Electing a Global Mediator in a Three-Layer Mediation Architecture

Onyeka Ezenwoye, Raimund K. Ege, Qasem Kharma, Li Yang and Shakil Siddique
School of Computer Science
Florida International University
oezen001@cs.fiu.edu

Abstract

Our ongoing research project is a system that mediates access to heterogeneous multi-media sources while considering the quality of service to clients. Several intermediate mediator components are involved in the delivery of data. One of the mediators is responsible for assuring the quality of presentation of the mediated information. In this paper, we present the algorithm used to elect this global-mediator.

1. Introduction

Modern information systems often comprise many interconnected subsystems that come together to share complex data. These subsystems, each housed in an individual function, business unit, region, factory, office or organization, come together to share domain-specific data. An example of such a system is one that integrates financial and accounting information, human resource information, supply chain information and customer information [1]. Others include systems that integrate geographical data [13, 14] and securities information [15].

The heterogeneous nature of the multiple sources in these systems does frustrate the promise of seamless integration of information [1]. The heterogeneity among the client devices that access these sources requires that special attention is paid to their individual Quality of Service (QoS) requirements in the delivery of information.

With varying client devices, recently more of which are mobile, users have access to a wide range of multimedia data. These multimedia data, used for applications which can be either presentational or conversational [10], require the monitoring of QoS parameters. These parameters which could be performance oriented, format oriented, synchronization oriented or monetary oriented, all influence the presentation of multimedia data to the user [10].

There are several scenarios where a user might need mediated access to multimedia data from heterogeneous sources. In an ambulatory environment, for instance, a physician might need to retrieve a patient's medical record. These medical records, in forms such as images, video or text may be housed in various organizational units. The physician will need the information delivered to whatever client device they have at the time. This could range from a desktop computer to a PDA with a wireless connection. Prior to delivery, related data would need to be integrated.

There is a need for architectures that deal with the heterogeneity of information received from diverse resources [17]. Architectures that provide a value-added service to higher level applications by abstracting, transforming and integrating matching data [19] while dealing with various QoS issues that might include the intermittent connection associated with mobility [20].

Mediators [22, 23] are typically employed in a situation where the client data model does not coincide with the data model of the potential data sources. They are facilitators that search for likely resources and ways to access them [22, 23]. Although many mediator systems have been proposed for a variety of applications, a major problem often encountered is how to seamlessly query, integrate and present data from heterogeneous data sources.

Our approach to enabling high quality access is to build a layered framework of mediators [6, 21]. Low-layer mediators connect to the actual data sources; the middle-layer mediators provide a logical schema for integrated information while the top-layer mediators deal with the presentation of mediated information to applications.

In this paper we describe the algorithm used to elect the global-mediator. The global-mediator plays a key role in our mediation architecture by being the conduit through which the results of a query are sought and integrated. It also plays a vital part in meeting some the QoS requirements. Our algorithm is derived from a ring election algorithm [5]. The paper is organized as follows; Section 2 briefly describes our three-layer architecture. Section 3 presents the ring election algorithm, section 4 describes our election algorithm. Section 5 contains related work and section 6 summarizes the paper.

2. The Three-layer Mediation Architecture

Here we briefly describe the three-layer architecture, it useful in order to fully understand the concept of the global mediator. Our mediation architecture [6, 21] is based on three layers of mediation; presence, integration and homogenization. Within this architecture, we differentiate between three kinds of mediators. They are the presence-mediators in the presence layer, the mediator-composers in the integration layer, and mediator-connectors in the homogenization layer. This concept of the three-layers

0-7803-8865-8/05/$20.00 ©2005 IEEE.

constitutes a logical but not physical separation of mediators. Our system has been designed to give a high degree of autonomy to the data sources. This gives the data stores the freedom to join and leave the federation of mediated databases as they wish. This also allows the individual data stores to modify, maintain their content and schemas independently. The system thus exhibits a behaviour that is similar to peer-to-peer (P2P) architectures.

2.1. The Presence Layer

The high-level goal of the presence layer is acceptance of requests (queries) from clients and the presentation of the results of those queries. The intermediate level goal of this layer is to make sure the quality of service (QoS) criteria of these requests is met [6, 21]. The main steps taken to achieve this include the monitoring and advertising the QoS parameters of the client/query, the election of a global-mediator for the query, and other synchronization, performance and format oriented QoS functions [21]. The data interchange language between our mediators is XML. Queries are converted to XML in the presence layer before the search; results are converted back from XML to the desired format in this layer.

