1-1-2005

Adaptive Replication and Access Control of Multimedia Data in a P2P Environment

Sanjay Kumar Madria
Missouri University of Science and Technology, madrias@mst.edu

Sanjeev Agarwal
Missouri University of Science and Technology

Recommended Citation

S. K. Madria and S. Agarwal, "Adaptive Replication and Access Control of Multimedia Data in a P2P Environment," *Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05)*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2005.
The definitive version is available at https://doi.org/10.1109/DEXA.2005.42

Adaptive Replication and Access Control of Multimedia Data in a P2P Environment

Sanjay Madria and Sanjeev Agarwal
Departments of Computer Science & Computer Engineering
University of Missouri-Rolla, Rolla, MO 65401
(madrias, sanjeev@umr.edu)

Abstract: This paper explores some of the ideas and solutions related to replication and access control of multimedia data in a hierarchical P2P environment. We provided overview of the techniques to generate multiresolution of multimedia data and explored error recovery and access control issues.

1. Introduction

The mobility of the devices in mobile P2P (M-P2P) raises many issues for availability of the data. In addition, the memory and power constraints of the devices make it difficult to store and transmit multimedia documents of large sizes. Thus, the locality of the requested documents is a desirable feature for the performance reason. The replicating multimedia documents thus require deciding whether a given node should replicate the data with the same resolution or compute the desired resolution at run time. The other alternative is to request the desired resolution of data from the nodes hosting it. Also, many different resolutions of images are desired in the mobile P2P network to satisfy different QoS requirements and also to secure the good quality images from unauthorized users. In this paper, we explore some of these issues.

2. Hierarchical Document Resolution Model (HDRM)

Access to data by intruders is becoming easier [Zein] as peers' access replicated information over decentralized channels any time anywhere. Adaptable security can satisfy and perform well and deal with the overheads associated with traditional approaches. In addition, it is not always possible to keep a stringent security requirement and to simultaneously provide high-resolution data to a user due to access restrictions imposed. A peer may be satisfied by low-resolution data such as low pixel images before requiring the better quality image. Such results [ReFS92,SaCh98,MafB03] may be provided due to non-availability of the secured data, access restrictions imposed on the users/documents, or due to some real-time QoS constraints to meet a deadline. A peer may refuse to share original documents owned but may be willing to co-operate by providing some low-resolution replicated copy. Peers may have to perform some method executions/transformation of documents before delivery. The peers further can use the methods supplied along with the copy to generate lower resolution data. This will also reduce I/O, storage and communication costs. When a query is posed, a peer can decide to answer it immediately using the local low-resolution document version and later the high-resolution document can be obtained from the owner. This will allow a peer to have a cost model where peers can charge for QoS (Quality of Service) provided. *A hierarchical document resolution model (HLDRM) for providing data at different access control level needs to be designed and analyzed for supporting flexible access control policies based on resolution of data.*

Adaptive processing of users requests in the P2P environment will use low-resolution documents available locally. This does not require global document and constraint knowledge. In this approach, a peer feeds the information progressively to others using local data at different resolution along with methods available with replicated copies. Requests can be answered efficiently using documents from one of more peers holding multi-resolution copies. A local peer can index documents with different access restrictions and methods used to generate copies. The owners will also store error residues of images to recover documents in case they are corrupted. Multi-resolution documents must be transparent to peers without sacrificing their ability to obtain specific resolution. Object-oriented approaches allow approximation but the relationship between levels of resolution is not part of these models; For example, a peer must submit a query repeatedly till the desired resolution is obtained [GFJK+03, SaGG02]. For performance consideration, a peer has to guess which resolution will satisfy the constraints. A multi-resolution document model will adapt and deal with such issues. This model will provide a high availability and implicit security.
The research issues are summarized as follows:

