RESEARCH ARTICLE OPEN ACCESS

Offsite One Way Data Replication towards Improving Data Refresh Performance

Nazia Azim, Yasir Qureshi, Fazlullah Khan, Muhammad Tahir
Syed Roohullah Jan, Abdul Majid
Department of Computer Science
Abdul Wali Khan University
Mardan- Pakistan

ABSTRACT

Concerned with the availability and access latency of data large organizations use replication to keep multiple copies of the data at concerned data centers that is easily accessible for their use. High bandwidth of network and uninterrupted power supply are essential for a normal and smooth replication process frequently used during the refresh process. During replication a scenario is occurred of an instance failure will jeopardize the process and efforts. Instance failure occurs due to low bandwidth of network or interrupted power supply. Versions compatibility for RDBMS is also problematic during the refresh process, in Pakistan chief and reliable replication tools are expensive enough to deploy. also incompatible with RDBMS and the existing replication environment that can replicate data from source to destination neglecting the need for online network connectivity and uninterrupted power supply and should be compatible with all versions of RDMS in a distributed database system. The really problems arises in many distributed database systems of almost all wide area establishments of Pakistan taking an example of any computerized department that needs to replicate its data to all servers on daily basis, using the already available techniques for replicating data they are unable to replicate data to remote districts of Pakistan like karak, sibbi etc. because there are concerned problem of continuous power supply and high rate of network bandwidth is still not available in that areas. At the other edge if considering the available techniques by oracle or else we can face the platform issues between different RDBMS because all of them can support the homogeneity not heterogeneity. In this paper we present a new offsite one way replication for distributed database using less bandwidth and also reduce the network overhead without compromising consistency and data integrity and to support heterogeneity. Compatible with all versions of RDMS. in offsite replication a CDF file is created which capture the altered data during transaction in client database. Through change data file only change will be forward to server from clients. The offsite data replication plays an importance role in the development of highly scalable and fault-tolerant database system.

Keywords: - CDF, RDBMS, Transaction Distributed, Offsite.

I. INTRODUCTION

Data replication is also called data consolidation in a distributed database environment, which is becoming a matured field in databases. Different types of replication methodologies [13, 16] depend on requirements and * available resources of organization. Majority of the organizations is using data replications [9] in a distributed database environment to populate their reporting servers for decision making support Replicated data can be more fault-tolerant than un-replicated data, and use to improve performance [8] by locating copies of the data near to their use. Copies of replicated data are held at a number of replicas that consist of storage and a process that maintains the data copy .A clients process can communicate the replicas to read or update the data. Traditionally both the clients and replicas reside on hosts, which are connected to internetwork consisting of local-area networks with gateways and point-to-point links through the Quorum protocols [29]. Oracle,[20] one of the leading RDBMS vendors, provides Oracle Advanced Replication which is capable of handling both synchronous and asynchronous replications[12]. It replicates tables and supporting objects, such as views, triggers, and indexes, to other locations.

Scalability, performance, and availability can be enhanced by replicating the database across the servers that works to provide access to the similar database. In heterogeneous database environment, the connectivity between master and slaves databases wants more network resources due to different frameworks.

In large-scale distributed system, Replication is a good technique for providing high availability [12] for data sharing across machine boundaries, because each

machine can own a local copy of the data. Optimistic data replication is an increasingly important technology. It allows the use of ATM banking with network failure and partitions and simultaneous cooperation access to shared data on laptops disconnected from Networks [2].

Replication is the key mechanism to achieve scalability and fault-tolerance in database [7].Data replication is a fascinating topic for both theory and practice. On theoretical side ,many strong results constraint what can be done in term of consistency, e g the impossibility of reaching consensus in asynchronous system, the blocking of 2PC ,the CAP theorem ,and the need for choosing a suitable correctness criterion among the many possible .On the practical side, data replication plays a key role in wide range of context: caching, back-up[23], high availability , increasing scalability ,parallel processing , etc. Finding a replication solution that is suitable in as many such contexts as possible remains an open challenge. The following figure show a simple replication flow.

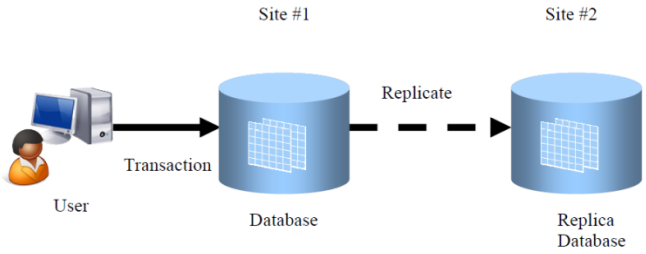


Figure 1.1: *simple replication flow*

II. DATA REPLICATION STRATEGIES

Oracle supports three types of replications.

2.1 MATERIZED VIEW REPLICATION

A materialized view is a replica of a target master from a single point in time [24]. These views are also known as snapshots. RDBMS uses materialized views to replicate data to non-master sites in a replication environment and to cache expensive queries in a data warehouse environment. The master can be either a master table at a master-site or a master materialized view at a materialized view-site. Where as in multi-master replication [13] tables are continuously updated by other master-sites, materialized views are updated from one or more masters through individual batch updates, known as a refreshes, from a single master site or master materialized view-site. Disadvantage of this type replication technique requires high speed network bandwidth [4]to refresh the data at master site after authentication. ***Figure 1.2*** illustrates materialized view flow.

