

# Design and Implement an EC-Application Based on Multi Database Environment

**S. M.Hadi**

**Department of Information and Communication Engineering  
College of Al-Khwarizmi Engineering, University of Baghdad**

## Abstract

The vast majority of EC applications are the web-based deployed in 3-tire Server-Client environment, the data within such application often resides within several heterogeneous data sources. Building a single application that can access each data sources can be a matter of challenging this paper concerns with developing a software program that runs transparently against heterogeneous environment for an EC-application.

*Keywords: EC, Heterogeneous DB, Three Tier Architecture, Middleware*

## Introduction

Today's organization faces an increasingly rapid rate of changes that are fueled by the rise of electronic business. To sustain competitive advantages, The IT professional must deliver rapid solutions for changing business needs. From the other side the distributed database system has recently become an important area of management information processing that eliminates many of the shortcomings of centralized DB and fits more naturally in the decentralized structure of many organizations [1].

Various commercial trends have lead to an increasing demand for distributed systems. First, the number of mergers between companies is higher and this trend is bound to continue. The different divisions of merged company have to deliver unified services to their customers and this usually demands an integration of their systems [2]. The time frame is often so short that building a new system is not an option and therefore existing system components have to be integrated into a distributed system that appears as an integrating computing facility, also the time pressures on providing new services or existing services to new customers are increasing rapidly.

Often, this can only be achieved if components are procured off the shelf and then integrated into a system rather than built from scratch. Components to be integrated often have incompatible requirements on the hardware and operating system platforms they run on then they have to be deployed on different hosts; after that the systems end up being distributed.

In this setting, it is difficult to estimate the scalability requirements for such a system. Any E-commerce site that designed to cope with a given number of transactions per day may find itself suddenly exposed to demand by orders of magnitude larger. The required scalability cannot usually be achieved by using centralized or client – server architectures and these systems often have to be distributed. The construction of distributed systems is appealing because it can possibly solve all these problems.

Distributed systems can integrate legacy components, they can decrease the time to market, and they can be scalable and tolerant against failures [3]. The caveat, however, is that the construction of truly distributed systems is considerably more difficult than building a centralized or client/server system. This is because there are multiple points of failure in a

distributed system, components need to communicate with each other through a network, which complicates communication and opens the door for security attacks. Building a proposed DB repository about customers, agents and products will manage the data from different sources of DB platforms into one application will be the optimum solution for such an application.

## **The Client / Server Model over Distributed DB**

In server / client architecture, multiple computer platforms are dedicated to special functions such as database management, printing, communications, and program execution. These platforms are called servers. Each server is accessible by all computers on the network. Servers can be computers of all sizes; they store both application programs and data files and are equipped with operating system software to manage the activities of the network. The server distributes programs and data files to the other computers (clients) on the network as they request them. An application server holds the programs and data files for a particular application, such as an inventory database. Processing can be done at the client or server side [4].

A client is any computer (often an end user's personal computer) that sends messages requesting services from the server on the network. This client can converse with many servers concurrently. A user at a personal computer initiates a request to extract data that resides in a database somewhere on the network. A data request server intercepts the request and determines on which data server the data resides. The server then formats the user's request into a message that the database server will understand. Upon receipt of the message, the database server extracts and formats the requested data and sends the results to the client. Only the data needed to satisfy a specific query is sent [5].

While the distributed processing involves placing processing units at different locations and linking them via telecommunications equipment, a distributed database is a database in which the data may be spread across several smaller databases connected via telecommunications devices works on much the same principle. And the DBMS determines where the data is physically located and retrieves it. Distributed database give corporations more flexibility in how database are organized and used. Local office can create, manage, and use their own databases, and people at other office can access and share the data in the local databases [6]. Giving local sites more direct access to frequently and used data can improve organizational effectiveness and efficient significantly.

Despite its advantages, distributed processing creates additional challenges in maintaining data security, accuracy, timeliness, and conformance to standards. Distributed database allows more users direct access at different sites; thus controlling those accesses and changes data is sometimes difficult [7].

Client server model is usually based on a simple connection (request / replay) protocol. In this suggested software as a middleware, we connect each client computer by the software communication to the server computer; this software layer can aid the transmission and resaving of the client data and control the information between them and the server. This model depends on one type of client / server architecture which is named Three Tier Architecture.

## **The Three Tier Architecture**

Figure (1) shows the three tier architecture that have been used for Internet-technology database applications. In this figure, three types of processing, or tiers, are shown from right to left, they are: the database server, the web server, and the browser or client computers. Each of these tiers can run on a different operating system [8].

The interface between the web server and the database server transmits SQL statements and relational data. While the interface between the web server and the browser transmits web pages, client code and data. The functions of the three tiers that are summarized in this fig show that the database server is to run the RDBMS to process SQL statements and perform database management task. In this case the DBMS is operating in its traditional role of serving up data; the RDBMS on the data server is not creating forms or reports or menus. Instead, it is a pure data receiving SQL requests and processing rows in tables.

