Comparitive Study of Advanced Database Replication Strategies

A. Pramod Kumar [1] M.Tech, B.SATEESH [2] M.Tech
ACE Engineering College, JNTUH[1]
CBIT, (OsmaniaUniversity) [2]

Abstract
In this paper we are comparing two advanced replication strategies namely materialized view replication and multi master replication are analyzed based on the Five partitioning algorithms namely range partitioning, hash partitioning, list partitioning, composite range-hash partitioning and composite range-list partitioning have been implemented for both the replication strategies. The performance of all these partitioning algorithms have been evaluated for each of the replication strategies with simulation results.

Keywords: Partitioning Algorithm , Replication, Distributed Database, Centralized Database.

1. Introduction
Database replication is nothing but copying of data from a Database in one server to a Database in another so that all users share the same level of information. The result is a Distributed Database in which users can access data relevant to their tasks without interfering with the work of others. Distributed Database Management Systems (DDBMS) ensures that changes, additions, and deletions performed on the data at any given location are automatically reflected in the data stored at all the other locations. Therefore, every user always sees data that is consistent with the data seen by all the other users. There are different strategies to replicate data among servers located in different places [1]. A proper replication method has to be determined before implementation of Database replication for huge installations

The centralized approach suffers from two major drawbacks:
• Performance problems due to high server load or high communication latency for remote clients.
• Availability problems caused by server downtime or lack of connectivity. Clients in portions of the network that are temporarily disconnected from the server cannot be serviced. The server load and server downtime problems can be addressed by replicating the Database servers to form a cluster of peer servers that coordinate updates.

To overcome above drawbacks Advanced replication strategy is implemented. Advanced replication is a fully integrated feature of the server; it is not a separate server. Replication uses Distributed Database technology to share data between multiple sites, but a replicated Database and a Distributed Database are not the same. In a Distributed Database, data is available at many locations, but a particular table resides at only one location. A centralized approach manages only one copy of the Database. This approach is simple since contradicting views between replicas are not possible[2].

Replication objects
A **replication object** is a Database object existing on multiple servers in a Distributed Database system. In a replication environment, any updates made to a replication object at one site are applied to the copies at all other sites.

Replication Groups
A replication group can exist at multiple **replication sites**. Replication environments support two basic types of sites: **master sites** and **materialized view sites**. One site can be both a master site for one replication group and a materialized view site for a different replication group. However, one site cannot be both the master site and the materialized view site for the same replication group.

2. Advanced Database Replication Strategies
The advanced Database replication strategies are materialized view replications and multimaster replication

2.1 Materialized view Replication
Materialized view replications are schema objects that can be used to summarize, precompute, replicate, and distribute data. E.g. to construct a data warehouse A materialized view replication provides indirect access to table data by storing the results of a query in a separate schema object. The existence of a materialized view replication is transparent to SQL, but when used for query rewrites will improve the performance of SQL execution. An **updatable materialized view replication [3]** lets you insert, update, and delete. A materialized view on a base table, partitioned table or view and define indexes on a materialized view. A materialized view replication can be stored in the same Database as its base table(s) or in a different Database.

Materialized views replication stored in the same Database as their base tables can improve query performance through query rewrites. Query rewrites are particularly useful in a data warehouse environment.
Partitioning is a powerful capability to help manage unwieldy table's indexes and materialized views replication. Table and index partitioning allows for data to be divided into smaller pieces to be managed and accessed as individual units.

Each partition can have its own unique storage attributes allowing the various partitions to be placed in different table spaces on different storage devices. By utilizing partitioning, the performance of certain queries and maintenance tasks can be vastly improved over using the traditional method of table and index design. Partitioning allows a table, index or index-organized table to be sub-divided into smaller pieces. Each piece of Database object is called a partition. A materialized view replication contains a complete or partial copy of a target **master** from a single point in time. The target master can be either a master table at a master site or a master materialized view at a materialized view site. A **master materialized view** is a materialized view that functions as a master for another materialized view. A **multitier materialized view** is one that is based on another materialized view, instead of on a master table [4].