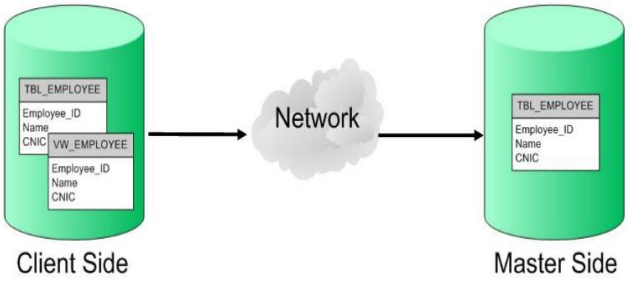


Figure 1.2: *Materialized View Replication*

2.2 MULTI-MASTER REPLICATION

Multi-master replication [10,13] is a kind of Synchronous replication [12] in which updates propagate immediately and ensures consistency. This is a procedural replication which runs same transaction at each site as a result of speed batch update. It can also be called 'peer-to-peer' or N-way replication [25], also this is equally participating in an 'update-anywhere' model. It is a method of database replication allows data to store by a group of computers and updated by any member of the group. All members are responsive to client data queries. The multi-master replication system is responsible for propagating the data modifications made by each member to the rest of the group, and resolving any conflicts that might arise between concurrent changes made by different members [6, 21]. Disadvantages include loosely consistent, i.e. lazy and asynchronous, violating ACID properties. Eager replication systems are complex and increase communication latency. Conflict resolution can become intractable as the number of nodes involved rises and latency increases. ***Figure 1.3*** describes Multi-Master replication process.

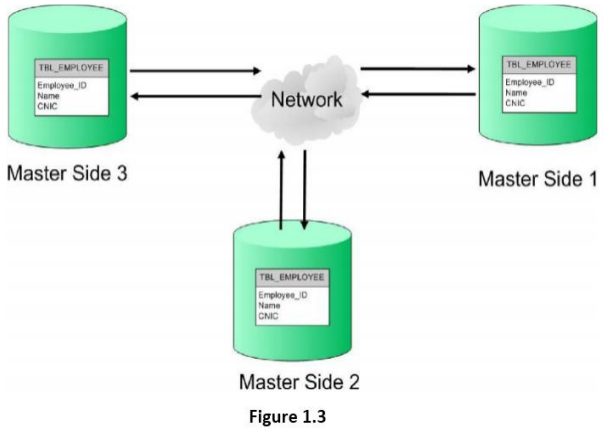


Figure 1.3: *Multi-master Replication*

2.3 STREAM REPLICATION:

Streams propagate both DML & DDL [1] changes to another (or elsewhere in same database). Streams are based on Log Miner which is an additional tool only available in enterprise editions of RDBMS. One can decide, via rules which changes are needed to be replicated and can optionally transform data before apply it at the destination. Log Miner continuously reads DML & DDL changes from redo logs. It converts those changes into logical change records (LCRs). There is at least 1 LCR per row changed. The LCR contains the actual changes, as well as the original data. There are basically three processes in stream replication in the first process the changes are captured into log files. In the second step which is background process the data is propagated to the destination site. And in the last process the data is read from log files and replicated into the destination database. Three processes in stream replication, First process changes is captured into log files. Second step which is background process the data is propagated to the destination site. And last process the data is read from log files and replicated into the destination database [20].For full replication technique see [16].

Streaming replication process.

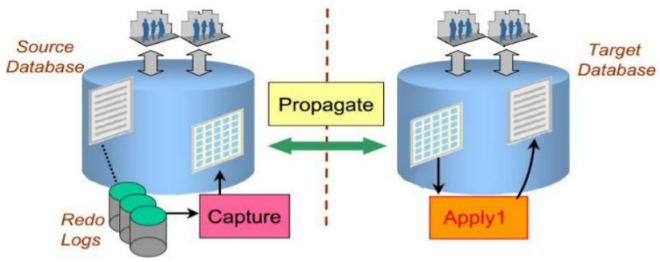


Figure 1.4

In the rest of this paper we present the problem statement in section 3 and section 4describethe proposed technique. Section 5 show the implementation of the discuss technique, in section 6 show the discussion on offsite replication, Section 7 related work in offsite replication, the conclusion is describe in section 8 and section 9 show the acknowledgment and last show the references.

III. PROBLEM STATEMENT

Many countries of the developing world are facing the electricity shortage. Pakistan is also a developing country and it is facing the acute shortage of electricity and low network bandwidth. High bandwidth of network and uninterrupted power supply are essential for normal and smooth replication process [4, 6]. Uninterrupted power supply can be provided by using high UPS available in the market however due to the 12 to 20 hours load shedding, any kind of high tech infrastructure cannot provide the required backup time for a smooth replication process. Due to the unscheduled load management plan; we conclude that throughout the year, master database servers and client database servers can hardly be interconnected with each other [1]. In traditional replications techniques, network connectivity or data link is established for transformation of data throughout the replication process which may remain active for hours. The offsite replication is the client/server replication and they can't require constant connectivity [28] to replicate data from source to destination. Offsite replication runs over two ways, replication over direct data path and replication over via WAN Accelerators. Direct path include production site and remote site. Uses the backup proxy which accesses the Veam backup server. Replication over WAN Accelerators uses in weak network link. Challenges like disasters recovery, backup consolidation and cast effectiveness for the solution we use offsite replication. In offsite replication we use the Change Data File which takes less bandwidth and no constant connectivity from source to destination database server. The system used the heterogeneity. The proposed one way replication model was implemented using standard DBMS tools (Oracle DBMS and PL SQL tools). For the Heterogeneous data replication the plate from must be different, i-e oracle, MY SQL, PLSQL, Oracle.

IV. PROPOSED TECHNIQUE

To develop a new unidirectional offsite replication technique for distributed databases environment this should transfer data using minimal bandwidth [21] without deactivating transactional operations neither at source database site nor at destination. The proposed technique not only reduces network overheads but also ensures effective and un-interruptible replication process without compromising on data consistency and integrity. Developed methodology can easily be adoptable, deployable and compatible with any RDBMS on any operating system platform [9] [27-33]. Our proposed system is composed of several algorithms to assure secrecy and preciseness with high availability and scalability of organizational data.

