MANAGEMENT SCIENCE AND ENGINEERING
Vol. 4, No. 4, 2010, pp. 113-119
www.cscanada.org

ISSN 1913-0341 [Print]
ISSN 1913-035X [Online]
www.cscanada.net

The Design of B2B E-commerce System Based on MVC Model and J2EE

GUO Chun-yan[1]

ZHU Kong-lai [2]

MA Zong-guo[3]

Abstract: In the paper, the architecture of J2EE and the method of development system with J2EE were described. Among of these technologies, Web application with the MVC model to develop was emphasized. On the base of the B2B business trade flow, requirement analysis was discussed in detail. It emphasized that the modeling and system designed and implemented. A B2B e-commerce system of four-layer architecture by EJB component technology was established.

Key words: B2B E-Commerce system; J2EE technology; EJB component; MVC model

1. INTRODUCTION

E-commerce is a new business model which realizes the consumer shopping online and transactions online between merchants and electronic payment online, that based on browser/server application in an open network environment. According to the demand for transactions online of some enterprise, we design and development a sale e-commerce system based on EJB. That realized transaction automation between buyers and sellers which made the orders as the center, solved the problems of data exchange and communication with suppliers, agents, distributors in commercial operation.

2. SYSTEM ARCHITECTURE

Combined the B2B enterprise's characteristic and the actual demand, four layers structure of J2EE was used in the system based on the Web, and the B/S model, the client do not need to install client software.

J2EE provides a comprehensive function support on the development of Web-based multi-layer application system. In the system's structure, the client is the thin customer, which provides a convenient interface for the customer, the supplier, the seller, the manager and the management activity; Web layer responses customer requests and submits requests to business logic layer, which improved the efficiency of the system greatly, reduced the burden of the Web layer; business logic layer separates the business logic from other layers, enhances system's extendibility and flexibility; data layer is a large relational database,

[1] Female, research direction: information management, soft power .Research Center of Soft Power, University of Jinan, Jinan Shandong, 250022, China

[2] Research Center of Soft Power, University of Jinan, Jinan Shandong, 250022, China

[3] Research Center of Soft Power, University of Jinan, Jinan Shandong, 250022, China

*Received 11 October 2010; accepted 1 December 2010

which storages customers, merchandise and resource information and so on, guarantees integrity and uniformity of all data of the system.

E-commerce system can be divided into 5 sub-systems which includes commodity management (purchase, sale, inventory, order, payment, etc.), order management (purchase orders), members of management, commodity distribution, information management and statements of statistics. System chart shown in Figure 1.

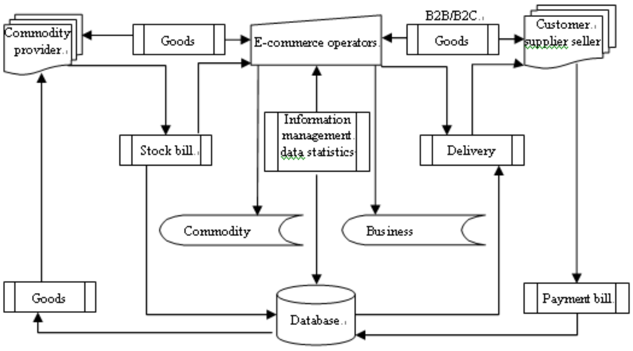

Figure 1. The system chart

2.1 The System Software Model -- J2EE Architecture

At present, the mainstream of enterprise-class application development technology: J2EE based on Java2 is introduced by Sun Corporation, .NET is introduced by Microsoft Corporation, CORBA is introduced by CORBA organization. Three kinds of technologies have their own advantage, I decided to adopt J2EE as the development of the system's main standard when I synthesized the present application situation and each kind of technical concrete characteristic.

