Modern Management Science & Engineering
ISSN 2052-2576 (Online)
Vol. 5, No. 2, 2017
www.scholink.org/ojs/index.php/mmse

Framework for Client-Server Distributed Database System for an Internal Revenue System "a Case Study of Bayelsa State"

Sese Tuperekiye E.[1*], Zuokemefa Enebraye P.[2], & Yerikema Paul jnr[3]

[1] Department of Computer Science, Bayelsa State College of Arts and Science, Bayelsa State, Nigeria

[2] Department of Computer Science, Isaac Jasper Boro College of Education, Sagbama, Bayelsa State, Nigeria

[3] Department of Computer Science, Bayelsa State College of Arts and Science, Yenagoa Bayelsa State

[*] SeseTuperekiye E., E-mail: tupsese@yahoo.com

Abstract

In recent years, the Bayelsa State government has encountered a lot of problems in trying to monitor and regulate the taxes collected at the local government area and this is due to the fact that there are no proper record keeping system and no linkage between the state and the local government. There are no accuracy and precision based on the employee data kept by the local government of the state. This result to the inability of the government to monitor an employee's tax records from the day of resumption of duty to the day of retirement. Some of the taxes generated at the local government areas are not remitted to the state thereby short changing the state and reducing its Internally Generated Revenue (IGR). Thus, there is need to integrate the operational data of an organization and provide controlled access to the data. The aim of this study is to create a distributed Internal Revenue system for different local government areas in Bayelsa State. The system consists of a relational database of internal revenue variables which could be shared by the various Local Government Areas of Bayelsa State. Each of the local government area will form a website, and the database will be hosted by the server at the Board of Internal Revenue Services (BIRS) Bayelsa State. All LGAs will access the database via a distributed network. The client/server distributed network architecture is used in the design and implementation of the system. The system is capable of monitoring an employee's tax records from the day of resumption of duty to the day of retirement, generation of reports concerning an employee or sets of employee's tax records, and also access information from the local government at all times.

Keywords

client, sever, distributed database, revenue, tax, network, framework, Local Government Area (LGA)

1. Introduction

Bayelsa State is located at the south-south region of Nigeria and can also be said to be an oil producing state. The state has eight local government areas and these local government areas generate their own revenue and pay their workers through the federal allocation to the LGA and in the process certain deductions are made in the form of tax, etc. It is of great concern to note that as these LGA's carryout their Internally Generated Revenue (IGR) through various taxes there is no proper remittance of these taxes collected by the LGA's to the Board of Internal Revenue (BIR) of the state and also lack of proper monitoring of revenue accruable to the state.

Based on these and other factors the researcher has considered using a client/server distributed database system to implement the Internal Revenue System of the state with a view to link all LGA's to the headquarters which is located in the state capital for easy access to information.

The major motivation behind the development of database system is the desire to integrate the operational data of an organization and to provide controlled access to the data. Although integration and controlled access may imply centralization, this was not the intention. In fact, the development of computer networks promotes a decentralized mode of work. The decentralized approach makes the structure of many companies and organizations which are logically distributed into divisions, departments, projects and so on, and physically distributed into offices, plants, factories were each unit maintains its own operational data.

A distributed computer system consist of a collection of autonomous computers linked by a computer network, and equipped with the necessary computer networking software's such as servers and browsers to effect its communication with other computers (www.answers.com). Navathe (2000) Defined Distributed Database (DDB) as a collection of multiple logically integrated databases distributed over a computer network. In addition, he defined a Distributed Database Management System (DDBMS) as a software system that manages a distributed database while making the distribution transparent to the user. The computer systems are connected together for the purpose of sharing data and information via sending message over a communication network to achieve a common objective. In this research, an attempt is made to design a framework for a client/server distributed database system for an Internal Revenue system for ministries and local government areas of Bayelsa state. Wikipedia (the free Encyclopedia) defined internal revenue as a government agency and a bureau of the department of the Treasury, and is under the immediate direction of the commissioner of Internal Revenue. The IRS is responsible for collecting taxes and administration of the internal Revenue code. Also it can be defined as A federal agency, part of the Department of the Treasury, that collects most federal taxes, including income and Social Security taxes (The American Heritage® New Dictionary of Cultural Literacy (3rd ed.)).