2.2. The Integration Layer

The mediators in this layer are the means for the discovery of other mediator. Thus the existence of a mediator-composer on a device is vital to achieve interconnection with other mediators [21]. The decomposition of a query, its distribution (search) and integration of the results is done at the integration layer. Due to the dynamic nature of the topology and the autonomy of the sources, the path from the client to the data stores cannot be static but must be dynamically constructed during the search. This dynamic construction also allows the mediators to form a path that best meets the QoS requirements of the client.

2.3. The Homogenization Layer

The third layer, homogenization, is where connection to the data sources is established. Data from these heterogeneous data source and their schemas are converted from their individual formats to a common data language of the mediators. The mediators in this layer act as wrappers/interfaces to the data sources. Unlike the other mediators, mediator-connectors only exist when there is an associated data store [21].

2.4. The Global-Mediator

When a client has a user request, the presence-mediator for that client, amongst some of its other presence layer functions, will elect a special kind of mediator-composer called the global-mediator. This global-mediator will be responsible for the particular query for which it was elected. It will be responsible for composing the path from the client to the data sources that represents the tree for the composition of the query result. A new global-mediator is elected for every new request based on predefined QoS criteria. By new request we mean any request that is not associated with a previous request. In the event that a global-mediator fails, another mediator-composer is elected to replace it. This particularly useful for dealing with the intermittent connection associated with mobility [21]. Figure 1 helps to illustrate our architecture.

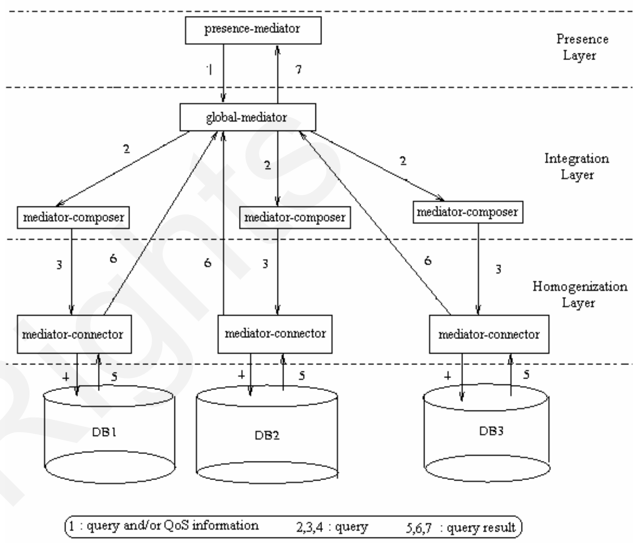

Figure 1: Three-layer architecture

3. The Ring Election Algorithm

This election algorithm [5] is based on the use a ring. It assumes logical or physical process (computing entity) ordering so that each process knows it successor. When a process notices that its coordinator is no longer functioning, it builds an 'election' message. It inserts its own process number in the message and sends the message to its successor. If the successor is down, it goes to the next member along the ring. Before each processor sends the election message to its successor, it appends its own process number to the list, making itself candidate. The election message eventually gets back to the initiating process. This process then changes the message to 'coordinator' and the message is circulated once again, this time to inform everyone who the coordinator is. The coordinator is the list member with the highest number. Figure 3 illustrates the algorithm.

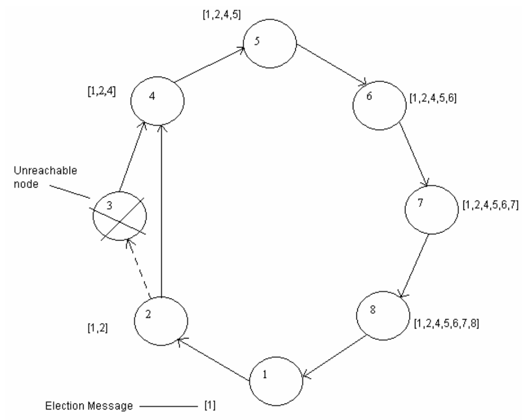

Figure 3: The ring election algorithm

4. Electing the Global-Mediator

At the inception of a client request, the presence-mediator goes through the process of electing a *mediator-composer* to be the *global-mediator* for that request. This is in order to find the best possible *mediator-composer* to serve the request. Selection criteria are QoS parameters such as available bandwidth, processing power, network traffic and load. The elected mediator will be the one that best meets the required QoS criteria. Because our architecture is dynamic and distributed, that is, lacks central control or coordination, the selection of *global-mediator* needs to be dynamic.

Our election algorithm is based on the ring election algorithm [5] with some modifications. The ring algorithm is suitable because it assumes logical ordering of processes and that this forms the basis for election message circulation. Like the ring algorithm, our algorithm circulates an election message. Also, the initiator of the election is a candidate for the election. Note that, it is the *presence-mediator* that initiates the election and only *mediator-composers* can play the role of *global-mediator*. However, for a *presence-mediator* to exist there has to be a *mediator-composer* on the same device (section 2.2). So, when a *presence-mediator* initiates an election, its *mediator-composer* becomes a candidate. This ensures that at least there is one candidate and that a *global-mediator* is always elected.

