[Alguliev, 2004] R.M. Alguliev, R.M. Aliguliev. Genetic approach to the optimum setting of tasks in the distributed system // Artificial intelligence. − 2004. − № 4. − Pp. 79-88.

[Harti, 1990] R.E. Harti. And global convergence proof for class of genetic algorithms. – Wien: Technische Universitaet, 1990. – 136 p.

Author′s information

Vitaliy Snytyuk – Taras Shevchenko National University of Kyiv, postdoctorate researcher of cybernetics faculty; Glushkov's av. 2, building 6, Kiev, Ukraine; e-mail: snytyuk@majar.com

Alexander Dghulay − Cherkasy Institute of Fire Safety, senior teacher; Onoprienko Str. 8, Cherkasy, Ukraine; e-mail: djulaj@ukr.net

INFORMATION TECHNOLOGIES OF THE DISTRIBUTED APPLICATIONS DESIGN

Safwan Al Salaimeh

Abstract: The questions of distributed systems development based on Java RMI, EJB and J2EE technologies and tools are rated. Here is brought the comparative analysis, which determines the domain of an expedient demand of the considered information technologies as applied to the concrete distributed applications requirements.

The Distributed Systems' Design Conception

Distributed system, from the information systems point of view, consists of several independent computing devices. The state of distribution concept shares and belongs to data warehouses, hardware and software in equal parts. The application program, whose program code, data warehouses and computing resources are distributed, is called a distributed application. The composite parts of a distributed application interacts between themselves using information telecommunication technologies of seven-level OSI model, where TCP/IP sockets used on lower levels and remote access tools like DCOM, CORBA, RMI on higher levels.

The next problems can appear during the distributed applications design [1-4]: *telecommunications reliability*-communicational tools must provide a reliable interaction between the objects which execute severs and clients functions; *interface dependence*- client and server interfaces have and influence on the queries and answers; *object activation*- during the application execution the necessity of server object activation by client's request can appear; *create and remove*- during the application execution the necessity of object creating and removing can appear; *transaction support*- the control of operations set which executes by the distributed application objects, if at least one operation of the current set cannot be executed then the rest of operations must be also cancelled.

The distributed applications are based on possibility to send objects from one application program to another and to allow an invocation by one application program of object methods, which are located in anther application program. The processing of user's interaction with the database in realized in four levels: web browser, database client and database server. The user's level contains web browser, which displays web pages and collects information for processing. The middle level contains web server and application server, which are the client programs for database. The lowest level contains database servers.

The Technologies of the Distributed Applications Design

The interest to distributed applications is explained by increased requirements to modern program tools. The major of them are:

- *application scalability*- the capability for an effective maintenance of any quantity of clients at the same time;
- *application reliability* to the client application errors and communication failures;

- *transactions reliability*- is the secure system junction in functioning process from one stable and authentic state to another;
- *long-term behavior*- the nonstop run for 24 hours in a week (24x7 run model);
- *high security level* of applications, which guarantees not only the access control to different data, but also guarding in all stages of functioning;
- *high application development speed, simplicity of maintenance* and *possibility to modify* them by the programmers of a medium qualification.

At present time, there are some known technologies for static and dynamic distributed applications realization, which meet the requirements, described above: socket programming, RPC (Remote Procedure Call), DCOM (Microsoft Distributed Component Object Model), CORBA (Common Object Request Broker Architecture) and Java RMI (Java Remote Method Invocation). At the same time the most important of them are the last three- DCOM, CORBA, Java RMI.

The DCOM technology is object-oriented which is supported by the next operation systems: Windows 98, Windows NT, Windows 2000, Sun Solaris, Digital UNIX, IBM MVS, etc. The most important merit of this technology consists in possibility to integrate the applications, realized in different programming systems.