The system consists of a relational database of internal revenue variables (entities) that could be stored by the local governments of the state. Each of the Local Government will form a website, and the database will be hosted by the Board of Internal Revenue in the state capital. The local government areas will access the database via the distributed network. The client/server distributed network architecture is used for the development of the system. The system is capable of monitoring all revenue payable to the board of internal revenue of the Bayelsa State.

2. Internal Revenue Bayelsa State

The Nigerian tax system imposes tax on the income of individuals who are considered to be tax residents in Nigeria, and they will be taxed on their worldwide income which is also applicable to Bayelsa State. Income tax is levied at progressive rates on an individual's taxable income for the year, which is calculated by subtracting allowable deductions from the total assessable income.

The enabling legislation, which is the Personal Income Tax (PIT) Act, provides the conditions for deeming income taxable in Nigeria. It provides, among others, that the gain or profit from an employment shall be deemed to be derived from Nigeria if the duties of the employment are wholly or partly performed in Nigeria (or the employer is in Nigeria), unless the duties of the employment are wholly performed, and the remuneration paid, in a country other than Nigeria. The exception to this being whether or not the individual performed such duties during a temporary visit or leave in Nigeria.

In Bayelsa State the Board of Internal Revenue (BIR) is in charge of collecting these taxes through the enabling legislative acts. The BIR collects not only the Personal Income Tax (PIT) but also have the following as revenue payable to the board: Pay As You Earn (PAYE), Direct Tax, Withholding Tax,

Property Tax, Stamp Duty, Drivers License, Vehicle Registration and Bayelsa Infrastructure Maintenance Tax. These taxes are collected by the local government and other establishments saddled with this responsibility and they remit to the BIR.

3. Theory of Distributed Database and Distributed Database Management System

A distributed system is defined as an information processing system that contains a number of independent computers that cooperate with one another over a communication network in order to achieve a specific objective. It can also be defined as a database in which portions of the database are stored on a multiple computers within a network. The definitions above means that a distributed system is not stored in its entirety at a single location. Instead, it is spread across a network of computers that are geographically dispersed and connected through a communication network by which each site is a database of its own. The sites have agreed to work together so that data user can access data anywhere in the network exactly as if the data where all stored at the user's own site. Users have access to the portion of the database at their location so that they can access the data relevant to their tasks without interfering with the work of others. Distributed databases system use a client/server architecture to process information requests. It allows faster local queries and can reduce network traffic. The diagram below illustrates it all.