## The proposed distributed software design

When the network technology had achieved sufficient penetration and visibility, the term of the middleware software had evolved into a much richer set of paradigms and services and offered to help make these services easier and more manageable.

The programming of distributed system is in general much more difficult without a special software program to manage the heterogeneous operation, the suggested software for the proposed EC-application designed to work transparently and to be suitable in the heterogeneous relational database management system environment, where two or more different DBMS are used for the EC- web based application and running under different operating system.

The idea of the main architecture for this proposed software is shown in Figure (2) The software design for this process can be extended to take advantages of extended tools for remote computing and distributed application that is needed in the EC application.

This software ware provide simple mechanism to ensure that the application to be distributed over multiple heterogeneous platforms will reduce the complicity of developing applications that span multiple operating systems and network protocols.

The flowchart diagram for this software can be explained through Figure (3) in this flowchart the main EC application based a web page used, as the interface for all the functions of this application and different Data Base will be connected by using the open database connectivity (ODBC).

## Software Implementation

The designed application can be written and implemented in any language used for processing database applications over the Internet technology. According to that the Open Database Connection (ODBC) must be used in which it is the standard that will be used between the windows application and the middleware application that is independent of the data source. The ODBC middleware will provide the application software with a generic format for the database access.

The designed software has been written by using Oracle 9i language with developer 2000 to build the database and the EC application and in order to use three other databases platforms to create the heterogeneous environment the following languages have been used (Access 2003, VFoxPro. and Informix).

By implementing the steps explained in Figure (3) ,the system will distribute the data transparently through this environment by choosing a specific ODBC driver as Oracle to be used for each database being accessed. For any EC application, it should requeste the five steps sales life cycle (search, identifying selecting, purchasing, delivery and services after sales) this proposed software started by the registration of any new customer or otherwise they have only to sign in the registration forums explained in the windows shown in Figure (4)a and (4)b While the other forums will be used to establish a connection to a non oracle database sources other than the primary one, and the data stored in that non oracle database

will be retrieved as needed. Figure (5)a and (5)b show the appliances order and the connection with the other Data Base for this EC Application.

## The Conclusions

Developers today are challenged to integrate their applications with other systems over a network in a multi platform and multi databases. By this flowchart design and its implementation software we reach this idea, it can be designed to manage the complicity of the heterogeneous inherent in the distributed database used in any internet based application just the example taken here for the EC application. Also it helps to ease the task of designing and managing different distributed database system over different operating system by providing and improving the consistency, integrity and security.

## References

1. Ralph, M. Stair, (2003).Principles of Information Systems, International Thomson published company, 2<sup>nd</sup> edition, Canada
2. Rob, P. and Copnel, C. (2005). Database System Design, Implementation and Management, International Thomson Publishing Company ITP, 3<sup>rd</sup> edition, USA.
3. Michael, J. Hernandez, (2003). "database design for mere", Addison Wesley, 2<sup>nd</sup> edition, USA,
4. Charles, J.(1999).Middleware Moves to the Forefront, Addison Wesley publishing company, Inc., USA.
5. Hergula, K. ;Harder, T.; DaimlerChrysler, A.G.( 2000). "A Middleware Approach for Combining Heterogeneous Data Sources - Integration of Generic Query and Predefined Function Access," in Proceedings of the 1st International Conference on Web Information Systems Engineering, 1, 26-33.
6. Chen, Q; Yao J, and Xing R, (2006). "Middleware Components for E-commerce Infrastructure: An Analytical Review" Issues in Informing Science and Information Technology, 3, 137-146.
7. Ling R.R.; Yen, D. C.and Chou, D, C.,( 2000-2001), " From Database to Web Browser: The Solutions to Data Access", Journal of Computer Information System ,winter
8. Cavazos, F.A. and Jarquín, J.C.(2004),"A 3-Tiered Client-server Distributed Database System Component-Based", in Proceedings of the winter international symposium on Information & communication technologies,.