1529-4188/05 $20.00 © 2005 IEEE

1. Algorithm design for hierarchical resolution model and storage structures to handle different resolutions of same copies. Analytical study of a cost model to process queries at different resolution considering I/O, storage and communication cost. How the cost model will change by adaptive document processing techniques verses the QoS required by the peers?
2. How to fragment and replicate the multimedia documents at different resolution? How can a peer decide when to regenerate the low-resolution data? How to incrementally generate the required resolution documents from various fragments?
3. How to trade resolution v/s response time and QoS? What policy to use for selective revelation of data to others and ensure access restrictions?
4. How to recover images from error and authenticate them?
5. Algorithm design for reverse role based access control model for dynamically changed roles of replicas and users. In reverse access control, the peers would like to provide the replicated copy with the similar execution methods.

2.1 Replication of the Multi-resolution Documents

The proposed replication scheme is to consider the number of queries per object, document size, and trust of the peers hosting and the quality of service requested by peers who access the documents. The document size should be considered low-resolution smaller size documents require less memory. The proposed approach is to consider the probability of access frequency P_j of the document j. The probability of access can be combined with the average time interval T of access. Let the trust of the peer for a document D with a resolution r be D_r. Let S_r be the size of the document with resolution r. Thus, a function for a replicating factor (RF) for a document j can be given by the following function:

$$RF = \left(P_j * T* (D_r / S_r)\right)$$

Once a document is selected for replication then the next question is which nodes should replicate it. This can be calculated by taking into account the access probability at that node, the time the server is up during that time period, time the server is static in that region, and the trust value of the node and the available memory.

We propose to cluster the documents with similar RF values. In addition, if fragments of a document are replicated then we need to first find the cluster so that all the fragments are available in close proximity. One technique is to form the bipartite graph of the nodes hosting the fragments of the related documents with similar resolution. We will explore this technique in addition to our work in [HaMa04] and evaluate by varying different size, resolution, hops, and access probability of documents. Once clusters are identified, one can replicate the clusters and relocate the replicas over a period of time based on the parameters discussed.

We propose to replicate the documents with similar RF factor in a cluster with some reasonable hops (this will be decided by the extensive simulations) to provide reasonable response time. The higher resolution data will be closer from the hosts whereas low resolution data will be kept far from the peers. The immediate update scheme will be used to update the replicas with higher resolution and lazy-replication will be used for lower resolution data. Thus, the cost of updates will be lower for lower resolution data.

In addition, there is a need to investigate the integration of QOS requested by users to decide the location of replicas of the documents [TaXu04]. Since users are at fixed locations, we propose to design heuristics to balance the replica allocation with the QoS requested by the users based on the current location of the users along with the bandwidth available. There is a need to consider different P2P architecture [YaGa01] and evaluate the schemes designed.

3. Generation of Documents at Different Resolution

P2P systems can store different resolution of data while replicating (see Figure 1). These could be based on transformation, quality degradation by reducing the number of pixels, content transcoding, etc. Consider two documents D_i and D_j and a method $F_{ij} : D_i \rightarrow D_j$ defines a generalization of the document D_i to D_j . The generalization function depends on the application, context, users, etc. For example, a coarsen function may combine two objects into one, or can aggregate multiple objects. Intuitively, an aggregated object has a lower resolution than original entities. In generalizing data, there could be type mismatch and one has to define these types dynamically at run-time (static types can create problems in case documents can not be stored under pre-defined type). There could be intra-type and inter-type of aggregation. Current systems [FuMa03] do not support resolutions, for example, if a user wants multimedia data at different resolution, then she has to submit the request again and the burden of generating multi-resolution images lies on the users.

HLDRM model is defined where the lowest layer of the documents corresponding to the primitive and the high-resolution most secured information stored and with higher layers storing more general and low-resolution information extracted from one or more lower layers. Some the methods used to reduce the resolution of a multimedia document are as follows:

Sampling: A sample of an attribute value is returned. For example, giving a sample picture of an attraction or part of a video could be returned.