One of the modifications to the ring algorithm is that our algorithm does not send a coordinator message. Whereas the elected *mediator-composer* will coordinate the search for and integration of results from other mediator, it does not serve as a leader, in the real sense, to the election participants.

Unlike the election message of the ring algorithm that contains a list, our election message contains two variables, V and k. V contains the heights of the participants and k is a hop count. So, when a node i receives an election message, it adds $height_i$ to V. The height of each node is a 4-tuple (i, β_i, ρ_i, π_i). The first component i, is the id of the node, the remaining three component are values that represent the device's bandwidth, processor speed and load. These tree components constitute the *quality factor* of i. There are various ways to calculate these values, of which is outside the scope of this paper. When the election message gets back to the initiator, the *quality factor* from all the heights in V are lexicographically compared and the node with the highest is elected. It is from V that the next best candidate is picked to replace the *global-mediator* in the event of failure (section 2.4) of the *global-mediator*. The order of the components in the *quality factor* can be rearranged depending on which component is deemed the most desirable QoS parameter for the application. In the *quality factor* above, bandwidth of the nodes is the most important component and load is the least important. Although here three components comprise the *quality factor*, in practice, more QoS variables can comprise the *quality factor*.

The hop count k is used to prevent the election message from traveling perpetually through the network. This count, which is set arbitrarily, does constitute a draw back to the algorithm. This is because it sets a ceiling to the number of nodes that can take part in the election, thus the "best" node may not get to take part in the election.

The pseudocode below summarizes the algorithm. We explain the following variables:

- *initiator* : the id of the node which started the election
- *gm* : the id of the node which is elected as global mediator
- *next* : the id the neighbor of i to which i sends the election message
- N : the set of ids of neighbors known to i
- *sender*: the id of the node from which the election message was received

A. When node i receives an election message:
  1. if *initiator* = i then
  2. $gm = j$ such that $x = \max\{(\beta_j, \rho_j, \pi_j) \mid j \in V\}$
  3. else
  4. $V = \{V\} \cup \{(i, \beta_i, \rho_i, \pi_i)\}$
  5. $k = k - 1$

B. When node i is ready to send election message to the next node
  1. if $k = 0$ then
  2. *next* = *initiator*
  3. else
  4. *next* = x such that $x \in N$ and $x \notin V$
  5. *next* = *initiator*

The condition ($x \notin V$) in step 4 of Part B is used to prevent looping election messages. That is, i will not send the election message to a node that has already been visited. In the event that all the neighbors of i have been visited, the election message is sent to the initiator (step 5)

5. Related Work

A lot of work has been done on mediation systems [4, 16, 18, 12, 9, 8, 7]. Some of these architectures are centralized and as such, all search and integration is done through a single mediator. Some are distributed [9, 16, 18] and mediators are able to access and communicate with each other in coordinating search and integration. None of the works known to us employs the concept of dynamically electing a mediator, based on some QoS criteria, to play a central role in serving a request.

The idea of electing a capable node is similar to the concept of super-peers [11] in peer-to-peer (P2P) networks. In certain P2P networks, the disparity between peer capabilities is being exploited [1, 3] by selecting more capable peers (super-peers) to play special or additional roles in the network. Although it is thought that these super-peers are appointed by a central server, the algorithms used for selecting these peers are proprietary [11]

6. Conclusion

We presented an algorithm for electing a global-mediator in distributed mediation architecture. The algorithm fulfils the need to dynamically elect a mediator based on some desired QoS criteria. It ensures that there is always a global-mediator for every request. The hop count is a limitation of the algorithm because it sets a ceiling to the number of nodes that can take part in the election and thus the "best" node may not be considered in the election.

7. References

[1] Y. Zhu, H. Wang and Y. Hu, "A Super-Peer Based Lookup in Structured Peer-to-Peer Systems", in *Proceedings of the 16th International Conference on Parallel and Distributed Computing Systems (PDCS'03)*, Nevada, USA, August 2003.

[2] T. H. Davenport, "Putting the Enterprise into the Enterprise System", *Harvard Business Review*, July 1998.

[3] R. Kurmanowytsch, M. Jazayeri and E. Kirda, "Towards a Hierarchical, Semantic Peer-to-Peer Topology", in *Proceedings of the Second International Conference on Peer-to-Peer Computing* (P2P'02), IEEE Computer Society Press, Linköping, Sweden, September 2002.