A local file is created which contain characteristics of Change Data File on the client side. Changed Data File (CDF) tracks and captures only altered data / row during the transactions in the client database [14]. The

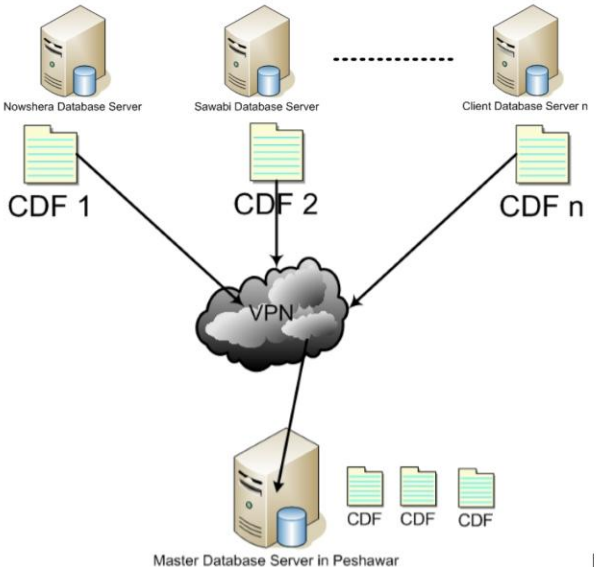
CDF is copied automatically at master site on network using Virtual Private Network (VPN) and by email services by consuming minimal bandwidth. The discuss network infrastructure can easily be configured throughout the country without any additional resources. During whole process the network resources were only consumed for small Period of time to transfer CDF file from client to master site. Offsite technique can improve data Refresh performance, because the system captures the DML operation on daily bases in a record [34-41].

4.1 CHANGED DATA FILE (CDF):

In traditional replication the whole record, data, or replication object is replicated from one site to another. In offsite replication CDF file was created on client site in a distributed database environment which captures all the changes including row insertion, update, deletion and as well as creation of new database objects. CDF contains DML operations [13]. The CDF provides incremental dump facility. Oracle the leading database vendors, does not truly supports the incremental backup utility especially in their express editions [23]. The commonly used EXP utility of Oracle's products gives incremental backup to a certain limitations which creates backup of the whole table instead of the changed data so increasing size of the dump file [23]. The incremental backup drops the existing table and re-creates the table for loading the complete data which time consuming but also create difficulties to identify the changes the data. Suppose we have a table contains two million rows with size 100 MB and a DML operation i-e single row is updated at the time of incremental backup the whole table will be exported instead of a single row which was updated. At the time of import in destination database same table will be deleted and fresh copy will be imported with a size of 100 MB. When change occurs in client side and replicate to master site through the Quorum-oriented Multicast protocols which helps in conflict detection [19] [42-48].

CDF supports huge data transformation from client to master site by dividing the larger CDF into smaller chunks which can be easily transferred using less bandwidth. The offsite system capture the change and import only change is occurring at the client side and export to the destination database/sever. The change data file capture only change and export to the main sever.

The Proposed offsite CDF data consolidation technique is used for one-way replication provided an effective data consolidation mechanism. Moreover the change data file can be successful in the country like Pakistan where high speed bandwidth and uninterrupted power supply is not continual for 24 / 7.



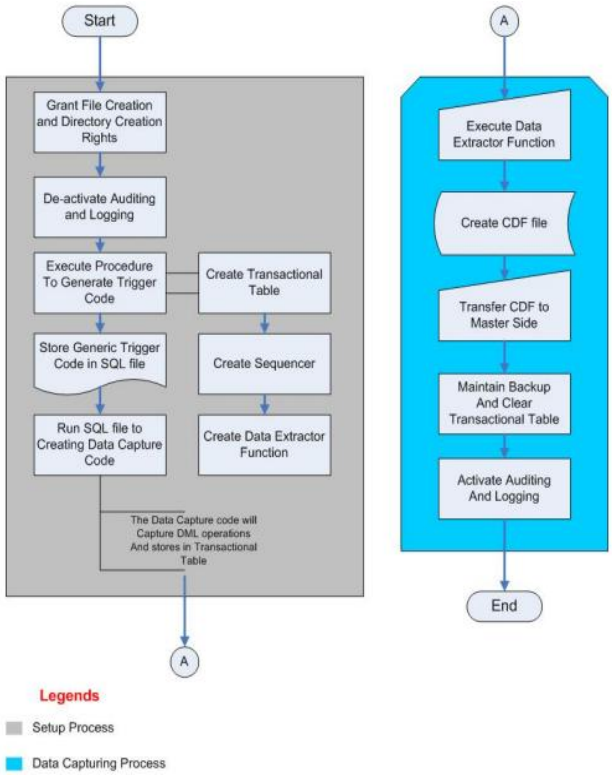
PROPOSED REPLICATION TECHNIQUE:

CDF is populated for every client side and capture the change during row insertion, deletion and Update. The change is replicate to destination through Virtual Private Network (VPN) and by email services by consuming minimal bandwidth [49-53].

V. IMPLEMENTATION MODEL

The proposed technique will implement in many ways by the requirements of the organization we have two databases EXPENSE and DATACENTER. EXPENSE is the source and DATACENTER the destination system or EXPENSE is the client and DATACENTER is the server. For the CDF population first the client side is activated for replication. As mention that CDF support heterogeneous RDBMS environment which provide a professional mechanism for migrating data in other RDBMS. Through CDF technique master database and web server are up to date. Which provided the data refresh performance.CDF methodology is compose of two different packages for master and client side servers. The client side package is installed on client side. The master side is installed on the master side server the client side package is further divided into two more processes, the setup process and the execution process. The client side package can be elaborated briefly in ***figure 1.6*** the gray shaded block represents the setup process executed once for the all table. The data capturing process is presented in sky blue shade is executed on client side when data replication required.

Figure 1.6showClient side process



5.1 CLIENT SIDE PACKAGE (SETUP PROCESS):

Directory and privileges are granted for the EXPENSE and DATACENTER to create the CDF file in a specific directory. The scot/tiger is connected to oracle database. Then EXPENSE is connected in scot/tiger schema in oracle database. Next the file is import and connects to EXPENSE. They import the export file and same process for DATACENTER to export the import file. like

CONN SCOTT/TIGER

GRANT DBA TO EXPENSE IDENTIFIED BY EXPENSE;

CONN EXPENSE/EXPENSE

CREATE OR REPLACE DIRECTORY DIR AS 'D:\EXPENSE\THESIS_CODING;

The above coding called the client side setup which creates the directories for the Change Data File to save. The GENERATE_TRIGGER_SCRIPT is executed to create generic code of transactional triggers for all tables in the schema of a particular table. The procedure reads data from data dictionary to generate code of transactional triggers which will capture the DML transactions. The generated transactional triggers code is stored in a SQL extension file named DYNAMIC_CODE.SQL. The code like this CREATE OR REPLACE PROCEDURE GENERATE_TRIGGER_SCRIPT.