Transformation: High-resolution pictures consume more storage and bandwidth to download. Therefore, it is desirable to transform the resolution of the images to a lower resolution, or to reduce the number of pixels used in the visual objects.

Concept Hierarchy Climbing: A primitive concept is replaced by a more general concept using a concept hierarchy, which organizes concepts according to their generality levels. A user that has not been granted access privileges to the lower layer concepts could still be answered using a generalized concept at higher layers.

Categorization: In some cases, a user may request a document which is highly sensitive and available only to authorized users then instead of giving the particular document, the system can chose some other similar document from the cluster and present to the user.

Aggregation: Aggregate functions such as "part-of" an image can be used and if a user selects a specific part of an image then the complete image can be provided.

Summary: Summarize the contents of a text attribute. For example, an unpublished paper can be replaced by its short abstract.

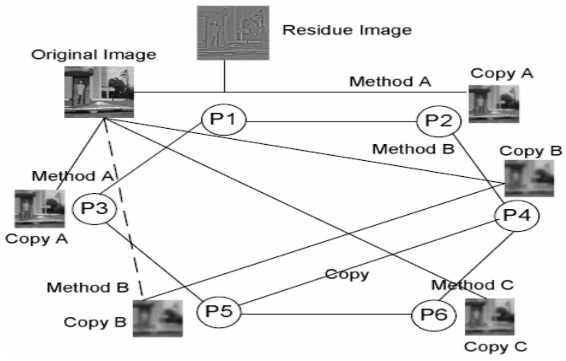

Figure 1. Hierarchical Multi-Resolution Model

These standard sets of methods will be designed, and objects that return the similar results will be grouped together. Application developers may add their own methods. Specifications of the methods will be standard while implementations of the methods may be system dependent. The behavior of the methods at different levels will be analyzed in terms of its operational cost, bandwidth utilization and storage requirements. There should be a balance between the number of generalization levels and storage and processing cost. This will be evaluated by means of experiments and also by developing an analytical model.

3.1 Storage of Multi-resolution Documents

Special storage structure will be developed to exploit lower-bandwidth requirements of low-resolution documents to manage cache storage of different speed to provide good performance across the entire spectrum of resolutions. For example, the quad-tree representation (Q-hash) [FrMS04] of spatial-temporal queries is a multi-resolution data structure, which can provide queries at different levels of accuracy. We propose the use of Q-hash index structure to maintain distributed indexing for changing documents will be developed.

Since there are many possible ways to vary a resolution of a document using many different methods, it is practically impossible to store all possible generalizations a priori. Too many versions will cause excessive interactions among peers that will increase the communication cost. Thus, we propose methods to be associated with the documents. The number of document versions in case of XML documents will be reduced using [ChMB04] algorithm so that only one document version with a sequence of deltas can be stored, and a document can be constructed from the deltas at run time. Similarly, in case of multimedia objects, error image residues will be stored at owner's site along with the original image. One static way is build and store documents by applying most popular execution methods.

The number of levels of in a document hierarchy is decided based on the application and the methods used at each level independent of the query patterns. The Dynamic way is to build document version depending on the query patterns and peers usability pattern and access level. An agent will responsible for keeping track of different attributes/methods peers request as part of their requests in a cluster. The agent determines and generates different levels of HLDRM at different time periods. The design of an agent-based architecture will construct the Dynamic HLDRM. The idea is to have an agent at each peer; these agents can interact and learn from the peers' behavior. The PIs will investigate algorithms to cluster queries and methods/attributes that are accessed in a group based on some data mining techniques. Using this knowledge, the agent will decide what types of queries and attributes are asked frequently by peers and will generate HLDRM. Some guidelines for building the HLDRM are discussed here: (a) The appropriate generalizations to form useful layers of documents based on the applications, access restrictions, peer profiles, device configuration, etc. (b) In general, the layers can be constructed based on heuristics and domain knowledge. It should also be noted that the higher layers should be adaptive since peers' interests, their access behavior and access permissions change over time. (c) With methods for generalization available, the important question is how to selectively perform appropriate method executions to form useful layers of the documents. In principle, there could be a large number of combinations of possible generalizations by selecting different sets of methods/attributes to generalize and selecting the levels to reach in the generalization. However, in practice, few layers containing most frequently referenced methods and patterns will be sufficient to balance the implementation efficiency and practically. (d) Frequently used methods and patterns should be determined before generation of new layers of a HLDRM by the analysis of the statistics of query history or by receiving instructions from users or experts.

