concept and by the minimum Metadatabase possibilities.

It is also now possible for an (extended) enterprise to use sensor networks and RFID systems as its real-time data sources to feed real-time data to its enterprise databases. The Efficient Computing design facilitates the implementation of the Local Site on these thin computing nodes. Possibilities of connecting the RFID systems to sensor networks are also opened by the model. These possibilities may contribute to the emerging push for RFID-based integration of customer and product information across supply chains and demand chains (e.g., FedEX, GE, IBM, and Wal-Mart). They could also be relevant to other applications in the public sector and healthcare.

Two main research problems remain. The first is the registration and the metadatabase distribution issues discussed in the paper (Sections III and IV). The second is the Efficient Computing designs for the sensor networks and RFID chips, since the results have to be coupled with the continuing progress of the technologies. Our vision here is to design an Erlang-based Operating System to enable sensor nodes and RFID chips performing the functionality of the Local Site. In addition to these two problems, we need to connect the TSCM results with industrial standards. For instance, the exMQL expressions may be carried by an XML wrapper and the model-based common schema may complement the ongoing Oasis/UN effort.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless Sensor Networks: a Survey,” *Computer Networks,* vol. 38, 2002, pp. 393-422.

[2] G. Babin and C. Hsu, "Decomposition of Knowledge for Concurrent Processing," *IEEE Transactions on Knowledge and Data Engineering,* vol. 8, no. 5, 1996, pp 758-772.

[3] A. Baker, “Metaphor or Reality: a Case Study Where Agents Bid with Actual Costs to Schedule a Factory,” in Market-Based Control: a Paradigm for Distributed Resources Allocations, ed. S. Clearwater, World Scientific Publishing, River Edge NJ, 1996.

[4] A. Baker, “A Survey of Factory Control Algorithms That Can Be Implemented in a Multi-Agent Hierarchy: Dispatching, Scheduling, and Pull,” *Journal of Manufacturing Systems*, Vol. 17, No. 4, 1998, pp. 297-320.

[5] V. Boonjing and C. Hsu, “Natural Language Interaction Using a Scalable Reference Dictionary,” *Proceedings of the 8th International Conference on Applications of Natural Language to Information Systems,* Burg, Germany, June 2003.

[6] M. Bouziane and C. Hsu, "A Rulebase Management System Using Conceptual Modeling," *J. Artificial Intelligence Tools*, Vol. 6, No.1, March 1997, pp. 37-61.

[7] R. Braumandl, M. Keidl, A. Kemper, D. Kossmann, A. Kreutz, S. Seltzsam, and K. Stocker, “ObjectGlobe: Ubiquitous Query Processing on the Internet,” The VLDB Journal, Vol. 10, 2001, pp.48-71.

[8] C. D. Carothers, K. Perumalla and R. M. Fujimoto. “Efficient Parallel Simulation Using Reverse Computation,” *ACM Transactions on Modeling and Computer Simulation*, Vol. 9, no. 3, 1999, pp. 224--253.

[9] C. D. Carothers, and R. M. Fujimoto. “Efficient execution of Time Warp Programs on Heterogeneous, NOW Platforms,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, no.3, 2000, pp. 299--317.

[10] C. D Carothers, D. Bauer and S. Pearce. “ROSS: A High-Performance, Low Memory, Modular Time Warp System, *Journal of Parallel and Distributed Systems,* 2002.

[11] A. Cesta, A. Oddi and S.F. Smith, "Iterative Flattening: A Scalable Method for Solving Multi-Capacity Scheduling Problems", *Proceedings National Conference on Artificial Intelligence (AAAI-00)*, Austin, TX, July, 2000.

[12] W. Cheung and C. Hsu, "The Model-Assisted Global Query System for Multiple databases in Distributed Enterprises," *ACM Transactions on Information Systems,* vol. 14, no. 4, 1996, pp 421-470.

[13] C.W. Chung, “DATAPLEX: an Access to Heterogeneous Distributed Databases,” *Communications of the ACM*, vol. 33, no. 1, 1990, pp. 70-80.