TBL_TRANSACTION table is created for storing captured transactions along with the serial no. The serial no used to keep a track of all the transactions. A sequencer is created with a name as SEQ_FOR_TRANSACTION .like

CREATE SEQUENCE SEQ_FOR_REPLICATION START WITH 1 INCREMENT BY 1 NOCYCLE NOCACHE This sequencer provides serial numbers to every DML transaction. A function POPULATE_CDF is created to read data from TBL_TRANSACTION table. Due to this CDF file is created and populate, also called data extractor function.

CREATE OR REPLACE TRIGGER TRG_REP_TBL_ASSET
AFTER INSERT OR UPDATE OR DELETE
ON TBL_ASSET
FOR EACH ROW
IF INSERTING THEN

5.2 CLIENT SIDE (DATA CAPTURING PROCESS):

When replication required Data capturing process execute to populate CDF file and transfer to master side. Execute the function named as POPULATE_CDF. This creates and populates CDF with the transactional data stored in the TBL_TRANSACTION table. Transfer CDF file to the master side either via VPN or email depending upon the file size. Create a backup of transactional data stored in the TBL_TRANSACTION table and truncate the table.

5.3 SERVER SIDE (LOADING PROCESS)

At master side CDF file receive data extraction process started automatically after execution of the batch file. The batch file executed manually or operating system scheduler depending upon the requirement [5, 6]. The batch file find the newly available CDF file, the data extraction process is combination of multiple steps. De-activate default auditing and logging at master side performance. Receive CDF file either from VPN or email. Place the CDF file in the specific directory as following master side!

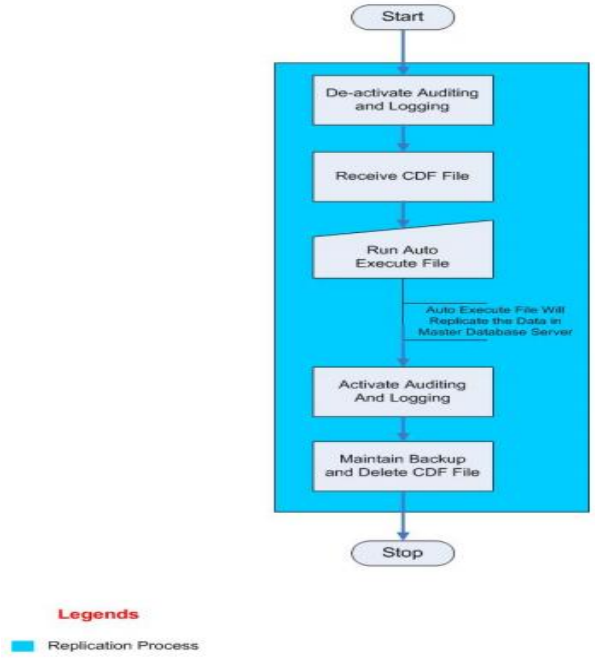


figure 1.7 depicts the entire workflow at master side:

VI. DISCUSSIONS

Replication is widely used to achieve various goals in information systems, such as better performance, fault tolerance, backup, etc. Benefits of offsite Replication like 24x7x365 availability of mission-critical data, Data integrity between source and target databases, disaster recovery and backup.

The widening demand supply gap has resulted in regular load shedding of eight to ten hours in urban areas and eighteen to twenty hours in rural areas. So the clients and server database cannot connected with each other Data replication is the most matured field in database world; however there are still some issues of network which is a vital part of replication process. The purpose of this study is to solve the network and load shading problem using offsite CDF replication. Through the CDF data refresh performance because data is up to date.

Suppose table in EXPENSE client with 11,15,193records and the size of full database dump file is 147 MB. At initial full database dump file is imported in the master side (DATACENTER) server for both the traditional and CDF methodologies. After a week 5831 transactions are performed on EXPENSE database. When the replication is required some operation are performed: Take full database dump of EXPENSE which file size=149 MB time taken 4 min. Compress Dump file size = 22.39 MB Time Taken: 30 sec .Upload and email compressed dump file Time required 15 min Downloading compressed dump file time 7 mints Un-compress the dump file Time Taken: 30 sec Import dump file into dummy database time taken 4 mint. After that delete all record from all tables of master database (DATACENTER) Time Taken: 6 min. Select all records from dummy database and load them into master database for all tables. Time Taken 6 min Delete dummy database. For replicating data of EXPENSE into master database an average time required is approx. 45 min. The average time consumed for replicating EXPENSE data is calculated using following formula:

Time (T) = Export Time (TE) + Upload Time (TU) + Download Time (TD) + Full Import Time (TI) + Shifting Data Time (TS).

After implemented CDF technique EXPENSE data is replicated into master database in approx. 5 min for 5831 transactions we have Execute POPULATE_CDF function. Time Taken: 1 min CDF file size = 0.71 MB after that Compress CDF file of size 0.0072MB Time Taken: 10 sec. Now transfer created CDF file via VPN or email to the master database server Time Taken: 1 min. Download and place CDF file in pre-defined directory on master database Time Taken: 1 min. Now Un-compress CDF file Time Taken: 10 sec. we should Double click on the ORACLE.BAT batch file Time Taken: 1.30 min. After execution batch file the data was loaded into master database. The average time consumed for replicating data using CDF technique is calculated as:

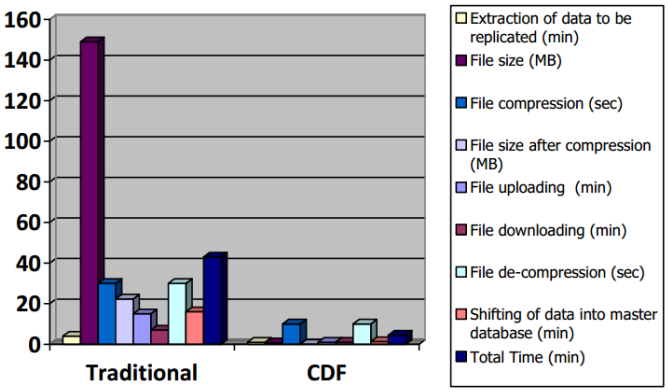
Time (T) = Creation CDF file (TC) + Upload Time (TU) + Download Time (TD) + Extraction of CDF (TE).