The CORBA technology is the part of OMA (Object Management Architecture), which is developed to standardize the architecture and interaction interfaces of object-oriented applications. The interfaces between the CORBA-objects encode using a special language for interface definitions IDL (Interface Definition Language). Such interfaces can be realized on any language of applied programming and connected to CORBA-applications. In the context of standards it is proposed to connect CORBA-object with DCOM-objects through the special CORBA-DCOM bridges.

The Java RMI applications consist of client and server as usual. Some objects are created on server which can be transmitted through the network or methods which are declared shared to remote application calls. On the client side there are realized applications, which use remote objects. The distinguishing feature of RMI is possibility to transmit through the network 4either methods or objects. This feature provides, finally, mobility (portability) realization.

Today, the Java RMI and CORBA technologies are the most flexible and effective to create distributed applications. These technologies are relatives by their features. The major merit of CORBA is the IDL interface, which unifies communication tools between the applications and the interoperability with other applications. On the other hand, Java RMI is more flexible and powerful tool to distributed applications development using the Java platform, including possibility of mobile applications realization.

The Java RMI technology includes two constituents: Java language instrumental tools and remote method invocation (RMI) to Java-objects. The Java language tools let to create complex distributed network applications, which is possessed of high security level and reliability, realize object-oriented programming, integrated multithreading and platform independence. The RMI technology assigns a set of tools, which let to get access to remote object on server by the special stub-object methods calls.

The Java RMI specification shows:

1) to set a remote interfaces for classes whose methods can be called through the network;
2) to create stub-objects using a special compiler;
3) to get a full copy of remote object, not only the reference to it;
4) to transmit objects in such a way that their behaviors will it not be changed when they are transmitted to another virtual machine;
5) to register a server object in the RMI registry and to support that registry in accessible state with the help of special background process. The clients can access to this registry when they are looking for the needed objects;
6) to produce marshalling and de-marshalling with the help of API serialization which translate object to a byte stream before transmitting and then, after receiving, backwards;
7) to work over the IIOP protocol that gives a possibility to communicate with the CORBA-objects (IIOP protocol can transmit a data of a different types, including structures and unions, even if they contains reverse definitions).

The RMI technology organizes the programmable interface to work with the networks unlike the TCP sockets. Such an interface has higher level, it is based on method invocation and makes an impression allegedly the remote object is being operated locally. RMI is more suitable and more natural than interface based on sockets, however, it requires the Java programs execution on the both side of a connection. The network connection, nevertheless, may use the same TCP/IP protocol.

The Network Information Technologies

There are three main parts in a client-server technologies:

- *user interface* to display an information, realize graphical user interface and form a requests to server;
- *functional logic* to realize required computing and business rules;
- *database* to execute selections, modify data and process them in accordance with received commands.

Depending on ordering method of these components on client and server machines can be 2-tier, 3-tier and n-tier client-server technologies for corporate systems functioning.

The 3-tier technology with distributed services provides independent functional relationships between the user interface, functional logic and database. The user interface and the smaller part of a functional logic are located on client computers. The bigger part of a functional logic is located on application server. The database locates on database server. The applications are independent in the model with distributed services. They interact through the networks with application server.

The object -oriented network technologies of distributed services connect in themselves the models of a distributed databases and services. The software of such kind of systems consists of a set of object units which interact between themselves through the computer network using standard interfaces. This approach lets to use units many times and to spend computer resources more economically. Each object, depending on conditions, can be either client or server in this technology. The object computing architecture, which based on distributed network services, presents a new highly upcoming kind of computer technologies, which are widely uses to distributed corporate systems' design.

The object-oriented approach is perspective to create dynamic web-oriented applications. The problem of "super thin" clients' realization is closely tied with them, when the client-applied program uses the browser environment for execution. The effectiveness of such technology for corporate systems' design is described by the possibility to realize the standard HTML-browser in any operation environment. If we'll take into account that the maintenance cost for one server and the maintenance cost for a thousand of connected to it "thick" clients are not comparable then we can make a conclusion: successfully realized corporate system sharply reduces overhead charges for maintenance.

