Volume 2, Issue 11, November 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Survey for Optimizing the Attachment Files in E-Mail

K.SriRam Kumar
Dept of Computer Science and Engg
K.S.R.College of Engineering,
Tiruchengode,India.

Mrs.E.Baby Anitha
Dept of Computer Science and Engg
K.S.R.College of Engineering,
Tiruchengode,India.

Abstract - In this work we present a Survey of different methods of effective distributed memory access and with the special consideration of effective usage of data retrieval using the meta data technology. By implementing the more secured data access in the distributed shared memory and by adapting the meta data for fast and effective access of the distributed data we can rapidly reduce the unwanted usage of memory in the distributed network.

Index Terms – Distributed shared memory, Meta Data Management, Cluster File System, and Inverted Files.

I. Introduction

Now a day's usage of E-Mail is increasing rapidly and hence large amount of space is needed to store and process all the Emails. Today's email systems are based on a store-and-forward model. Email servers accept, forward, deliver and store messages. While attaching the file, all the attached files are stored individually for each and every user and the contents of the received mails are similar for all. To eliminate the redundancy of stored files, a single copy is kept in a centralized server. Just by using the reference we can view or download the attached files from the centralized server for data storage whenever needed.

II. Metadata

Metadata is defined as data providing in sequence about one or more aspects of the data, such as:

- Means of making of the data
- Purpose of the data
- Time and date of creation
- Creator or author of data
- Location on a computer network where the data was created
- Principles used

A. Web Search Engine:

Against the user keyword for searching the data a list of keywords will be displayed and the relevant web page that contains the key word will be loaded when selected.[1] describes the methodology of indexer module for extracting all the keyword fro each page and records the URL where each word occurred. The result is from "lookup table" that can point page of the URL where the keyword occurred.

B. Metadata in Learning

Searching the learning object by using the keyword against the specific application becomes the complex operation. To reduce the complexity [2] introduce the concept of Application Profile (AP), which standardize and act as portal for learning resources. The AP has become one or more name space which act as the data element. For learners with the help of meta data can search access, use & reuse of learning objects. Learning object Repository (LOR) which stores metadata information as keywords & all of the LOR are combined and optimized for specific application.

C. Metadata Management in Large File Systems

For Ultra large file system [3] presents a scalable and decentralized metadata lookup scheme with logically organize metadata servers (MDS) as multilayered query hierarchy, and exploits grouped boom filter for effective routing of metadata request to describe MDS through hierarchy.

D. Distributed Metadata Management

For decentralizing metadata management with in a collection of metadata servers,[4] presents a technique called Hierarchical Bloom Filter Arrays (HBA) for mapping filenames to the metadata servers holding their metadata. Two levels of arrays. One Level of array, with lower accuracy and representing the distribution of the entire meta data

whereas the other array, with higher accuracy. Both arrays are replicated to all metadata servers to support fast local lookups.

E. Metadata service of the Ursa Minor distributed storage system

[5] presents the metadata check of the Ursa Minor distributed storage system scales metadata throughput as metadata servers. While doing so, it correctly handles metadata operations that involve items served by different metadata servers, consistently and atomically updating the items. It reusing existing metadata migration functionality to avoid complex distributed transaction protocols. It will assign ID for each item to minimize the occurrence of multi server operations. Ursa Minor's allows one to apply a preferred feature with less difficulty and with minimal performance penalty.

III. Cluster File System

It is a distributed file system consists of group of servers that perform all work together for providing high performance service to their clients. To the clients, the group of Servers is transparent - it is just "the file system", but it internally deals with distributing requests to elements of the storage cluster.

A. Metadata Distribution Techniques for Cluster File Systems

Now a day's super computers that suggest decentralized metadata processing technique are based on the large clusters. For maximizing metadata throughput, a best metadata distribution policy should automatically balance the namespace locality and even distribution. [6] Propose a novel metadata distribution policy called Dynamic Dir-Grain (DDG), which seeks to balance the requirements of keeping namespace locality and even distribution of the load by dynamic partition of the name space into size-adjustable hierarchical units. From the outlook of file system reliability, metadata consistency is an equally important issue but it is difficult by dynamic metadata distribution. [6] Proposed a reliable metadata processing protocol which combines the two-phase commit algorithm with metadata processing to reduce overheads.

IV. Distributed File System

File systems that allow access to files from multiple hosts sharing the same computer network. This makes it possible for multiple users can share files and store resources parallel.

A. Distributed Memory Multiprocessors Framework

In distributed memory multiprocessor system, to improve the efficiency the framework have been designed that maps the context automatically by choosing the good data and parallelization strategy which allows the alignment, distribution and redistribution problems to solve together by a single graph representation. The Communication Parallelism Graph (CPG) is designed with a structure for holding symbolic representation regarding the potential data movement and parallelism inherits to the whole program.