Following table 1.1 elaborates the comparison between traditional and CDF methodologies for 5831 transactions.

TABLE 1.1

S #	Description	Traditional Approach	CDF Approach
1	Extraction of data to be replicated	4 min	1 min
2	File size	149 MB	.71 MB
3	File compression	30 sec	10 sec
4	File size after compression	22.39 MB	.0071 MB
5	File uploading	15 min	1 min
6	File downloading	7 min	1 min
7	File de-compression	30 sec	10 sec
8	Shifting of data into master database	16 min	1.30 min
	Total time	**43 min**	**4.5 min**

following ***figure 1.8***depicts the table 1.1 more evidently:



INCREMENTAL BACKUP USING CDF:

CDF provide a true incremental backup. In traditional mechanism for incremental backup [23], the whole table is exported no matter how many rows are altered. A table TBL_SALES contains 100,000 number of records with size of 100 MB. If single record in a TBL_SALES is affected with a DML transaction then at the time of incremental backup the whole table will be exported instead of the changed record. On the CDF methodology, incremental backup, exports only the changed data instead of the whole table.

Table 1.2 *CDF incremental backup vs. Traditional backup*

Table Name	No of Rows	Table Size (MB)	Changed Rows	Traditional Incremental Backup File Size (MB)	CDF Incremental Backup File Size (MB)
CNG	5125	6	100	6	.0119
FAST FOOD	6234	7	100	7	.0119
PAY	13000	13	100	13	.0119
Tax	1200	12	100	12	.0119

CDF incremental backup vs Traditional backup.

The above table identified that the size of incremental backup the CDF and traditional technique.

SUPPORT FOR HETEROGENEOUS RDBMS:

The CDF methodology provides the facility of replication in heterogeneous RDBMS environment. Traditional and conventional technique support homogenous RDBMS. CDF is flexible used for any RDBMS with DML transactions. CDF file was used to replicate data in Oracle database and MySQL database [13]. The CDF is compatible with all the versions of Oracle and MySQL.

The platforms used for CDF methodology of oracle and MySQL are as follow.

Table 1.3

Oracle RDBMS	MySQL RDBMS
Database 9i	MySQL 4.00
Database 10g XE	MySQL 4.90
Database 10g EE	MySQL 5
Database 11g XE	MySQL 5.02
Database 11g EE	

XE stands for Express Edition

EE stands for Enterprise Edition.

VI. REVIEW OF LITERATURE

Much research has been done on replicated data little of it relates directly to mobile computing. Replication for database systems. We surveyed the relevant work in replication for databases and remote distributed databases.

Chanchary and Islam [9]. The system composed of several algorithms to assure secrecy and preciseness with availability of organizational data. The scenario give the idea of a large organization composed of offices having own premises data centers in different locations. All data centers dedicated for organization managed by a central server, provided a platform for different query formats such as SQL or NoSQL and different database like MySQL, SQL Server and NoSQL. Thomson[1] system called Calvin designed a scalable transactional layer, all storage system implement a basic CRUD interface (create / insert, read, update, and delete). It is possible to run Calvin on top

distributed non-transactional storage systems such as Simple DB. Calvin assuming that the storage system is not distributed out of box. Example the storage system could be a single-node key-value store that is installed on multiple nodes. The Calvin has three separate layers; sequencing layer, scheduling layer and storage layer. Functionalities partitioned across a cluster. McElroy & Pratt [20].Oracle Database contains various styles from a master/slave replication where updates must be applied at master, a peer-style replication [22] where updates performed at one replica are forward other replicas. Sybase's SQL Remote product [27] supports optimistic client/server replication for remote database. It allow remote computer to selectively replicate a part of database. Mobile client maintains log of updates, which shipped to server using email when required. Liao [12] focuses issues of data synchronization and resynchronization in case of architecture failures. Difference between synchronous and asynchronous replication. The model divided the replication process in five phases, request forwarding, lock coordination, execution, commit coordination and client response. Heidemann [26]. Much of the research on replication for database systems with mobile computing. "Consistency in a Partitioned Network: A Survey", contain a good survey on optimistic replication [11] A number of optimistically replicated file systems include Ficus, Coda designed to support mobility. Reconcile, Rumor provided useful lessons for development offsite data replication using less bandwidth methodology [55-63], including insights the costs and benefits of different methods of detecting updates, handling conflicting data, and data structures of update. Bayou [6] described a replicated storage system capable of supporting both file-oriented and database-oriented operations. Bayou takes application-aware approach to optimistic peer replication Li [23] highlighted the limitation of incremental and cumulative backup techniques in available RDBMS. Some latest work can be find in [64-73]

XIII. CONCLUSION

RDBMS vendors are enticing / facilitating their customers with advance and easy to configure replication utilities in enterprises versions at an additional cost. Customers identify their requirements and purchase these utilities with respect to their needs after detailed analysis at their own. These add-ons utilities can easily be adopted and implemented in Western countries where network communication is available consistently without any disruption and likewise the power supply is also unswerving. The dilemma of this country is that the high cost IT related development schemes are started ambitiously without any requirement assessment. To further worsen the situation, it is a common practice in administration that IT equipment and software are procured before recruitment of competent IT personnel due to the lapse of budget in every fiscal year. These purchases are made without any prior need- based assessment. In due course, at the time of software application deployment throughout the country or province, the purchased RDBMS does not support or have adequate utilities to replicate the data from other remote servers into the centralized server usually installed in Provincial capitals and hence resulting in the busting of the complete project.

Developed CDF methodology facilitate one-way replication / consolidation technique which is quite helpful in the scenario mentioned above and can save the entire project without any additional cost and not jeopardizing the development scheme. This technique can save not only human resources and extra efforts but can also be helpful to overcome financial constraints of the project.

Developed CDF technique also provides incremental backup facility which is capable of exporting on the changed data instead of entire table. Above all the developed technique is applicable in heterogeneous database environment which is a key feature of CDF.