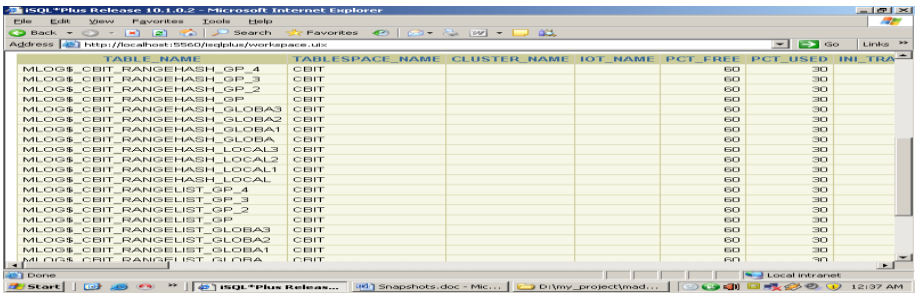

Figure 2.1: shows the list of materialized views replication in remote Database.

2.2 Multimaster Replication

Two types of master replication: single master replication and multimaster replication. Multimaster replication includes multiple master sites, where each master site operates as an equal peer.
In single master replication, a single master site supporting materialized view replication provides the mechanisms to support potentially hundreds or thousands of materialized view sites. A single master site that supports one or more materialized view sites can also participate in a multiple master site environment, creating a hybrid replication environment(combination of multimaster and materialized view replication).
Materialized views can be based on master tables at master sites or on materialized views at materialized view sites. When materialized views are based on materialized views, a multitier materialized view environment. In such an environment, materialized views that have other materialized views based on them are called master materialized views.

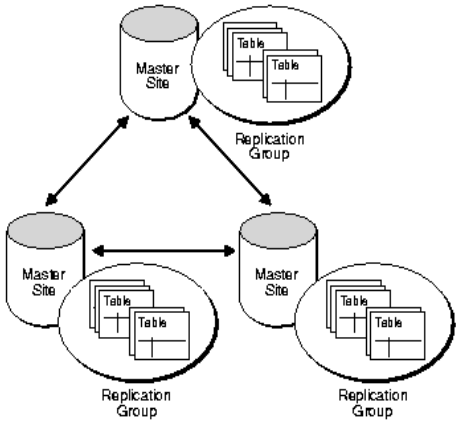

Figure 2.2: shows the multimaster replication

Multimaster replication (also called peer-to-peer or *n*-way replication) enables multiple sites, acting as equal peers, to manage groups of replicated Database objects. Each site in a multimaster replication environment is a master site, and each site communicates with the other master sites. Applications can update any replicated table at any site in a multimaster configuration.

Database servers operating as master sites in a multimaster environment automatically work to converge the data of all table replicas and to ensure global transaction consistency and data integrity.

2.2.1 Single Master Replication

A single master site can also function as the target master site for one or more materialized view sites. Unlike multimaster replication, where updates to a single site are propagated to all other master sites, materialized views update only their target master site. Conflict resolution is handled only at master sites or master materialized view sites. Materialized view replication can contain complete or partial replicas of the replicated table.

Master Sites

A master site can be both a node in a multimaster replication environment and the master for one or more materialized view sites in a single master or multimaster replication environment. The replicated objects are stored at the master site and are available for user access.

Master Definition Site

In a multimaster replication environment, one master site operates as the master definition site for a master group. This particular site performs many of the administrative and maintenance tasks for the multimaster replication environment.
Each master group can have only one master definition site, though the master definition site can be any of the master sites in the multimaster environment. Additionally, the master definition site can be changed to a different master site if necessary. A single master site supporting materialized view replication is by default the master definition site.