[4] H. Garcia-Molina, Y. Papakonstantinou, D. Quass, A. Rajaraman, Y. Sagiv, Y. D. Ullman, V. Vassalos, and J. Widom, The TSIMMIS approach to mediation: Data models and languages. *Journal of Intelligent Information Systems*, 1997, pp. 117 – 132.

[5] A. S. Tanenbaum and M. Van Steen; "Distributed Systems: Principles and Paradigms". Pearson Education, September 2001, pp. 263.

[6] R. K. Ege, L. Yang, Q. Kharma and X. Ni, "Three-layered Mediator Architecture based on DHT ", in *International Symposium on Parallel Architectures, Algorithms, and Networks* (I-SPAN), IEEE Computer Society Press, Hong Kong, May 2004.

[7] V. Josifovski, and T. Risch, "Comparison of Amos II with other Integration Projects", Technical Report, EDSLAB/IDA, Linköping University, April 1999.

[8] M. Karjalainen, "Integrating Heterogenous Databases with the Functional Data Model Approach", http://www.cs.chalmers.se/~merjaka/report04d.pdf, January 2004

[9] T. Katchaounov, "Query Processing for Peer Mediator Databases", Doctoral thesis, Uppsala University, 2003

[10] A. Vogel, B. Kerhervé ,G. V. Bochmann and J. Gecsei, "Distributed Multimedia Applications and Quality of Service: A Survey", in *Proceedings of the 1994 conference of the Centre for Advanced Studies on Collaborative research*, Toronto, Ontario, Canada, October 1994

[11] B. Yang, H. Garcia-Molina, "Designing a Super-Peer Network", in *Proceedings of the 19th International Conference on Data Engineering*, Bangalore, India, March 2003

[12] T. Risch, V. Josifovski, and T. Katchaounov, AMOS II Concepts, http://www.dis.uu.se/~udbl/amos/doc/amos_concepts.html, June 2000

[13] E. Leclercq, D. Benslimane, and K. Yetongnon, "ISIS: A Semantic Mediation Model and an Agent Based Architecture for GIS Interoperability", in *International Database Engineering and Applications Symposium* (IDEAS), Montreal, Canada, August 1999, pp. 87 –91.

[14] L. Stoimenov, S. Djordjevic–Kajan and D. Stojanovic, "Integration of GIS Data Sources over the Internet Using Mediator and Wrapper Technology", in *Proceedings of the 10th Mediterranean Electrotechnical Conference* (MELECON 2000), Cyprus, May 2000, Volume 1, pp. 334-336

[15] A. Moulton, S. E. Madnick and M. Siegel, "Knowledge Representation Architecture for Context Interchange Mediation: Fixed Income Securities Investment Examples", in *Proceedings of the 12th International Workshop on Database and Expert Systems Applications* (DEXA 2001), IEEE Computer Society, Munich, Germany, September 2001, pp. 50-54

[16] A. Tomasic, L. Raschid, and P. Valduriez, "Scaling Access to Heterogeneous Data Sources with DISCO", *IEEE Transactions on Knowledge and Data Engineering*, 1998, Volume 10, pp. 808 – 823

[17] G. Wiederhold, "Mediation to Deal with Heterogeneous Data Sources", in *Proceedings of the Second International Conference on Interoperating Geographic Information Systems*, Zurich, Switzerland, March 1999

[18] L. Yan, M. T. Özsu, L. Liu, "Accessing Heterogeneous Data through Homogenization and Integration Mediators", in *Proceedings of the Second IFCIS International Conference on*

Cooperative Information Systems, South Carolina, USA, June 1997, pp. 130-139

[19] G. Wiederhold, “Value-added Mediation in Large-Scale Information Systems”, in *Proceedings of the Sixth IFIP TC-2 Working Conference on Data Semantics* (DS-6), Georgia, USA, May 1995, pp. 34-56

[20] C. Mascolo, L. Capra, and W. Emmerich, “Middleware for Mobile Computing (A Survey)”, in *Networking 2002 Tutorial Papers*, volume 2497 of LNCS, Springer, 2002.

[21] O. Ezenwoye, R. K. Ege, L. Yang and Q. Kharma, “A Mediation Framework for Multimedia Delivery” in *Proceedings of The Third International Conference on Mobile and Ubiquitous Multimedia*, Maryland, USA, October 2004, pp. 251-256

[22] G. Wiederhold, and M. Genesereth, “The Conceptual Basis for Mediation Services”, *IEEE Expert*, Volume 12 No.5, September 1997, pp. 38-47

[23] G. Wiederhold, “Mediators in the Architecture of Future Information Systems”, *IEEE Computer*, March 1992, pp. 38–49