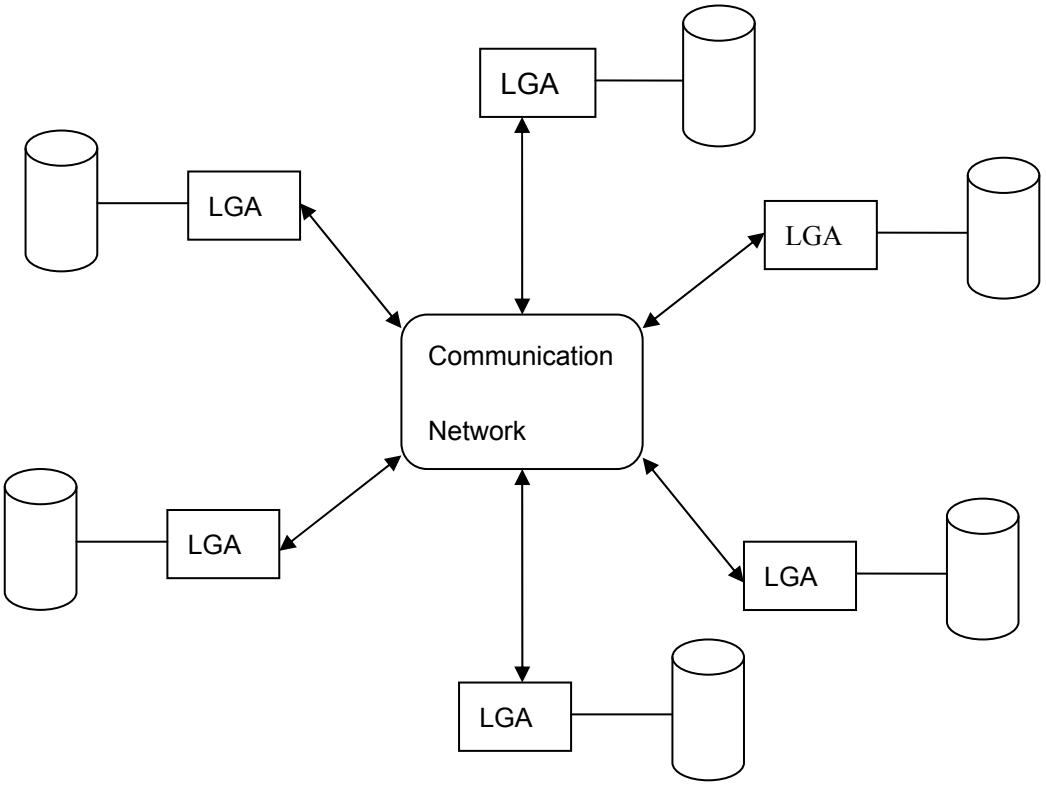

Figure 1. A Distributed Database Architecture

3.1 Advantages of Distributed Database System

- Management of distributed data is done with different levels of transparency (Replication, Fragmentation, Location, Performance, Transaction and Catalog transparency).
- Capacity and Growth: An advantage of distributed databases is that as the organization grows, new sites can be added with little or no upheaval to the DBMS. This situation can be compared to a centralized system, where growth entails upgrading with changes in hardware and software that affect the entire database.
- Reliability and Availability: An advantage of distributed database is that even when a portion of a system (i.e., a local site) is down, the overall system remains available. With replicated data, the failure of one site still allows access to the replicated copy of the data from another site. The remaining sites continue to function. The greater the accessibility the more it enhances the reliability of the system.
- Efficiency and Flexibility: An advantage of distributed databases is that data is physically stored close to the anticipated point of use. Hence, if usage pattern changes, then data can be dynamically moved or replicated to where it is most needed.
- Distributed Database Sharing: An advantage of distributed databases is that users at a given site are able to access data stored at other sites and at the same time retain control over the data at their own site.
- Protection of Valuable Data: Assuming there was a catastrophic event such as a fire outbreak, all of the data would not be in one place, but distributed in multiple locations.
- Improved Performance: Data is located near the site of greatest demand, and the database systems themselves are parallelized, allowing load on the databases to be balanced among servers (A high load on one module of the database would not affect other modules of the database in a distributed database).

3.2 Disadvantages of Distributed Database System

- Increased storage and infrastructure requirements because multiple copies of data are required at various separate locations which would require more disk space.
- Security lapses have increased since data are in multiple locations.
- Integrity control becomes more difficult.
- Database design becomes more complex.

A key objective for a distributed system is that it looks like a centralized system to the user. The user does not need to know where a piece of data is stored physically.

3.3 Categories of Distributed Data

There are five categories of distributed data. They include:

- Replicated data.
- Horizontally fragmented data.
- Vertically fragmented data.
- Reorganized data.
- Separate-schema data.

A Distributed Database Management System (DDBMS) is a software system that permits the management of a distributed database and makes the distribution transparent to the users. A centralized Distributed Database Management System (DDBMS) manages the database as if it were all stored on the same computer. The DDBMS synchronizes all the data periodically, and in cases where multiple users must access the same data, it ensures that updates and deletes performed on the data at one

location will be automatically reflected in the data stored elsewhere. The users and administrators of a distributed system should with proper implementation, interact with the system as if the system was centralized. Query optimization is essential if a DBMS is to achieve acceptable performance and efficiency. Distributed Database Management System is required to maintain distributed database and make it transparent to clients. Sometimes, distributed database is used to refer jointly to the distributed database management system. Consequently, an application can simultaneously access and modify the data in several databases in a network. The main thing that all of such systems have in common is the fact that data and software are distributed over multiple sites connected by some form of communication network. However, the Distributed Database Management System basically addresses the following technical processes:

- Replica Synchronization: This is about synchronizing data based on relatively smaller transactions where the said transactions may consist of several read and write operations on the server. But some applications can take relatively bigger data production jobs. Also, it can write a whole file which can be a relatively large transactional file.
- Synchronous and Asynchronous Replication: Replication may be done through synchronous or asynchronous or batch replication method which makes replicas be in synch (synchronous) or out of sync (asynchronous) for a certain period of time. Update reconciliation may be done at certain time intervals such as every hour or every night.
- Network Servers and Load: This refers to the management of computer on the network networks nodes which can act either as server or client or both server and client at certain circumstances. Under this area, other important considerations include traffic management and security aspects.
- Heterogeneous Data Stores Management: Different computer servers may be implemented on different platforms, so support for heterogeneous data store should be greatly considered. Different kinds of data may be stored in different formats by different vendors. Even in the case of two different database paradigms namely; relational and object oriented, a DDBMS needs to consider this aspect. The standard protocol used for directory information such as Lightweight Directory Access Protocol (LDAP) falls under this consideration.

3.4 Client/Server

Client/server systems are constructed so that the database can reside on a central computer, known as a server, and be shared among several users. Users access the server through a client or server application. In large client/server systems, thousands of users may be connected to a SQL Server installation at the same time. SQL Server has full protection for these environments, with safeguards that prevent problems such as having multiple users trying to update the same piece of data at the same time. SQL Server also allocates the available resources effectively, such as memory, network bandwidth, and disk input/output, among the multiple users.

Advantages of Client/Server

- More efficient division of labour.
- Horizontal and vertical scaling of resources.
- Better price/performance on client machines.
- Ability to use familiar tools on client machines.
- Client access to remote data (through standards).
- Full DBMS functionality provided to client workstations.

3.5 Collaborating Server

Here, we can have a collection of database servers, each capable of running transactions against local data, which cooperatively execute transactions spanning multiple servers. When a server receives a query that requires access to data at other servers, it generates appropriate sub-queries to be executed by other servers and puts the results together to compute answers to the original query. Ideally, the decomposition of the query should be done using cost-based optimization, taking into account the costs of network communication as well as the local processing costs.

3.6 Middleware

Middleware is a crucial component of modern IT infrastructure. It is a set of common business-unaware services that enable applications and end users to interact with each other across a network. In essence, middleware is the software that resides above the network and below the business-aware application software. The services provided by these routines are available to the applications through Application Programming Interfaces (APIs) and to the human users through commands and/or Graphical User Interfaces (GUIs).

4. Framework for the Client/Server Architecture of a Distributed System for an Internal Revenue System for Bayelsa State

Among the types of system architectures for information processing, the client-server architecture is chosen for this framework. A Client is a computer or device requesting data from the Server, which is a computer hosting the data. It works through a network protocol, more likely than TCP/IP. Servers are computers that hold the actual databases and run only the database management system and related software. They are usually multiprocessor computers, with generous memory and raid disk arrays used for stable storage. Hardware database accelerators, connected to one or more servers via a high-speed channel, are also used in large volume transaction processing environments. Clients rely on servers for resources, such as files, devices, and even processing power. Interaction between client and server might proceed as follows during the processing of a SQL query:

- The client parses a user query and decomposes it into a number of site queries. Each site query is sent to the appropriate server site.
- Each server processes the local query and sends the resulting relation to the client site.
- The client site combines the results of the sub-queries to produce the result of the originally submitted query.

In this approach, the SQL server has also been called a transaction server (or a Database Processor (DP) or a back-end machine), whereas the client has been called an Application Processor (AP). In large client/server systems, thousands of users may be connected to a SQL Server at the same time. SQL Server has full protection for these environments, with safeguards that prevent problems such as having multiple users trying to update the same piece of data at the same time. SQL Server also effectively allocates the available resources, such as memory, network bandwidth, and disk I/O, among the multiple users. While SQL Server works effectively as a server, it can also be used in applications that need stand-alone databases stored locally on the client. SQL Server can configure itself dynamically to run efficiently with the resources available on a client, without the need to dedicate a database administrator to each client. However, the interaction between client and server can be specified by the user at the client level or through a specialized DBMS client module that is part of the DBMS package. For example, the user may know what data is stored in each server; break down a query request into site sub-queries manually; and submit individual sub-queries to the various sites.