IX. RECOMMENDATIONS

After complete implementation and evaluation of developed CDF methodology, following deficiencies are identified and can be enhanced in future work:

1. A GUI interface is required to handle all the setup activities for deploying CDF methodology on client and master side database servers. This will expedite the setup process as well as execution processes.

2. The developed methodology at this instance only supports one way data replication. It can be extended to two-way or n-way data replication types.

3. The CDF file can only capture DML operations. In case of DDL or DCL operation the CDF approach cannot be feasible at all.

4. The critical issue of CDF file is that it contains textual data which can be altered by other sources. The textual data needs to be encrypted by using public / private key cryptography.

REFERENCES

[1] Alexander Thomson; Thaddeus Diamond; Shu- Chun Weng; Kun Ren; Philip Shao; Daniel J.Abadi; Calvin, "*Fast Distributed Transactions for Partitioned Database Systems*" ,SIGMOD '12, pp.1-12, 2012.

[2] A. Wang; P.L. Reiher; R. Bagrodia, "*A simulation evaluation of optimistic replicated filling in mobile environments*" Proceedings of International Performance, Computing and Communications Scottsdale, AZ, USA, pp. 43-51, 1999.

[3] Bettina Kemme; Gustavo Alonso, "*Don't Database Replication*", Proceeding of 26th VLDB Conference, Cairo, Egypt, 2000.

[4] Khan. F., Bashir, F. (2012). *Dual Head Clustering Scheme in Wireless Sensor Networks*. in the IEEE International Conference on Emerging Technologies (pp. 1-8). Islamabad: IEEE Islamabad.

[5] Syed Roohullah Jan, Khan. F., Zaman. A., *The Perception of students about Mobile Learning at University Level*. in the khgresearch.org Turkey

[6] Khan. F., Nakagawa. K. (2012). *Cooperative Spectrum Sensing Techniques in Cognitive Radio Networks*. in the Institute of Electronics, Information and Communication Engineers (IEICE), Japan , Vol -1, 2.

[7] M. A. Jan, P. Nanda, X. He, Z. Tan and R. P. Liu, "*A robust authentication scheme for observing resources in the internet of things environment*" in 13th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 205-211, 2014, IEEE.

[8] Bipin Pokharel, "*Power Shortage, its impacts and the Hydropower Sustainability*", NRSC 616 Project Paper, 2010.

[9] B. Walker; G. Popek; R. English; C. Kline; G. *Thiel, "The LOCUS distributed operating system"*

a. *Operating Systems Review*, vol.17, (no.5, spec. issue) pp.49-70, 1983.

[10] Khan. F., Nakagawa, K. (2012). *Performance Improvement in Cognitive Radio Sensor Networks*. in the Institute of Electronics, Information and Communication Engineers (IEICE) , 8.

[11] M. A. Jan, P. Nanda and X. He, "*Energy Evaluation Model for an Improved Centralized Clustering Hierarchical Algorithm in WSN*," in Wired/Wireless Internet Communication, Lecture Notes in Computer Science, pp. 154–167, Springer, Berlin, Germany, 2013.

[12] Khan. F., Kamal, S. A. (2013). *Fairness Improvement in long-chain Multi-hop Wireless Adhoc Networks*. International Conference on Connected Vehicles & Expo (pp. 1-8). Las Vegas: IEEE Las Vegas, USA.

[13] M. A. Jan, P. Nanda, X. He and R. P. Liu, "*Enhancing lifetime and quality of data in cluster-based hierarchical routing protocol for wireless sensor network*", 2013 IEEE International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (HPCC & EUC), pp. 1400-1407, 2013.

[14] Jabeen. Q., Khan. F., Khan, Shahzad, Jan. M. A. (2016). *Performance Improvement in Multihop Wireless Mobile Adhoc Networks*. in the Journal Applied, Environmental, and Biological Sciences (JAEBS), Print ISSN: 2090-4274 Online ISSN: 2090-4215

[15] Khan. F., Nakagawa, K. (2013). *Comparative Study of Spectrum Sensing Techniques in Cognitive Radio Networks*. in IEEE World Congress on Communication and Information Technologies (p. 8). Tunisia: IEEE Tunisia.

[16] Khan. F., (2014). *Secure Communication and Routing Architecture in Wireless Sensor Networks*. the 3rd Global Conference on Consumer Electronics (GCCE) (p. 4). Tokyo, Japan: IEEE Tokyo.

[17] M. A. Jan, P. Nanda, X. He and R. P. Liu, "*PASCCC: Priority-based application-specific congestion control clustering protocol*" Computer Networks, Vol. 74, PP-92-102, 2014.

[18] Khan. F., (2014). *Fairness and throughput improvement in mobile ad hoc networks*. in the 27th Annual Canadian Conference on Electrical and Computer Engineering (p. 6). Toronto, Canada: IEEE Toronto.

[19] Mian Ahmad Jan and Muhammad Khan, "*A Survey of Cluster-based Hierarchical Routing Protocols*", in IRACST–International Journal of Computer Networks and Wireless Communications (IJCNWC), Vol.3, April. 2013, pp.138-143.

[20] D.B. Terry; M.M. Theimer; K. Petersen; A.J. Demers; M.J. Spreitzer; C.H. Hauser, "*Managing update conflicts in Bayou, a weekly connected replicated storage system*,"

Operating Systems Review, Vol. 29, (no.5), pp.172-83, Dec. 1995.

[21] Bettina Kemme; Gustavo Alonso "*Database Replication: a Tale of Research across Communities"* VLDB Endowment, Vol. 3, No. 1, 2010.

[22] David Ratner; Peter Reiher; Gerald J. Popek "*ROAM: A Scalable Replication System for Mobility" Computer Science Department*, UCLA, Los Angeles, CA 90095.

[23] Khan. S., Khan. F., (2015). *Delay and Throughput Improvement in Wireless Sensor and Actor Networks*. 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW) (pp. 1-8). Riyadh: IEEE Riyad Chapter.

[24] Khan. Shahzad, Khan. F., Jabeen. Q., Arif F., Jan. M. A. (2016). *Performance Improvement in Wireless Sensor and Actor Network, in* the Journal Applied, Environmental, and Biological Sciences Print ISSN: 2090-4274 Online ISSN: 2090-421.