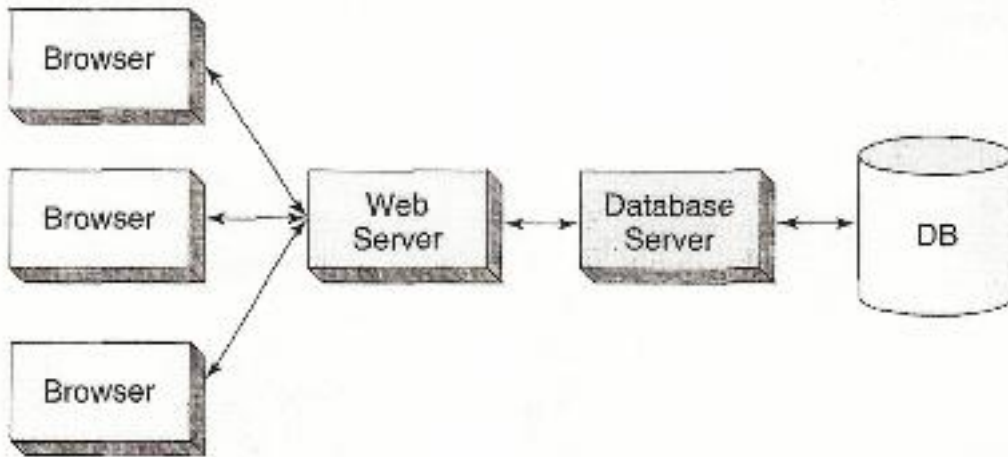


Fig. (1): The Three Tire Architecture

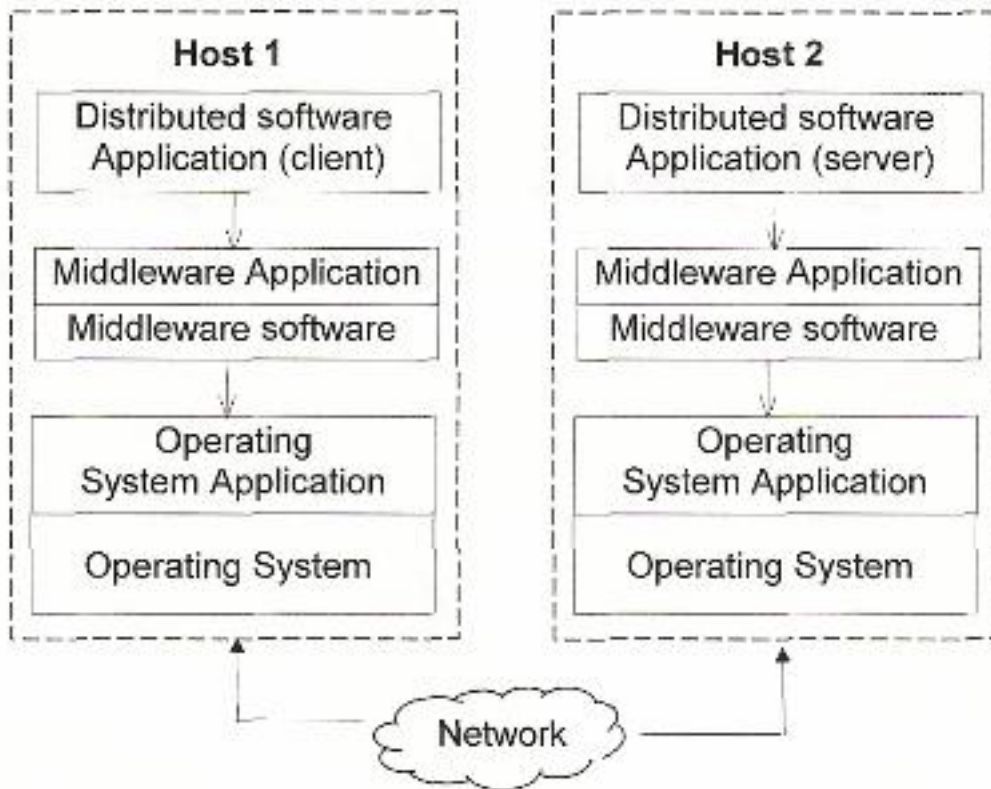


Fig.(2): The Main Structure of the Proposed Algorithm

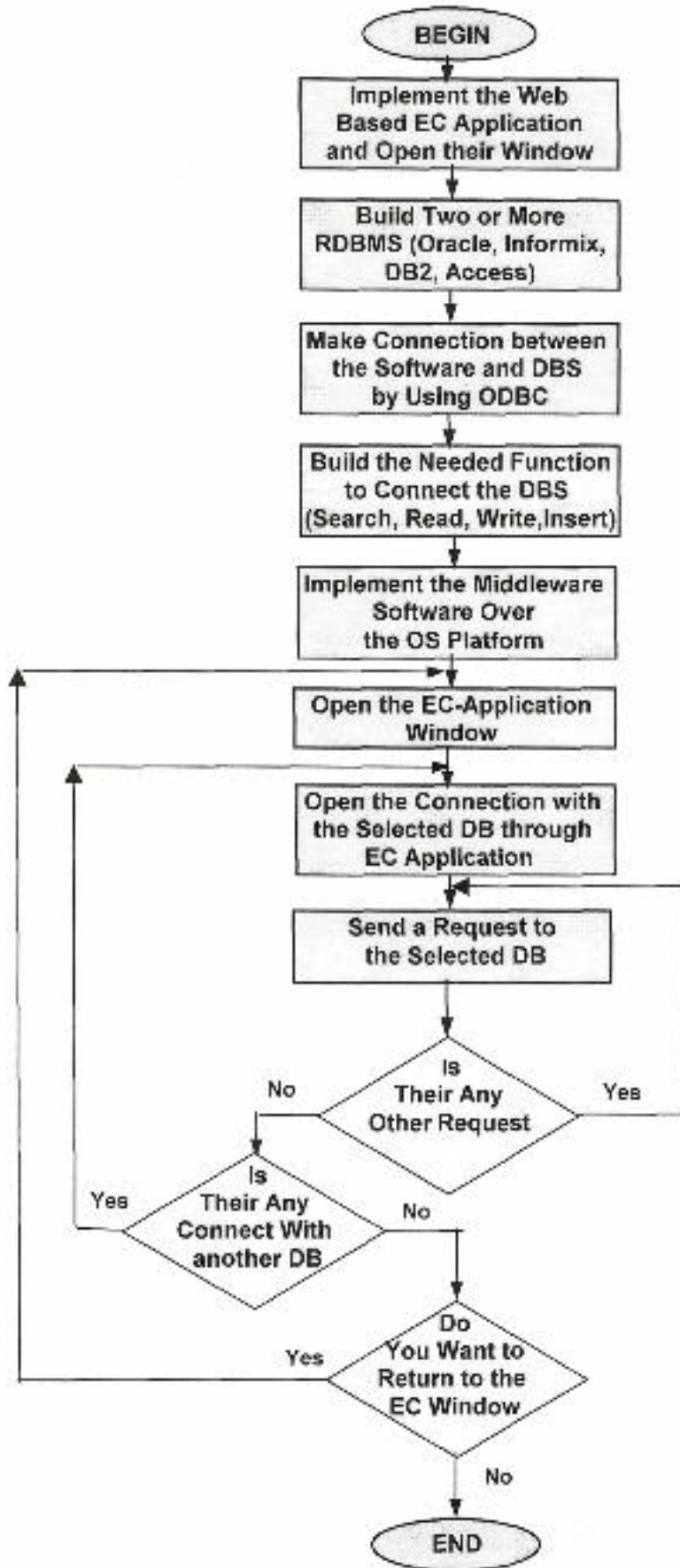


Fig. (3) :The Flowchart of the Proposed Program

<b>REGISTER</b>	SING IN	CATALOG	BUY
-----------------	---------	---------	-----

USER NAME   
 PASSWAORD

---

USER NAME   
 COUNTRY   
 ADDRESS   
 E-MAIL

---

(a)

REGISTER	<b>SING IN</b>	CATALOG	BUY
----------	----------------	---------	-----

USER NAME   
 PASSWAORD

---

(b)

**Fig. (4): The Registration and Sign in Operation for EC Application**

REGISTER | SING IN | **CATALOG** | BUY

Oracle | Foxpro | Ms-Access | Informix

CODE	PRODUCT	CATALOGE	MODEL	PRICE	BUY
100	Aircooler	Coolers	Vstel	300 \$	Details ...
200	Fan	Coolers	Konka	75 \$	Details ...
300	Stand fan	Coolers	national	100 \$	Details ...
400					Details ...
					Details ...

BUY

(a)

REGISTER | SING IN | CATALOG | **BUY**