2.2.2 Benifits of Multimaster Replication

From a very basic point of view, replication is used to make sure that data is available when and where you need it. The following sections describe several different environments that have different information delivery requirements. replication environment may have one or more of the following requirements[6].

Failover

Multimaster replication can be used to protect the availability of a mission critical Database. For example, a multimaster replication environment can replicate data in Database to establish a failover site should the primary site become unavailable due to system or network outages. Such a failover site also can serve as a fully functional Database to support application access when the primary site is concurrently operational. Net to configure automatic connect-time failover, which enables Net to fail over to a different master site if the first master site fails. Configure automatic connect-time failover in tnsnames. file by setting the FAILOVER_MODE parameter to on and specifying multiple connect descriptors. For more information about configuring connect-time failover

Load Balancing

Multimaster replication is useful for transaction processing applications that require multiple points of access to Database information for the following purposes:

- Distributing a heavy application load
- Ensuring continuous availability
- Providing more localized data access

Applications that have application load distribution requirements commonly include customer service oriented applications.

3. Simulation Results:

VIEW REPLICATION

Initially all possible data retrieving and manipulating operations will be examined on Range, List, Composite Range – List and Composite Range – Hash partitioning techniques.

3.1 Full scan operation on materialized view replication

Full scan operation has to be examined in all partitioned techniques by gradually inserting the data in all materialized views replication. In the graph given below x-axis shows different types of partitioning techniques and gradual massive increment of data is noted as first phase , second phase ,third phase and fourth phase respectively and y-axis shows the cpu cost in m-seconds.

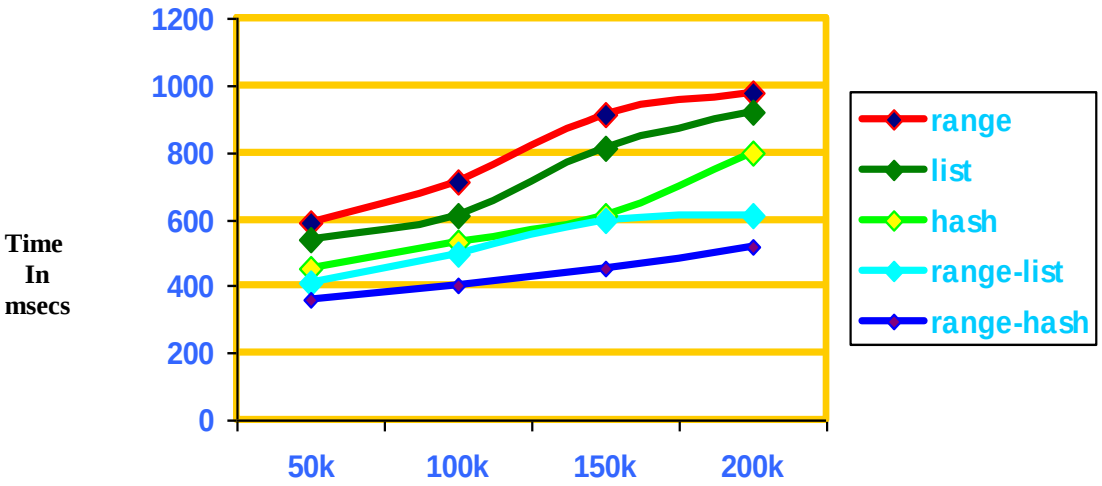

Figure 3.1 shows the response times of full scan operation.

The full scan operation graph shows the response times of different partitioning techniques when a full scan operation is applied. It has been observed that at first phase of full scan , range-list and range-hash performance is good when compared to range, list and hash partitioning techniques and when data is gradually growing , the performance of hash , range-list and range-hash are improved .Composite Range-hash is unchangeable even after massive insertion of data during second , third and fourth phases. In overall, the performance of Range-hash is better compared to other partitioning techniques when full scan operations are performed. Any full scan operations or partial scan operations with out using index objects performs well when underlying materialized views are partitioned with either Composite Range-List or Range-Hash partitioning techniques.