The Component-oriented Technologies

At present time, for the development of corporate systems with a distributed database branches and remote services (functional logic) the most considerable are integral technologies like an Active X/DCOM, RMI/CORBA, Enterprise Java Beans/CORBA and CORBA/J2EE. The distributed component-oriented Active X/DCOM technology is intended for application server development, registration and management by the distributed program objects. The main its demerits are: in the first place, server computer operates under Windows NT/2000 Server operation system only; and in the second place, the tools of multi-user access for a several servers' operations coordination which is managing by transactions are completely absent.

The remote access technology RMI organizes interconnection with a remote objects, lets to develop a qualitative Internet-applications, possesses by all Java language environment merits, provides an object-oriented programming, guarantees a high security and reliability level, multithreading, multiplatform support, an independence to operation system. The RMI built on the concept of call and its parameters translation to a byte stream for transmits them through the network. The backward operation is carried out on the server side, method invocation and result transmission back to the client.

In the base of the common object request broker architecture lies an idea of uniting multivinder applications in a common tooling environment, than lets to this technology operate in heterogeneous systems, i.e. the network can serve, at the same time, computers of a different types, operating under different operation systems and serving applications written on different programming languages. The CORBA technology provides development

and maintenance of software for distributed corporate systems with a maximal convenience. Its software is called CORBA-application.

Each object requests broker ORB producer, which, practically, is the CORBA, theoretically can propose its own protocol of transport service for data transmission. In this case the expanded name of technology, for example CORBA/IIOP, which reflects the name of the IIOP network protocol. With the aid of IIOP protocol any CORBA/IIOP application can interact with other CORBA/IIOP-applications independent of hardware, software and operation systems producers.

Conclusion

The realized scientific research and practically developed works certificate that in the near future the most perspective are object-oriented Java RMI technologies and component-oriented EJB-and J2EE-technologies for the distributed corporate systems design. The EJB-and J2EE-technologies application is expedient in systems, based in using a powerful DBMS as the systems kernel. For a example, it can be the e-commerce systems, banking systems, business platforms and e-shops.

Bibliography

1- Mike Morgan, Java 2 for Professional Developers, Sams Publishing. 1999, 720 P.;

2- Eckel B, Thinking in Java, Prentice Hall, 2000, 880p.;

3- Perrone P., Chaganti V., Building Java Enterprise Systems with J2EE, Sams Publishing. 2000, 1184P.;

4- Riccardi G., Princeples of Database Systems with Internet and Java Applications, Addison-Wesley Publishing Company, 2001, 480 P.

Author's Information

Dr. Safwan AI Salaimeh – Assistant Professor, Department of Computer Science, Irbid National University, Jordan

USE OF DYNAMIC TECHNOLOGIES FOR WEB-ENABLED DATABASE MANAGEMENT SYSTEMS

Galina Bogdanova, Todor Todorov, Dimitar Blagoev, Mirena Todorova

Abstract: In this paper we consider two computer systems and the dynamic Web technologies they are using. Different contemporary dynamic web technologies are described in details and their advantages and disadvantages have been shown. Specific applications are developed, clinic and studying systems, and their programming models are described. Finally we implement these two applications in the students education process: Online studying has been tested in the Technical University – Varna, Web based clinic system has been used for practical education of the students in the Medical College - Sofia, branch V. Tarnovo

Keywords: Computer systems and Web technologies, Databases, PHP, JSP, education.

ACM Classification Keywords: H.2.4 Systems, H.4 Inforamtion Systems Applications, J.3 Life and Medical Sciences, K.3.1 Computer Uses in Education

Introduction

For some time past World Wide Web is migrating to active websites, which allows to send to the user pages, configured according to his/her own requirements and offer more dynamic experience with browsers. Such sites are created with the help of a combination of programming languages and technologies, which can be used at once, or one by one.