Evaluation *The effectiveness of the data model will be evaluated based on storage and bandwidth requirements, responsiveness, accuracy, adaptability, graceful degradation, contextual and previous usage, etc using different P2P architectures [YaGa01].*

3.2 Access Control of Replicated Documents and Clustering

Both the discretionary access control model and the mandatory access control ideas [JaSa92,StTh90] will be extended to work on HLDRM. Using discretionary access control in HLDRM, permissions can be granted. Using mandatory access control in HLDRM, documents and peers are classified into classes of security concern, such as top secret, secret, restricted, and public. Each security class of a peer represents the security clearance of the peer. Each security class of data determines the confidentiality of the documents. A peer is authorized to access data in the same or lower security class. Every peer, replica, methods in HLDRM is assigned a security class, which is specified in the security hierarchy. Moreover, if a document t' is generalized from another document *t*, the security class of *t*' should be the same as or less than that of *t*. The methods of *t*' should also have the same class as or a less secure class than that of their counterpart in *t*.

If a replica is allowed to execute a method M on a client request then all other replicas in that role must be allowed to execute that request. A replication role is the set of all replicas with equivalent functionality. If a set of replicas is allowed to execute a method M, while others are not than those replicas will belong to different clusters. Thus, we will develop a scheme to encode replicas with particular methods based on their role and peers access level. If these replicas get updated then we need to develop schemes to propagate the updates the right replicas in different clusters.

There are some important issues such as how to dynamically control role based access in different clusters? Where to store the security classes and hierarchy? Efficient algorithms are needed to decide whether an access to be granted or denied. A peer may have multiple roles; a different role in each cluster. How roles will be affect access control for same peer in different clusters? Automatic mechanism for verification of permissions will be designed. Rules and policies for permission assignment will be established. A more important issue is when permission is granted to a relation; how it affects permissions on relations that are generalized from it and relation it generalized from. Thus, dynamic adaptive role based access control techniques need to be studied. These issues need to be explored further. .

3.3 Error Recovery and Authentication in HDRM Multimedia Documents

An error may occur in the storage of images or image may be corrupted for various reasons at any of the peer' site. This error may affect a fragment of the multimedia document or the document as a whole. Therefore, it is important for the site to recover the document. The proposed hierarchical replicated document provides multiple avenues for error recovery. Two main source of error recovery are the redundancy afforded by hierarchical replication at different peers' sites and residue document saved at the parent site. In figure 1, P1 is the parent site for the image document. When a copy of this image is provided to the site P2 using method A, a corresponding residue image shown in figure is saved at the parent site. If an error occurs in the copy of the image at site P2, it can simply be re-created from the original image at site P1 or copied from other site with the same resolution image. If the error occurs in the residue image at parent site P1, it can be re-generated using original image and the appropriate method. If the error occurs in the original image at site P1, the site P1 can request for a copy from say site P2. Since a known method A is used to create this copy, an inverse transform can be applied and error-less original is recovered using the residue image available at P1. Note that the above discussion motivates error recovery in image data stream. However, the same concept with appropriately defined residue document is true in case of video, text, graphics and other multimedia data streams. We propose to develop robust algorithms and test by experimentation the parameters such as recovery time for low resolution images verses request for a new copy, distance among different copies, and network conditions.