J2EE that Java2 Platform Enterprise Edition is a brand-new concept model which was introduced by U.S. Sun Corporation, it has unparalleled advantage in compared with the traditional Internet application. It provides a component-based approach to design, develop and deploy the enterprise system. The J2EE uses multi-layer's distributional model, application logic is divided into components according to the functions. J2EE component is an independent software unit; they compose the J2EE application program through related class and the document, and interact with other components. The following is a typical four-layer structure of J2EE:

The client layer component which is running on the client machine;

The web layer component which is running on J2EE server;

The business logic layer component which is running on J2EE server;

The software of enterprise information system layer which is running on EIS server.

In the design of system architecture, I take into account the following factors: First, the system's users are mainly various kind of enterprises, in order to enable them to use the system in different locations, different times and different circumstances, it is must be taken into account about the system's cross platform. Second, the majority of operations of the system involve to interaction with the user dynamically, therefore it should issue data dynamically. Third, the system has many similar but not identical functions, it is necessary to improve the system modules reusability, cohesion and scalability. Fourth, system's operation needs the backstage database to provide the data management and the control service.

2.2 the Combination of J2EE and MVC

2.2.1 MVC model

MVC (Model-View-Controller) model originated in the Small-talk-80 language, applies in the interface design early, and now as a typical object-oriented design pattern is widely used. MVC model divides the application into the model (Model), view (View) and controller (Controller). Model is the application core that on behalf of the data structure and business logic; View is the external performance of the model that enable customers interact with the system; Controller responses the user's input and relates model and view. From its definition, we can see, MVC model use "partitioning" thought that separates data access and logic process, it solved the problem of the B/S pattern fundamentally, which has the remarkable advantages in constructs in the web application:

(1) MVC model could be applied to multi-user, scalable, maintainable and highly interactive systems, such as the following B2B e-commerce system we introduced.

(2) MVC model can be a very good expression of the interaction model between user and system, expanding the structure of the whole system model.

(3) MVC model can be very convenient to use multiple views to display multiple sets of data, so that we can make the system support of other new types of client easily. In addition to operating desktop browser, but also can run in the PDA, WAP.

(4) Because MVC has separated data control and data performance in the pattern for developers, we can distinguish the responsibility of developers, the backstage developers can focus on the business processing while the onstage focus on user interaction interface, thus can greatly enhance the efficiency of development, shorten the cycle.

2.2.2 MVC Model to Build the J2EE Platform

The structural design of software modules in this system, we have adopted ideas of combination of MVC model and J2EE framework. Following I discussed the method on the combination of system. The system components to the MVC mapping is diverse, entirely assign flexibly according to the actual situation. In the usual situation, we mapped J2EE elements to the three components of the MVC in accordance with the following pattern.

Controller: Servlets;

View: Java Server Pages;

Model: JavaBeans and Enterprise JavaBeans.

J2EE technology unifies MVC design pattern in the construction enterprise Web application[4], JSP corresponds to the view, because the entire system interact with the outside through JSP primarily, may enable developers to develop the web pages in mastering the few programming skill, developers focus their work on the design of the page, without having to care about the source of data and logic; Servlet corresponds to the control, as intermediate between JSP and EJB, they transform user's request into the Model understandable request, and decide to show which view; EJB and JavaBeans deal with the request of data by the Controller, which are corresponding to the model and generate the result sets according to the request, EJB mainly used in packaging business logic and data logic. MVC pattern build the Web application framework shown in Figure 2:

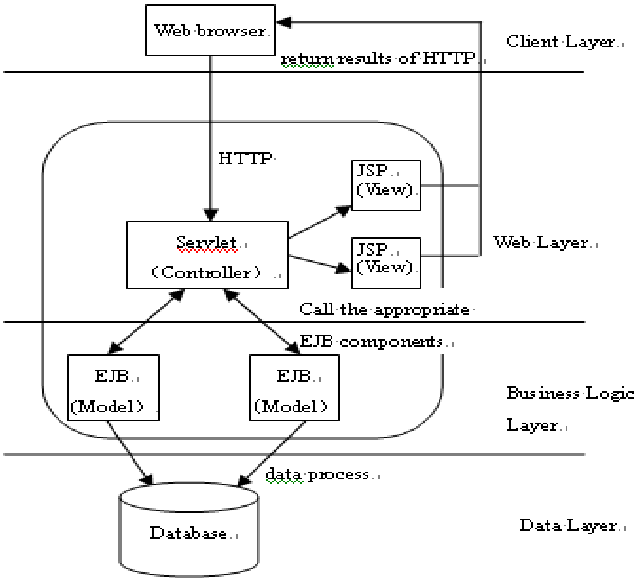

Figure 2. MVC model to build the J2EE platform performance

3. DEVELOPMENT AND IMPLEMENTATION OF COMPONENTS IN THE SYSTEM

In front I introduced the relevant technology in the system, the following I introduce specifically each level design development in combination with the realization of the process.

3.1 Client layer Design and Development

Client is the first layer in J2EE four-layer structure, whose primary role is for users to operate, through the layer to reflect customer demand and show the result of the system to the user. In the several client software commonly, Web browser has obvious advantage, it does not need client maintenance, only one computer will be able to access the Internet as our client, and system users are mainly customers, suppliers, agents and vendors, the large number, the use of dispersion. So I choose B / S mode, the user will be able to realize the function module of the system only through the browser, the operation is simple and convenient.

3.2 Web Layer Design and Development

Web is the second layer in J2EE four-layer structure, whose primary role is to respond to client requests, and return the results to the client. Web layer is mainly JSP and Servlet, JSP and Servlet can deal with customer requests and establish response, which is running on the Web container. Through a combination of two technologies, making the system has a clearer page performance, a clearer delineation of roles and improving the efficiency of development.

3.2.1 JSP Page Development

JSP is running on the server-side component of Web containers, JSP source files are implemented by JSP engine on the Web server. JSP pages are compiled into Servlet on the server. We develop different JSP pages according to different modules, its main function is to provide a functional operation interface to the user and convey the request of users to the Servlet on Web layer which is responsible for control logic.

Users add the JSP page through the browser and enter basic information about users. The page passes the values to the Servlet through the POST method, related processing by Servlet.

3.2.2 Servlet Development

Servlet is the small application procedure that is running on the server based on the Java language. It is the server software which has nothing to do with the agreement and the platform; they operate on the server or application server's Java virtual machine. It is responsible for receiving HTTP requests, dynamically generated HTTP response, Servlet and Servlet engine running on web server interact through request and response. Servlet is responsible for concrete realization and the call EJB, the client call EJB, the main step is as follows:

Find the Home interface of EJB through JNDI.

Obtain the Remote Interface reference of EJB.

call method defined in EJB.

The following is an example which calls EJB to add a user entity Bean in Servlet.

We received the user basic information parameter transmitted by JSP page: userid, name, email, phone, postalcode, then called Bean which add users: AddUser

InitialContext ctx=new InitialContext(); //get the initialized context environment for Bean

Object obj=ctx.lookup(“java:comp/env/ejb/AddUser”); //use JNDI to find adduser

AddUserHome home=(AddUserHome) javax.rmi.PortableRemoteObject.narrow(obj,AddUserHome.class);

//get the Home Interface of Bean

addinstance=home..create(userid,name,email,phone,postalcode); //create a Bean example

addinstance.add(); //call business method of EJB, here is to call a method used to add users.

3.3 Business Logic Layer Design and Development

In the system, the main criteria of design and implementation of Business Logic Layer used EJB2.0. EJB development must be achieved interface definitions (Home and Remote interface) of javax.ejb package and use class implements the public methods of bean. Client accesses Enterprise Bean method through the Home and Remote interface of Enterprise bean. Home interface provides the method which can create, delete, and position enterprise bean, client uses the Create () method to create EJB instance, or uses find method to identify a single EJB instance, and Remote interface provides business processing method which operates EJB instance, so as to be client access. In the deployment, these interfaces create classes in the container, enable customers to create, delete, position or call the business method on enterprise bean. Enterprise bean class provides the realization of business method, the creation of method and query method.