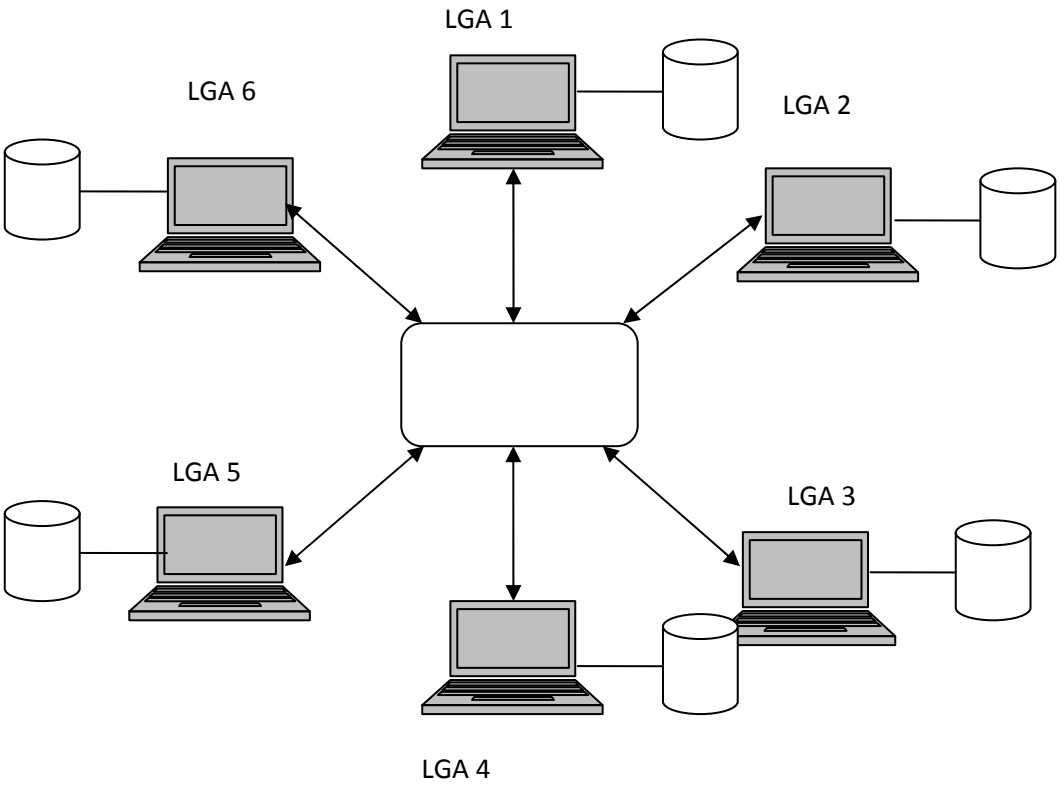

Figure 2. Client/Server Architecture of a Distributed System

Software modules in a typical DDBMS are divided into three levels:

- The server software is responsible for local data management at a site, much like the centralized DBMS software.
- The client software is responsible for most of the distribution functions; it accesses data distribution information from the DDBMS catalog and processes all requests that require access to more than one site. It also handles all user interfaces.
- The communications software (sometimes in conjunction with a distributed operating system) provides the communication primitives that are used by the client to transmit commands and data among the various sites as needed. This is not strictly part of the DDBMS, but it provides essential communication primitives and services.

The client is responsible for generating a distributed execution plan for a multi-site query or transaction and for supervising distributed execution by sending commands to servers. These commands include local queries and transactions to be executed, as well as commands to transmit data to other clients or servers. Hence, client software should be included at any site where multi-site queries are submitted. Another function controlled by the client (or coordinator) is that of ensuring consistency of replicated copies of a data item by employing distributed concurrency control techniques.

5. System Design

The Internal Revenue will have its design as follows.

Database Design

A database is a collection of logically related files (tables). A database management system is a software program used in the creation and management of database. A file (table) is a collection of logically related records. Each record is uniquely identified by a primary key. A record is a collection of logically related fields (columns) that uniquely identifies a person. Here is a list of information that will be required at the BIRS to show that an individual pays his tax.