Another important issue in error resilient document sharing in P2P environment is data authentication. Consider the copy B of the original image in figure 1, where the site P4 receives a copy B of the document from site P1 using method B. Now, site P5 also request a copy B of the document. Since, the copy B exists at site P4, site P1 will simply direct the site P5 to get the copy from P4. However, P5 would like to make sure that the copy it receives is in fact the original and has not been modified by P4. This is achieved using hash encoding method. When the site P5 requests a copy of the original document from site P1, P1 provides a low bandwidth hash code for copy B, along with the name of the site P4. Once site P5 get the copy B of the document from P4, it can run the hash code method and compare the result with the hash code provided by P1 to authenticate the copy. Same method will be used by P1 to authenticate the copy A when it requests it back from site P2 for error recovery.

4. Conclusions

In this paper, we posed research questions and provided overview of some of the techniques for handling replication and access control of multimedia data in a P2P environment. Currently, we are developing solutions to many of the issues raised and discussed here.

References

[CyMB04] Yan Chen, Sanjay Madria, Sourav Bhowmick, DiffXML–Change Detection in XML Data, in proceedings of 9th Intl Conference on Database Systems for Advanced Application, (DASFAA 2004), Lecture Notes in Computer Science, Vol. 2973, pp. 289-301, Springer-verlag, March 2004

[FuMa03] Y. Fu and S. K. Madria, Multilevel Database Model for Mobile Computing, NSF workshop, Providence, RI, USA, and also in Mobile Data Management, 2003, proceedings by Springer-verlag, LNCS, Vol.2574, pp.381-385.

[FrMS04] D. Francis, S. K. Madria, and C. Sabharwal, Constraint Based Hash Indexing for Moving Objects, under communication.

[GFJK+03] Z. Ge, D. R. Figueiredo, S. Jaiswal, J. Kurose and D. Towsley, Modeling Peer-Peer File Sharing Systems, in Proceedings of IEEE INFOCOM, 2003.

[HaMa04] Takahiro Hara, Sanjay Kumar Madria: Dynamic Data Replication Using Aperiodic Updates in Mobile Adhoc Networks. DASFAA 2004: 869-881, extended version under communication.

[JaSa91] S. Jajodia and R. Sandhu,, Towards a Multilevel Secure Relational Data Model, in Proceedings of ACM International Conference on Management of Data (SIGMOD, 1991), pp. 50-59, Denver, Colorado, USA.

[MaFB03] S. K. Madria, Y. Fu and S. Bhowmick, A Multi-layered Database Model for Mobile Environment, in Proceedings of International Conference on Mobile Data Management, 2003, LNCS, Vol.2574, pp.381-385.

[ReFS92] R. L. Read, D. S. Fussell and A. Silberschatz, A Multi-Resolution Relational Data Model. VLDB 1992: 139-150.

[SaCh98] R. Sandhu and F. Chen, The Multilevel Relational (MLR) Data Model, TISSEC 1(1): 93-132 (1998)

[SaGG02] S. Saroiu, P. K. Gummadi, and S. D. Gribble, A Measurement Study of Peer-to-Peer File Sharing Systems, in Proceedings of Multimedia Computing and Network (MMCN), 2002, San Jose, CA.

[StTh90] P. Stachour and B. Thuraisingham, Design of LDV: A Multilevel Secure Relational Database Management System. TKDE 2(2): 190-209 (1990)

[TaXu04] X. Tang and J. Xu, On Replica Placement for QoS-Aware Content Distribution, in Proceedings of INFOCOM, 2004

[YaGa01] B. Yang and H. Garcia-Molina, Comparing Hybrid Peer-to-Peer Systems, in Proceedings of 27th International Conference on Very Large Data Bases, pp 561-570, September 11-14, 2001, Roma, Italy.

[Zein] D. Zeinalipour-Yazti, Exploiting the Security Weaknesses of the Gnutella Protocol - Department of Computer Science University of California Riverside, CA 92507, http://citeseer.nj.nec.com/572398.html.