ORDER NO; :- 750  
 ORDER DATE :- 20-FEB-2008

CODE	PRODUCT	UNIT PRICE	QUANTITY	SUM
100	Aircooler	300 \$	2	600 \$

CLEAR ORDER | CHACK OUT | TOTAL PRICE: 600\$

(b)

Fig. (5): The Connection with Other DB for EC Application



## تصميم وتنفيذ تطبيق للتجارة الالكترونية يعتمد على بيئة قواعد البيانات التوزيعية

سها محمد هادي

قسم هندسة المعلومات والاتصالات ، كلية الهندسة الخوارزمي ، جامعة بغداد

### الخلاصة

ان من أهم التطورات الكبيرة والمهمة في تطبيقات التجارة الالكترونية هي تطوير هذه التطبيقات من خلال صفحات الويب المعتمدة على هيكلية أو معمارية الخام والمستفيد (Server- Client)، اذ ان البيانات ومن خلال هذه التطبيقات تعمل ضمن بيئة العديد من مصادر قواعد البيانات التوزيعية المختلفة .

ان بناء نظام واحد متكامل يمكنه الدخول والوصول الى كل مصدر من مصادر قواعد البيانات التوزيعية والمختلفة لهذه التطبيقات والتعامل معها يعد تحديا " كبيرا" أن هذا البحث يهتم بإنشاء وتطبيق خوارزمية وبرنامج يتعامل بشفافية مع مصادر قواعد البيانات التوزيعية المختلفة والخاصة بتطبيقات التجارة الالكترونية.