These are some of the inputs that are to be considered when gathering information of a tax payer,

ENROL INDIVIDUAL: Name: Surname, First Name, Middle Name, Nationality, Phone number.

RESIDENTIAL ADDRESS: State, LGA, Ward, City/Town/Area, Street Name, House No, Issuing Authority.

MEANS OF IDENTIFICATION: National ID, Driver's License, International Passport, Government Parasternal, Resident Permit, Registered Organization, Letter from a Recognized Individual, ID No, Issuance Date, Expiry Date, Place of Issuance, ID Issuing Authority.

ADDITIONAL INFORMATION: Title (Mr, Mrs, Miss, others), Sex, Date of Birth, Occupation, Marital Status, Mothers Name, Mothers Maiden Name, Email Address.

MAILING ADDRESS: P.O BOX.

Tax Assessment/Payment Details: Last Assessment Date, Last Assessment Amount, Last Payment Date, Last Payment Amount, Tax type, Total Income for the past three years.

Tax Representative Details: Name, TIN, Reason, Rep Type.

Representative Details: State, LGA, Ward, City/Town/Area, Street Name, House No, Phone No, Email Address, Source of Income, Employee, Self Employed, Organization Owner/Partner, No Income, Start Date, Employer Name, Employer TIN, Start Date of Employment.

Self-Employment Details: Name, Commencement Date, Business Name, State, LGA, Ward, City/Town/Area, Street Name, House No, Sector, Line of Business, No of Employees.

Ownership Details: Ownership Name, TIN, Ownership start Date, Shareholding (%).

Asset Details: Asset Owned (Types of Assets, Location of Asset, Market Value (N), Start Date of Ownership).

Dependent/Children/spouse: Surname, First Name, Middle Name, State of Origin, Date of Birth, TIN, Relationship Type.

A careful study and analysis of the above data suggest that while the different sites (the LGA's) collect data differently and through disjointed efforts, they all basically require the same set of data. In this research, an attempt was made to formulate a standard operational procedure for the BIRS alongside with the sites (the LGA's) so as to bring effectiveness, reduced fraud and financial distress in the area of generating internal revenue. Moreover, a centrally managed database is presented with an element of multi-level data access.

5.1 Components of the Distributed System

The distributed system has Internet facing Web-enabled applications that can be accessed remotely by the users either within the organization or remotely. The following are a list of the Information Technology (IT) infrastructure components of the system.

Firewall: A system designed to prevent unauthorized access to or from a private network. Firewalls can be implemented in both hardware and software, or a combination of both. Firewalls are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet,

especially intranets. All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria. There are several types of firewall techniques;

- *Packet Filter*: This looks at each packet entering or leaving the network and accepts or rejects it based on user-defined rules. Packet filtering is fairly effective and transparent to users, but it is difficult to configure. In addition, it is susceptible to Internet Protocol (IP) spoofing.
- *Application Gateway*: This applies security mechanisms to specific applications, such as File Transfer Protocol (FTP) and Telnet servers. This is very effective, but can impose performance degradation.
- *Circuit-Level Gateway*: This applies security mechanisms when a Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) connection is established. Once the connection has been made, packets can flow between the hosts without further checking.
- *Proxy Server*: Intercepts all messages entering and leaving the network. The proxy server effectively hides the true network addresses.

Router: A router is a special purpose computer or software device that enables two or more dissimilar networks to communicate. Routers route traffic, which consists of Transmission Control Protocol/Internet Protocol (TCP/IP) packets.

Host: A computer that is connected to a TCP/IP network, including the Internet.

Server: A server is a dedicated computer that allows other computers to connect to it. The following are the various types of sever available:

- Domain Name System.
- Web servers.
- Internet banking servers.
- E-mail servers.
- Proxy servers.

Workstations: In networking, a workstation refers to any computer connected to a local area network. It could be a workstation or a personal computer.

Intrusion Detection Systems: Intrusion detection is fundamentally the process of monitoring computer networks and systems for violations of computer policy.