[25] Mian Ahmad Jan and Muhammad Khan, "*Denial of Service Attacks and Their Countermeasures in WSN*", in IRACST– International Journal of Computer Networks and Wireless Communications (IJCNWC), Vol.3, April. 2013.

[26] Jabeen. Q., Khan. F., Muhammad Nouman Hayat, Haroon Khan, Syed Roohullah Jan, Farman Ullah, " *A Survey : Embedded Systems Supporting By Different Operating Systems"*, International Journal of Scientific Research in Science, Engineering and Technology(IJSRSET), Print ISSN : 2395-1990, Online ISSN : 2394-4099, Volume 2 Issue 2, pp.664-673, March-April 2016. URL : http://ijsrset.com/IJSRSET1622208.php

[27] Farah Habib Chan chary; Saimul Islam, "*Data Migration: Connecting Databases in the Cloud*",

a. ICCIT, pp. 450-55, 2012.

[28] Filip; I. Vasar; R. Robu, "*Considerations about an Oracle Database Multi-Master Replication*", Applied Computational Intelligence and Informatics, SACI '09, pp. 147-152, 2009.

[29] G.H. Kuenning; R. Bagrodia; R.G. Guy; G.J. Popek; P. Reiher; A. Wang "*Measuring the quality*

a. *of service of optimistic replication,*" Proceedings, Object-Oriented Technology. ECOOP'98 Workshop Reader, Brussels, Belgium, pp.319-20, 1998.

[30] M. A. Jan, P. Nanda, X. He and R. P. Liu, *"A Sybil Attack Detection Scheme for a Centralized Clustering-based Hierarchical Network*" in Trustcom/BigDataSE/ISPA, Vol.1, PP-318-325, 2015, IEEE.

[31] Jabeen. Q., Khan. F., Hayat, M.N., Khan, H., Syed Roohullah Jan, Ullah, F., (2016) *A Survey : Embedded Systems Supporting By Different Operating Systems* in the International Journal of Scientific Research in Science, Engineering and Technology(IJSRSET), Print ISSN : 2395-1990, Online ISSN : 2394-4099, Volume 2 Issue 2, pp.664-673.

[32] Syed Roohullah Jan, Syed Tauhid Ullah Shah, Zia Ullah Johar, Yasin Shah, Khan. F., "*An Innovative Approach to Investigate Various Software Testing Techniques and Strategies"*, International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Print ISSN : 2395-1990, Online ISSN : 2394-4099, Volume 2 Issue 2, pp.682-689, March-April 2016. URL : http://ijsrset.com/IJSRSET1622210.php

[33] Khan. F., Ullah. F., Jabeen. Q., Syed Roohullah Jan, Khan. S., (2016) *Applications, Limitations, and Improvements in Visible Light Communication Systems* in the VAWKUM Transaction on Computer Science Vol. 9, Iss.2,
DOI: http://dx.doi.org/10.21015/vtcs.v9i2.398

[34] Guoqiong Liao, "*Data Synchronization and Resynchronization for Heterogeneous Databases Replication in Middleware–based Architecture*", Journal of Networks, pp. 210-17, 2012.

[35] Hakik Paci; Elinda Kajo; Igli Tafa; Aleksander Xhuvani, "*Adding a new site in an existing oracle multi-master replication without Quiescing the Replication*", International Journal of Database Management Systems (IJDMS), Vol.3, No.3, August 2011.

[36] William D. Norcott; john Galanes "*Method and apparatus for Chang data capture in database system*" USA Feb.14, 2006.

[37] J. J. Kistler and M. Satyanarayanan, "*Disconnected operation in the Coda File System,*" ACM Transactions on Computer Systems, vol.10, (no.1), pp.3-25, Feb 1992.

[38] Matthias Wiesmann; Fernando Pedone; Andr´e Schiper; Bettina Kemme, "*Database replication Techniques: a Three Parameter Classification"* in 19th IEEE Symposium on Reliable Distributed System, pp. 206-215, 2001.

[39] M. J. Carey and M. Livny, "*Conflict detection tradeoffs for replicated data*," ACM Transactions on Database Systems, vol.16, (no.4), p.703-46, Dec. 1991.

[40] M. J. Fischer and A. Michael. "*Sacrificing Serializability To Attain High Availability of Data In an Unreliable Network*," Proceedings of the 4th Symposium on Principles of Database Systems, 1982.

[41] Richard A; Golding; Darrell D; E. Long" *Quorum-oriented Multicast protocols for Data Replication*" Santa Cruz, CA 95064 IEEE 1992.

[42] Patricia McElroy; Maria Pratt; "*Oracle Database 11g: Oracle Stream Replication*" in Oracle White paper 2007.

[43] P. Reiher; J. Heidemann; D. Ratner; G. Skinner; G. Popek, "*Resolving file conflicts in the Ficusfile system*", Proceedings of the Summer 1994 USENIX Conference, Boston, MA, USA, pp.183-95, 1994.

[44] P. Reiher; G. Popek; M. Gunter; J. Salomone; D. Ratner. "*Peer-to-Peer Reconciliation Based Replication for Mobile Computers*," Proceedings of the ECOOP Workshop on Mobility and Replication, July 1996.

[45] Qun Li; Honglin Xu, *"Research on the Backup Mechanism of Oracle Database*", International Conference on Environmental Science and Information Application Technology, ESIAT'09, pp. 423-426, 2009.

[46] M. A. Jan, "*Energy-efficient routing and secure communication in wireless sensor networks*," Ph.D. dissertation, 2016.

[47] Syed Roohullah Jan, Faheem Dad, Nouman Amin, Abdul Hameed, Syed Saad Ali Shah, " *Issues In Global Software Development (Communication, Coordination and Trust) - A Critical Review*", International Journal of Scientific Research in Science, Engineering and Technology(IJSRSET), Print ISSN : 2395-1990, Online ISSN : 2394-4099, Volume 2 Issue 2, pp.660-663, March-April 2016. URL : http://ijsrset.com/IJSRSET1622207.php

[48] Randall G. Bello; Karl Dias; Alan Downing; James Feenan; Jim Finnerty; William D. Norcott; Harry Sun; Andrew Witkowski; Mohamed Ziauddin, "*Materialized Views in Oracle*", Proceedings of the 24th VLDB Conference New York, USA, 2008.