[14] S. H. Clearwater (ed.), *Market-Based Control : A Paradigm for Distributed Resource Allocation*, World Scientific Publishing, River Edge, N.J., 1996.

[15] R. Conway, W. Maxwell, and W. Miller, Theory of Scheduling, Addison-Wesley, Reading, MA, 1967.

[16] D. Coppersmith and P. Raghavan, "Multidimensional On-line Bin Packing: Algorithms and Worst-Case Analysis,” *Operations Research Letters*,
Vol. 8, February, 1989.

[17] U. Dayal and H. Hwang, “View Definition and Generalization for Database Integration in MULTIBASE: a system for heterogeneous distributed databases,” *IEEE Transactions on Software Engineering*, vol. SE-10, no. 6, 1984, PP. 628-644.

[18] A. Deshpande and J. Hellerstein, “Decoupled Query Optimization for Federated Database Systems,” *Proc. 18th International Conference on Data Engineering,* 2002.

[19] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin. ``Highly-Resilient, Energy-Efficient Multi-path Routing in Wireless Sensor Networks." *Mobile Computing and Communications Review*,
vol. 4, no. 5, October 2001.

[20] R. Glushko, J. Tenenbaum, and B. Meltzer, "An XML Framework for Agent-based E-commerce," *Communications of the ACM,* vol. 42, no. 3, 1999, pp. 106-114.\

[21] Halevy, A.Y., Ives, Z.G., Madhavan, J., Mork, P., “The Piazza Peer Data Management System," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, July 2004, pp. 787-798.

[22] S. Heragu, R. Graves, B. Kim, and A. Onge, “Intelligent Agent Based Framework for Manufacturing Systems Control,” *IEEE Transactions on Systems, Man, and Cybernetics*, 2003.

[23] D. S. Hochbaum and D. B. Shmoys, "Using Dual Approixmation Algorithms for Scheduling Problems: Theoretical and Practical Results, *Journal of the ACM*,Vol. 34, No. 1, January, 1987.

[24] C. Hsu., *Enterprise Integration and Modeling: the Metadatabase Approach*, Kluwer Academic Publishers, Boston, 1996.

[25] C. Hsu, M. Bouziane, L. Rattner and L. Yee, "Information Resources Management in Heterogeneous Distributed Environments: A Metadatabase Approach," *IEEE Transactions on Software Engineering,* vol. 17, no. 6, 1991, pp 604-625.

[26] C. Hsu, Y. Tao, M. Bouziane, and G. Babin, "Paradigm Translations in Integrating Manufacturing Information Using a Meta-Model," *Journal of Information Systems Engineering,* vol. 1, September 1993, pp 325-352.

[27] C. Hsu, L. Gerhardt, D. Spooner, and A. Rubenstein, "Adaptive Integrated Manufacturing Enterprises: Information Technology for the Next Decade," *IEEE Transactions on Systems Man, and Cybernetics*, Vol. 24, No. 5, 1994, pp. 828-837.

[28] C. Hsu and C. Carothers, “A Self-Scheduling Model Using Agent-Base, Peer-to-Peer Negotiation, and Open Common Schema,” *Proceedings* 17th *International Conference on Production Research,* Blacksburg, VA, August 2003.

[29] C. Hsu, C. Carothers, and D. Levermore, “A Market Mechanism for Participatory Global Query: A First Step of Enterprise Resource Allocation,” *Information Technology and Management*, vol. 7, no. 2, 2006, pp. 71-89.

[30] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed Diffusion: a Scalable and Robust Communication Paradigm for Sensor Networks,” *Proceedings of the ACM Mobile Communications*, Boston, MA, 2000, pp. 56-67.

[31] W. Kim, ed., *Modern Database Systems*, Addison Wesley, Readings, MA 1995.

[32] R. Kozma and W.J. Freeman, “Basic Principles of the KIV Model and Its Application to the Navigation Problem,” *Journal of the Integrative Neuroscience*, vol. 2, no. 1, 2003.