5.2 Network Structure

A Metropolitan Area Network (MAN), provides long distance transmission of data, image, audio, video information over large geographic area that may comprise of a country, a continent, or even the whole world. MAN is best for this kind of system. This system grew from earlier community antenna systems used in areas with poor over-the-air television reception. In these early systems, a large antenna was placed on top of a nearby hill and the signal was then piped to the subscriber's homes. The diagram below describes the structure of this framework.

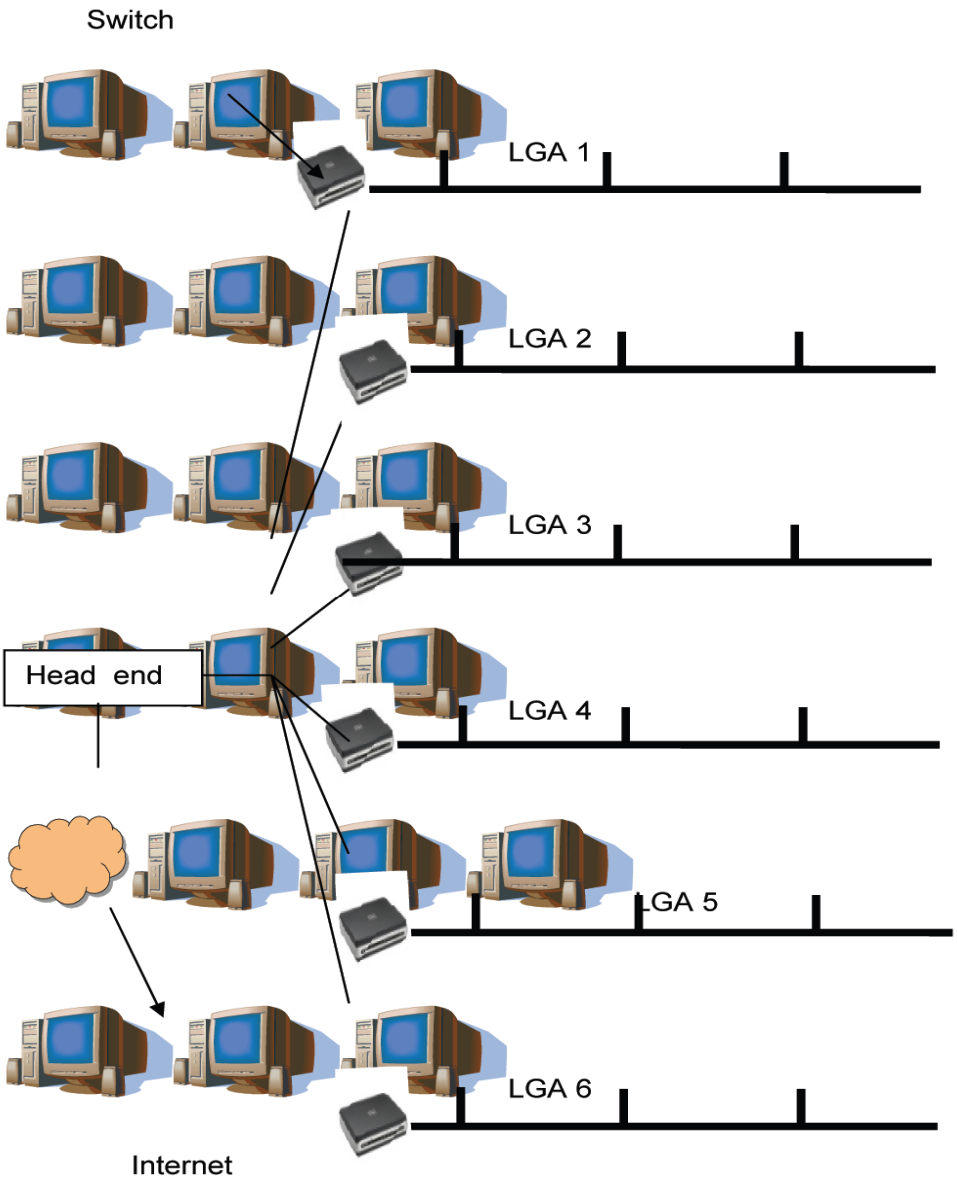

Figure 3. Metropolitan Area Network (MAN) Based System

6. Conclusion

The use of a Distributed Database Management System (DDBMS) by the local government is a reliable, secured, and effective way of monitoring an employee's financial record. It is also used in detecting and monitoring how much revenue that is been generated at the local government, and in generating reports concerning an employee's financial report as regards to its taxable income thereby making tax clearance a lot easier. This stems from the fact that the management of distributed data is done with different levels of transparency. However, such transparency include: replication transparency, fragmentation transparency, location transparency, performance transparency, transaction transparency, and catalog transparency. A distributed database gives an added advantage for new sites to be added with little or no upheaval to the DBMS as the organization grows and expands.