[49] R. Guy; P. Reiher; D. Ratner; M. Gunter; W. Ma; G. Popek, "*Rumor: mobile data access through optimistic peer-to-peer replication*," Proceedings, Advances in Database Technologies. pp. 254-65, 1999

[50] R. G. Guy; J.S. Heidemann; W. Mak; T.W.Page; G.J. Popek; D. Rothmeier, "*Implementation of the Ficus replicated file system*," In USENIX Conference Proceedings, pp. 63-71, June 1990.

[51] S. B. Davidson; H. Garcia-Molina; D. Skeen, "*Consistency in partitioned networks*," Computing Surveys, Vol.17, (no.3), p.341-70, Sept 1986.

[52] Sybase Inc, "*Sybase Replication Technical*", Technical White Paper, 2012.

[53] T.Ekenstam; C. Matheny; P.Reiher; G.J. Popek, "*The Bengal Database Replication System*",

a. Journal Distributed and Parallel Databases, pp. 187-210, May 2001.

[54] Azim. N., Majid. A., Khan. F., Tahir. M., Safdar. M., Jabeen. Q., (2016*) Routing of Mobile Hosts in Adhoc Networks.* in the International Journal of Emerging Technology in Computer Science and Electronics (in press)

[55] Azim. N., Qureshi. Y., Khan. F., Tahir. M., Jan. S.R, Majid. A., (2016) *Offsite One Way Data Replication towards Improving Data Refresh Performance.* in the International Journal of Computer Science and Telecommunications (in press)

[56] Azim. N., Majid. A., Khan. F., Tahir. M., Syed Roohullah Jan, (2016) *People Factors in Agile Software Development and Project Management*. in the International Journal of Emerging Technology in Computer Science and Electronics (in press)

[57] Azim. N., Khan. A., Khan. F., Syed Roohullah Jan, Tahir. M., Majid. A. (2016*) Offsite 2-way Data Replication towards Improving Data Refresh Performance.* in the International Journal of Engineering Technology and Applications (in press)

[58] Azim. N., Ahmad. I., Khan. F., Syed Roohullah Jan, Tahir. M., Majid. A. (2016*) A New Robust Video Watermarking Technique Using H.264/AAC Codec Luma Components Based On DCT*. in the International Journal of Advance Research and Innovative Ideas in Education (in press)

[59] Syed Roohullah Jan, Ullah. F., Khan. F., Azim. N, Tahir. M. (2016) *Using CoAP protocol for Resource Observation in IoT*. in the International Journal of Engineering Technology and Applications (in press)

[60] Syed Roohullah Jan, Ullah. F., Khan. F., Azim. N, Tahir. M,Safdar, Shahzad. (2016) *Applications and Challenges Faced by Internet of Things*- A Survey. in the International Journal of Emerging Technology in Computer Science and Electronics (in press)

[61] Tahir. M., Syed Roohullah Jan, Khan. F., Jabeen. Q., Azim. N., Ullah. F., (2016) *EEC: Evaluation of Energy Consumption in Wireless Sensor Networks*. in the International Journal of Engineering Technology and Applications (in press)

[62] Tahir. M., Syed Roohullah Jan, Azim. N., Khan. F., Khan. I. A., (2016) *Recommender System on Structured Data*. in the International Journal of Advance Research and Innovative Ideas in Education (in press)

[63] Tahir. M., Khan. F., Syed Roohullah Jan, Khan. I. A. and Azim, N., (2016*) Inter-Relationship between Energy Efficient Routing and Secure Communication in WSN*. in the International Journal of Emerging Technology in Computer Science and Electronics (in press)

[64] Safdar. M., Khan. I. A., Khan. F., Syed Roohullah Jan, Ullah. F., (2016) *Comparative study of routing protocols in Mobile adhoc networks*. in the International Journal of Computer Science and Telecommunications (in press)

[65] M. A. Jan, P. Nanda, X. He and R. P. Liu. 2016. A Lightweight Mutual Authentication Scheme for IoT Objects, IEEE Transactions on Dependable and Secure Computing (TDSC), "Submitted".

[66] M. A. Jan, P. Nanda, X. He and R. P. Liu. 2016. *A Sybil Attack Detection Scheme for a Forest Wildfire Monitoring Application,* Elsevier Future Generation Computer Systems (FGCS), "Submitted".

[67] Puthal, D., Nepal, S., Ranjan, R., & Chen, J. (2015, August). *DPBSV--An Efficient and Secure Scheme for Big Sensing Data Stream.* InTrustcom/BigDataSE/ISPA, 2015 IEEE (Vol. 1, pp. 246-253). IEEE.

[68] Puthal, D., Nepal, S., Ranjan, R., & Chen, J. (2015). *A Dynamic Key Length Based Approach for Real-Time Security Verification of Big Sensing Data Stream.* In Web Information Systems Engineering–WISE 2015 (pp. 93-108). Springer International Publishing.

[69] Puthal, D., Nepal, S., Ranjan, R., & Chen, J. (2016). *A dynamic prime number based efficient security mechanism for big sensing data streams* .Journal of Computer and System Sciences.

[70] M. A. Jan, P. Nanda, M. Usman and X. He. 2016. *PAWN: A Payload-based mutual Authentication scheme for Wireless Sensor Networks*, in 15th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-16), "accepted".

[71] M. Usman, M. A. Jan and X. He. 2016. *Cryptography-based Secure Data Storage and Sharing Using HEVC and Public Clouds*, Elsevier Information sciences, "accepted".

[72] Puthal, D., & Sahoo, B. (2012). *Secure Data Collection & Critical Data Transmission in Mobile Sink WSN: Secure and Energy efficient data collection technique*.

[73] Puthal, D., Sahoo, B., & Sahoo, B. P. S. (2012). *Effective Machine to Machine Communications in Smart Grid Networks*. ARPN J. Syst. Softw.© 2009-2011 AJSS Journal, 2(1), 18-22.