[33] D. M. Levermore, "Global Database Query in Collaborative Environments," Rensselaer Polytechnic Institute, Troy, Unpublished Ph.D. Thesis, May 2005.

[34] W. Litwin, L. Mark, and N. Roussopoulos, “Interoperability of Multiple Autonomous Databases,” *ACM Computing Survey*, vol. 22, no. 3, 1990, pp. 267-293.

[35] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson,`` Wireless Sensor Networks for Habitat Monitoring," *WSNA'02*, Atlanta, Georgia, September 2002.

[36] R. Min, M. Bhardwaj, S. Cho, N. Ickes, E. Shih, A. Sinha, A. Wang, and A. Chandrakasan. ``Energy-Centric Enabling Technologies for Wireless Sensor Networks." *IEEE Wireless Communications* (formerly IEEE Personal Communications), vol. 9, no. 4, 2002, pp. 28-39.

[37] M. Nandula and S. P. Dutta, "Performance Evaluation of an Auction-Based Manufacturing System Using Colored Petri Nets," *International Journal of Production Research,* Vol. 38, No. 38, 2000, pp. 2155-2171.

[38] V. Prabhu, "Performance of Real-Time Distributed Arrival Time Control in Heterogeneous Manufacturing Systems," *IIE Transactions*, Vol. 32, No. 4, 2000, pp. 323-331.

[39] T. Sandholm, "Making Markets and Democracy Work: A Story of Incentives and Computing," *IJCAI-03 Computers and Thought Award Talk Abstract*, 2003.

[40] R. Shah and J. Rabaey. ``Energy Aware Routing for Low Energy Ad Hoc Sensor Networks." *Proc. IEEE Wireless Communications and Networking Conference* (WCNC), Orlando, FL, March 2002.

[41] A.P. Sheth and J.A. Larson, "Federated Database Systems for Managing Distributed and Autonomous Databases," *ACM Computing Survey*, vol. 22, no. 3, 1990, pp. 183-236.

[42] J.F. Sowa, *Conceptual Structures: Information Processing in Mind and Machine*, Addison Wesley, Readings, MA 1984.

[43] M. Stonebraker, P. Aoki, A. Pfeffer, A. Sah, J. Sidell, C. Staelin and A. Yu, "Mariposa: A Wide Area Distributed Database System," *International Journal on Very Large Databases*, Vol. 5, No. 1, 1996, pp. 48—63.

[44] J. Swaminathan, S.F. Smith and N. Sadeh, "Modeling Supply Chain Dynamics: a Multi-Agent Approach", *Decision Sciences*, Vol. 29, 1998.

[45] L. Tao, "Shifting Paradigms with the Application Service Provider Model," *IEEE Computer,* vol. 34, no. 10, 2001, pp. 32-39.

[46] G. Thomas, G.R. Thompson, C-W Chung, E. Barkmeyer, F. Carter, M. Templeton, S. Fox, and B. Hartman, "Heterogeneous Distributed Database Systems for Production Use," *ACM Computing Survey*, vol.22, no. 3, 1990, pp. 237-266.

[47] A. Tyrrell, J. LaPre, C. D. Carothers, B. Roysam and C. V. Stewart, ``Transparent Migration of Off-Line Frame Rate Vision Systems to Real-Time", To appear in *IEEE Transactions on Information Technology in Biomedicine*, 2004.

[48] UN/CEFACT, United Nations Center for Trade Facilitation and Electronic Business, *Core Components Technical Specification*, version 2.01, November 15, 2003.

[49] W. E. Walsh and M. P. Wellman, "A Market Protocol for Decentralized Task Allocation", Proceedings International Conference on Multi Agent Systems, 1998.

[50] H. Wang, J. Elson, L. Girod, D. Estrin and K. Yao. ``Target Classification and Localization in a Habitat Monitoring Applicaton." *Proceedings of the IEEE ICASSP 2003*, Hong Kong, April 2003.