Furthermore, employees data stored are reliable and easily accessible. For instance, when a part of the system (i.e., a local site) is down, the overall system still remains available. With replicated data, the failure of one site still allows access to the replicated copy of the data from another site. Thus, the

remaining sites continue to function. It is efficient, flexible, and supports file sharing. As a result, users at a given site are able to access data stored at other sites and at the same time retain control over the data at their own site. The use of a distributed database for an internal revenue system helps to maintain high level of security and protection of employee's data in the various local government of the state. Assuming there was a catastrophic event such as a fire outbreak, all of the data would not be in one place, but distributed in multiple locations. As such, this makes it possible to have access to same data from other location. It has a firewall which helps to prevent unauthorized access to or from a private network. This fire wall might be in the form of packet filter, application gateway, circuit-level gateway, or a proxy server. Other components of the distributed network system that helps to maintain high level of security include a router, host, server, workstation, and an intrusion detection system.

Consequently, the use of a client-server distributed database system for an internal revenue system shows an improved performance. Data is located near the site of greatest demand, and the database systems themselves are parallelized, allowing load on the databases to be balanced among servers. This distributed database management system employs a Metropolitan Area Network (MAN). The Metropolitan Area Network (MAN) provides long distance transmission of data, image, audio, video information over large geographic area that may comprise of a country, a continent, or even the whole world. In conclusion, the use of the client-server distributed database for an internal revenue system would facilitate a secured, reliable, and available system for keeping records of internally generated revenues as well as financial records of employee's in the different local government areas in Balyesa State, Nigeria.

Limitation and Future Research

Despite all its advantages, there are some limitations as to which the system can operate and these includes:

- Increased storage and infrastructure requirements because multiple copies of data are required at various separate locations which would require more disk space.
- Security lapses have increased since data are in multiple locations.
- Integrity control becomes more difficult.
- Database design becomes more complex.

Based on these limitation I can say further research can be made on each these limitations as to encourage further studies.

References

Bayelsa State Board of Internal Revenue Services (BIRS). (n.d.).

Bayelsa state local government commission. (n.d.).

Candace, C. F., & Barbara, von H. (1989). *Handbook of Relational Database Design*. Addison Wesley Longman.

Charles, P. P., & Shari, L. P. (2003). *Security in Computing, Elmasri and Navathe, Fundamentals of database systems* (3rd ed.). Addison-Wesley Longman.

Fred, R. M., & Jeffrey, A. H. (1994). Modern *Database management*. England, Addison Wesley Longman. Retrieved May 17, 2004, from http://www.cs.rpi.edu/~noel/distr_scaleup/distriuted.html

Navathe, E. (2000). *Fundamental of Database Systems* (3rd ed.). Teturo Sawada Exclusive Publisher and Distributor.

Ozsu, M. T., & Valduriez, P. (2011). *Principles of Distributed Database System* (3rd ed.). Springer. Prentice Hall Professional Technical Reference, Upper Saddle River, New Jersey.

Rick, N. (n.d.). Scale Up in Distributed Databases: A Key Design Goal for Distributed System. *The American Heritage New Dictionary of Cultural Literacy* (3rd ed.). Copyright © 2005 by Houghton Mifflin Company. Published by Houghton Mifflin Company.

Thinking Beyond Borders: Management of Extended Business Travelers. (2012). *KPMG International Cooperative* ("KPMG International").

Thomas, M. C., & Carolyn, E. B. (2002). *Database Systems* (A Practical Approach to Design, Implementation, and Management (3rd ed.)). Pearson Education Limited. Addison Wesley.