B. Information Retrieval

[8] focused on the analytical model of performance in distributed computing, by predicting the performance of different collection fusion in analytical technique scenario. Information retrieval system works on both single and multiple processors, which increases the understanding of that system behavior. [8] has focus on the analytical model for describing the performance of single and multiple processors, together even the processor have the same or different parameter value.

C. Speculative Locking Protocols

Speculative Locking (SL) protocol is used for trading extra processing resources in Distributed Database System and increase its performance. After processing the data object SL produces the after image for the object. The transaction which is waiting for the processing will access both before and after images and retains one execution based on the termination mode. SL increases the parallelism by carrying out multiple executions for a transaction. By implementing the SL, the real fact will be, transactions that are submitted will be commit that abort. [9] proposes that by reducing the number of speculative execution, the transaction of the process will be done efficiently. The simulation result specifies that the performance of the variant significantly improves in Distributed Database System by using two phase locking protocol where transaction needs longer time.

D. Fragments Allocation:

To improve the efficiency of the distributed data base, the data base has to be divided into independent fragments and located at the various sites in the distributed network. The allocating of these fragments I the distributed network is the difficult task, as it can be done by only the mathematical programming on the linearization of nonlinear binary integer programs and it will be effective for very small problems only.[10] formulates the programming for these integer with no redundant version of the fragment allocation problem. This approach is effective during the availability of

capacity restriction. By reformulating the approach of [10] it will be effective on relatively large problems and also the need of heuristic approach will be reduced.

E. Hierarchical Computation Model

Commercial process increases the demand on memory and storage subsystem in server which increases traffic in I/O and memory buses. To reduce the movement of data between the storage subsystem and the processing unit, [11] proposes a concept called Hierarchical Computing (HC), which distributes the processing elements in storage hierarchy. The programming model proposed in [11] will decompose data base query and distributes to the different layer of the hierarchy depending on the affinity of the task to the particular layer and executes the query, The result from all the layers will be merged to form the final result of the single query. During evaluating the data transferred between the processor and memory is reduced and also the speed of the execution of the process is increased.

F. Data Accessibility

Large-scale distributed systems are loosely coupled in nature, and this makes data access unpredictable, and also unavailable in some instance. An increasing number of data-intensive need accessibility for adequate job allocations. [12] focus the technique of accessibility aware resource selection which helps to select nodes which have efficient data access to remote resources. To characterize the accessibility for the node, the observation collected from neighbors are sufficient. The experiments results of [12] shows that the resource selection guided by this principle significantly out perform conventional techniques such as random allocation.

G. Data Access Prediction and Optimization

Now a days large amount of data is produced by scientific applications. For analyzing and processing of those data need large scale computing such as clusters and grid. The studies have been undergone for improving the performance of those data by optimizing the data access. To achieve this technique of data replication, migration, distribution and access parallelism are considered. The main drawback is, they will not consider the application behavior to perform data access optimization. This limitation motivated [13] for the online prediction of application behavior to optimize the data access operation of the distributed system with out considering the past execution. To accomplish such a goal [13] organize application behavior against time series, with this the data will be analyzed and classifies according to their property. The experiments using Optor Sim simulation, proves that the execution time is reduced to 50 percentage, specially while handling large amount of data.

V. Secure Multimedia Database in a Distributed Environment

The Now a days using and sharing of large amount of multimedia information is becoming a common practices for internet based applications, but the security and privacy of information is become the great issue. As multiple multimedia document will be composed in distributed environment, and each document may employ different access control policies for its objects, when integrating all this will leads to great issue. To resolve this [14] proposes a security model that allows creation, storage indexing and presentation of secure multimedia document. This model is based on time and provides a flexible, multilevel access control mechanism that provides clearance-based access to different level of information in a document. It also provides multimedia synchronization mechanism which includes deterministic and non deterministic temporal relations and incomplete timing information among media objects.

VI. Inverted File System

Inverted files is an index data structure which stores a mapping from the content such as numbers , words , etc. to its data base file location, or in a single/multiple document. The main purpose is to allow fast full text searches when the document is added in the data base. It is most popularly used in large scale retrieval system as search engine.

A. Single-Pass Index Construction

In large collection of data the inverted files should be collected efficiently for better result. To improve the efficiency of constructing inverted files [15] proposes a single-pass inversion method that does not require the complete vocabulary of the collected index in main memory, and it can operate at the same speed with in limited resources with high temporal storage requirement. In [15] the inverted files is constructed in segments and reducing the cost of disk accesses during processing of large volumes of data.