3.2 Full scan operation on Multi master replication

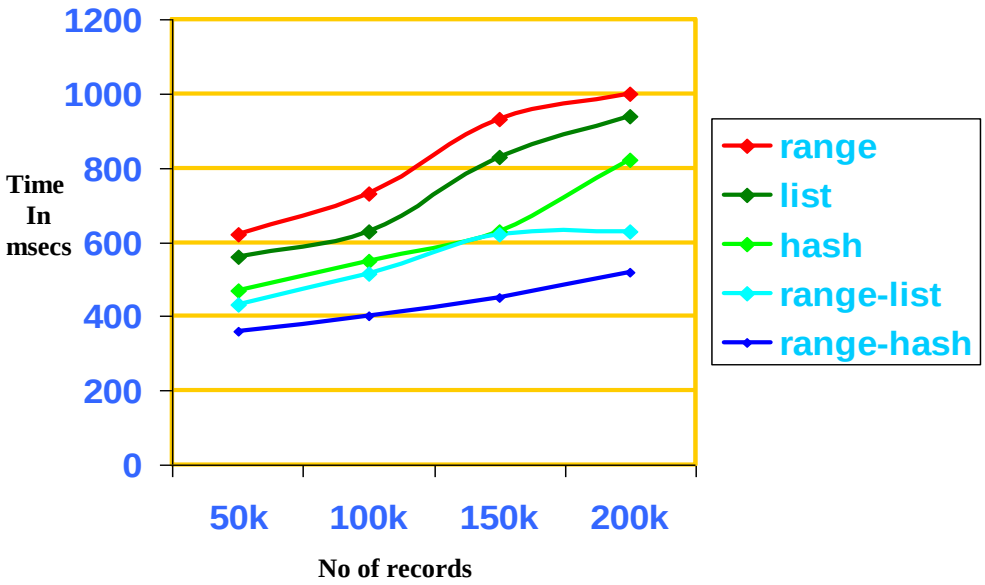

Figure 3.2 shows the response times of full scan operation.

The above graph shows the response times of different partitioning techniques when a full scan operation is applied. In overall, the performance of Range-hash is better compared to other partitioning techniques when full scan operations are performed. And range poor performance.

4. Conclusion

On the partitioned Databases using each of the algorithms materialized view replication strategy is applied and refreshment time is recorded. Similarly multimaster replication strategy is applied and refreshment times are recorded. From the results, it is observed that the performance of hash partitioning algorithm is poor when regular updates are performed. The composite range-hash partitioning and composite range-list partitioning algorithms showed better performance in both materialized view replication and multimaster replication strategies.

5. References

[1] Data base advanced replication and management A.P.I Reference

[2] Oracle Partitioning and QCO Space Management by Chuck Sodowsky, Dan Sherwin, Lenka Vanek and Thomas Szekely

[3] J. A. Blakeley, N. L. Martin. Join Index, Materialized View Replication, and Hybrid Hash-Join: A Performance Analysis Proc. IEEE Int'l. Con$ on Data Eng. Los Angeles,

[4] A. Gupta, I.S. Mumick, Maintenance of Materialized Views: Problems, Techniques, and Applications, IEEE Data Eng. Bulletin, Special Issue on Materialized Views**.**

[5] Materialized Views in Oracle (8.1.5/8i/9i/10g/11g) by Jangan pernah berhenti belajar

[6] Oracle Master Replication, Performance and Scalability, Metalink article

Author

A. PRAMOD KUMAR

received the Bachelor's degree from M.V.S.R Engineering college affliated to OSMANIA UNIVERSITY Hyderabad 2006. received the Masterr's degree from Chaitanya Bharathi Institute of Science and Technology affliated to OSMANIA UNIVERSITY Hyderabad 2009 Computer Science & Engineering department.. I'm currently Working as a Assistant professor ACE Engineering College in Information Technology Department affliated to JNTUH University.