The following is the stateful Bean's add () method code:

public void add (){ // In this method, stateful Bean calls a entity Bean: User

InitialContext ctx=new InitialContext ();

Object obj=ctx.lookup (“java:comp/env/ejb/User”); //find User’s entity Bean through JNDI

UserHome home= (UserHome)PortableRemoteObject.narrow (obj,UserHome.class);

//obtain the Home interface of Bean

home.ejbCreate (userid,name,email,phone,postalcode); //add an user entity

}

The completion of the above code and related links, you can use EJB add users in the Web client .

The above code is a small part of the system, which omitted the connectivity part with the JBoss server as well as part of exception, but can clearly understand how to use EJB to provide services to customers, which has typical.

3.4 Data Layer Design and Development

Data layer is a place to store data, we use SQL Server to store permanent e-commerce data.

The key of the entire database is table structure design, it is independent of the logic structure of the entire database, able to fully reflect the real world, including the links between the entities and can meet customer demand, and easy to expand and modify.

We established five base tables:

(1) Commodity information table (products): The table used to record information of goods (including goods number, name, price, type, units, classification of property, pictures, suppliers briefings, etc.).

(2) Enterprise information table (enterprise): record information of registered member.

(3) Order (orders): every registered member has an order to record their expense.

(4) Product Classification table (dep1, dep2): describe the classification of products.

(5) Business permissions table (userset): business users can operate the module according to division of authority.

4. CONCLUSIONS

We designed and developed component in the multi-layer structure based on the further integration of MVC model, which ensured the system scalability and maintainability. We focused on EJB components technology and applied EJB components theory to the design of the system. Application of a large number of component technology have greatly enhanced the reuse of code, improved development efficiency.

The system is still a lot of inadequacies in need of further improving and enhancing. In the actual work, we still remain the research stage in the model design and analysis, the real part of the achievement is still relatively simple. The true realization of model has a lot of hard work to do. In short, through specific system development, we can see, J2EE technology provides a powerful tool for application development of the B2B E-commerce system. In the future, if we can combine cluster technology and load balancing strategy, we will make the system more powerful.

REFERENCES

CHENG Kunshan, WU Huiyun. (2003). Research on EJB Component-based E-commerce Architecture and Services Platform. J*ournal of Ease China Jiaotong University, 20*(4), 66-67.

DEACON John. (2000). Retrieved from
Model-View-Controller(MVC)Architecture.http://www.jdl.co.uk/briefings/MVC.pdf.

LI Huan-zhe, LIU Xiao-liang, GUO Da-quan, WANG Fan, CAO Qiang, MA Shu-min. (2003). The Discussion of J2EE System Architecture. *Journal of the Hebei Academy of Sciences, 20*(3), 152-156.

LI Kangrong, MIAO Fang. (2008). The Study of E-Commerce System Architecture Based on MVC Model and J2EE Platform. *Computer Security, 9*,33-36.

LU Jiamiao, YE Shaozhen. (2004). Electronic Business System Integrated Design Based on J2EE Multi-patterns. *Journal of Fuzhou University(Natural Science), 32*(6), 677-680.

LU Rongxing, YU Zhou, RUAN Yongliang, WANG Zhiqiang. (2003).Study and Implementation of MVC Design Pattern on J2EE Platform. *Application Research of Computers, 3,*144-146.

ROMAN Ed, AMBLER Scott, and JEWEL Tyler. (2002). *Mastering Enterprise Java Beans, 2nd Edition.* New York: Wiley Computer Publishing, 92.

YANG Peng, LI Layuan. (2005). Applied Research of EJB Component Technique in Electronic Commerce System. *Journal of Wuhan University of Technology, 29*(2),223-226.