B. Constructing Inverted Files

The complexity of parallel programming is hided while constructing the inverted files using Map reduce framework, while is a high level programming model. [16] Proposes a alternative approach with the current and emerging architectures of multi core processors. The developed algorithm will produce parallel parsed streams on a high throughput pipelined strategy, which will be immediately consumed at the same rate by parallel indexers. The result of

[16] hut a light on the extent of the performance cost that may be incurred by using a simpler, higher level map reduce programming model for large scale applications.

VII. Conclusion

This work reviews the current strategy in the metadata management, Cluster file system, Distributed Memory storage, and the concept of Inverted Files with much secured features while accessing the data. With these methodologies we can optimize the attachment files in the E-Mail and reduce the unwanted usage of the mail server memory and can improve the efficiency of the mail server.

References

[1] ArvindArasu, Junghoo Cho, Hector Garcia-Molina, AndreasPaepcke, and SriramRaghavan, "Searching the Web" ACM Transactions on Internet Technology, Vol. 1, No. 1, August 2001

[2] Hilke Stuempel1, Gauri Salokhe1, Anne Aubert1, Johannes Keizer1, AndrewNadeau1, Stephen Katz1, Stephen Rudgard1 , "Metadata Application Profile for Agricultural LearningResources"

[3] Yu Hua,YifengZhu,HongJiang,DanFeng, Lei Tian"Scalable and Adaptive Metadata Management in Ultra Large-scale File Systems" in The 28th International Conference on Distributed Computing Systems

[4]Yifeng Zhu, Member, IEEE, Hong Jiang, Member, IEEE, Jun Wang, Member, IEEE, and Feng Xian, Student Member, "HBA: Distributed Metadata Management for Large Cluster-Based Storage Systems", IEEE Transactions on parallel and distributed systems, vol. 19, no. 6, June 2008.

[5] ShafeeqSinnamohideen,Raja R. Sambasivan,JamesHendricks,Likun Liu, Gregory R. Ganger," A Transparently-Scalable Metadata Servicefor the Ursa Minor Storage System"

[6] Jin Xiong,YimingHu,Guojie Li, Rongfeng Tang, and Zhihua Fan"Metadata Distribution and ConsistencyTechniques for Large-Scale ClusterFile Systems" IEEE Transactions on Parallel and Distributed Systems, VOL. 22, NO. 5, MAY 2011

[7] Jordi Garcia, Eduard AyguadeÂ , and JesuÂsLabarta," A Framework for Integrating DataAlignment, Distribution, and Redistribution inDistributed Memory Multiprocessors" , IEEE Transactions on Parallel and Distributed Systems, VOL. 12, NO. 4, APRIL 2001

[8] Robert M. Losee and Lewis Church Jr, "Information Retrieval with DistributedDatabases: Analytic Models of Performance", IEEE Transactions on Parallel and Distributed Systems, VOL. 15, NO. 1, JANUARY 2004.

[9] P. Krishna Reddy,MasaruKitsuregawa," Speculative Locking Protocols to ImprovePerformance for Distributed Database Systems" IEEE Transactions on Knowledge and Data Engineering, VOL. 16, NO. 2, FEBRUARY 2004

[10] SyamMenon, "Allocating Fragments in Distributed Databases", IEEE Transactions on Parallel And Distributed Systems, VOL. 16, NO. 7, JULY 2005.

[11] Juan Rubio,LizyKurian John, "Reducing Server Data Traffic Using aHierarchical Computation Model", IEEE Transactions on Parallel and Distributed Systems, VOL. 16, NO. 10, OCTOBER 2005

[12] Jinoh Kim, Abhishek Chandra,Jon B. Weissman, "Using Data Accessibility for Resource Selectionin Large-Scale Distributed Systems", IEEE Transactions on Parallel and Distributed Systems, VOL. 20, NO. 6, JUNE 2009

[13] Renato Porfirio Ishii and Rodrigo Fernandes de Mello, "An Online Data Access Predictionand Optimization Approachfor Distributed Systems", IEEE Transactions on Parallel and Distributed Systems, VOL. 23, NO. 6, JUNE 2012

[14] James B. D. Joshi,Zhaohui Kevin Li, HusniFahmi, BasitShafiq,ArifGhafoor," A Model for Secure Multimedia Document DatabaseSystem in a Distributed Environment", IEEE Transactions on Multimedia, VOL. 4, NO. 2, JUNE 2002

[15] Steffen Heinz and Justin Zobel, "Efficient Single-Pass Index Construction for TextDatabases"

[16] Zheng Wei, Student Member, IEEE, and Joseph JaJa, Fellow, IEEE , "An Optimized High-Throughput Strategy for Constructing Inverted Files" , IEEE Transactions on parallel and distributed systems, vol. 23, no. 11, November 2012

