

Technology of the Coordinator's Construction for Distributed Transactions in the Web Service Conditions

Ali Mohammed Saleh Ahmed
University of Diyala
Internet and Computer Center
dr.alisaleh80@gmail.com

Firas Mohammed Aswad
University of Diyala
College of Basic Education
altae13@yahoo.com

Recived : 23\11\2017

Revised : 3\12\2017

Accepted : 6\12\2017

Available online : 27/1/2018

DOI: 10.29304/jqcm.2018.10.1.360

Abstract:

The article is devoted to the transactions management in the heterogeneous distributed information environments is considered and the construction tool of the transaction coordinator on the basis of the scripting LUA language in the conditions of Web service is offered.

The LUA scripting language, in context of creation of LTM transaction coordinator using the modified query language LQL, has sufficient tools available in modern scenarios programming languages, When determining the library of LTM transaction coordinator built in conditions of LUA machine, using modernized scripting LQL.

Keywords: LQL language, LUA language, Database Management (DBMS), CORBA, IS, web-service.

Introduction

The priorities of company's activity change day by day: companies are being reorganized, break new ground and expand applications capacity, for example, developing a new web services. Under these changes, an organization faces the problem of effective and cost-beneficial resources provision to build them in line with priorities. It requires arriving at a solution, which allows the access to the informational structure at the time and location where there is a need for it, by implementing and

optimizing access to information, regardless of its physical structure. Thus, there is a need for information integration across the distributed environment, either within a single server or in a heterogeneous IT environment, for a number of independent systems or a combination of several solutions.

1- Providing access to information structure

For the solution of this problem, we are considering the integration process of all databases into a single service database, while the actual location of the considered database remains unchanged. At the creation of this structure information remains in its initial source where, it is being modified. Most data sources are a single integrated virtual database, masking the number and types of DBMS differences. An integrated solution for service DBMSs allows accessing directly the data not only of the most famous DB providers, such as Linter, Paradox, Clipper, MS SQL Server, MUMPS, Oracle, but also of modern DBMS vendors, without determination of the actual physical location of data from a web service.

As there is no need to duplicate any information into a service web-DBMS, this integration provides the most rapid way to information integration. This integration also configures a supporting zone of such information that exists in legacy web-applications, or in applications that require a local owner and simply cannot be consolidated. When using such an approach of integration of heterogeneous distributed DBMS two concepts arise: local transactions and global transactions⁽³⁾.

Division of transactions of general type into local and global appeared in connection with the application of the distributed multilink systems. Local transactions in case of a mismatch with service DBMS are known as transactions controlled by databases. From the point of view of a software support, which uses universal API, to access a database the local transactions are generated and terminated in conjunction with a certain call to DBMS. Usually, the open management of local transactions is implemented by the call of such methods as "to save transaction in DB" or "to roll back transaction from a DB". This mode of operation is typical for building applications of "client-server" architecture. If the transaction termination procedure is implemented in a single command, then DBMS uses a single phase mode of transaction termination. With such approach, there is no way to create a transaction, which "joints" actions using several connections simultaneously.

In "client-server" architecture the implementation of procedure of single-phase local transactions use is possible until no distributed DBMS is used. At the moment of such architecture, the client application code creates a connection to DBMS, a function is created either implicitly or explicitly, that begins a local transaction. Then selection of data, and as a result, completion of transaction is made. If it was the last, and all the necessary actions have been executed, the connection with the DBMS breaks.

If a distributed DBMS is used that defines the implementation of data change procedure in several DBMSs in one transaction context — so the possibility to form a universal transaction shall be provided for this task solution, to be executed for several databases of this type. This implementation procedure is conventionally called "two-phase commit local transactions".

2- Implementation of Transaction Coordinator

Requirements enhancement to difficult projects design technologies, to increase of their reliability and scalability level, as well as to capacities of a creation of heterogeneous RDBMS, have led to changes of the situation as follows: complete complex technologies of real time distributed systems design emerged. The most famous and perfect ones are CORBA, J2EE and .NET. Previously, the definition of the transaction for a RDBMS was extended to an entire IS, which is a small part of such system. Not only information content in long-term storages became transactional — final objects, of which all information system consists, became the same.

Owing to such system development, a task of personal operation emerged at level of provided distributed object transactions and transactions at level of heterogeneous databases ⁽²⁾. To operate such object transactions in multilink information systems, there is a special component, which is usually called the transaction coordinator. In the developed information system the transaction coordinator is realized on the basis of the scripting programming language LUA, which allows solving a problem of transaction model with two-phase end. This programming language has been choose because of its ease of use, simple syntax, and thus sufficient power: the language supports objects, meta-tables make its type ultimately flexible and use of API-functions allows excellent integration, increase of scripts and expansion of basic language capabilities. LUA can be used together with "client-server" applications, written in various high level programming languages, including the support of web languages, such as PHP, Perl, Python, Ruby, ASP.NET, and Groovy. The considered language of the transaction coordinator is simple in training, thanks to extension possibility with help of C-functions, can be applied to wide range of tasks, using object-oriented programming mechanisms with support of prototypes.

3- Information system setup phases

When determining the library of LTM transaction coordinator built in conditions of LUA machine, using modernized scripting LQL language, there are several actions to set the system work. The sequence of these actions is given below⁽⁷⁾.

3.1. Loading of the main configuration file. In the directory of executable software a search of a file named conf/ORB.conf shall be done. Then its values are read to system buffer by special system tools.

3.2. Initialization of ORB space. At this stage a stack of necessary libraries omniORB 4.3 is being compiled. Further, using the loaded settings, the initialization method omniORB is called, which in turn prepares the CORBA environment for interaction with the general information system of web-application.

3.3. Initialization of the LUA machine. As the LUA-language interpreter is required, the need of loading of main concepts into memory arises, that accomplishes by methods of calling lua_open () function from LUA-libraries stack. This operation creates a new state and returns the index. In case of memory lack as a result of work the zero index will be defined.

3.4. Loading of necessary libraries. Considering item 3 in which process of creation of a new independent status of LUA for which to be made loading of standard alternative libraries, such as functions for operation with lines, mathematics, additional functions of debugging, the function for operation with an operating system constructed in the scripting LUA language or on another JAVA the compatible to the general loader provided interpreter was realized.

3.5. Creation of a container of alias. At this stage, the procedure of creation of an object of a special class, which is developed especially, is made and contains a full range of alias to the DBMS, is engaged in their creation, deleting and search in system. The object of the considered container registers unique a method in LUA-machine. The challenge of the provided instruction is necessary for addition of references to the distributed databases the considered information system in a container.

3.6. Registration of external instructions in the LUA machine. In considering this step alone, because of the need to register previously prepared operation for the interpreter. After this step there is a procedure of modernization of standard LUA language in the LQL language.

3.7. Loading of alias file. The file an alias is presented in the form of the list of descriptors on remote DBMS. As a result of which accomplishment in the interpreter, an object of a container of references will contain a set of objects of alias when which using the appeal to required components of a general information system is created. If to consider the class diagram for an object of an alias - we define that a base class for it is LQLObject, which realizes functionality on fixing external and additional instructions in the LUA interpreter. Further, using parent methods, an object of the class LQLAlias creates own global meta-data sheet with a unique identifier (name) comprising instructions of the virtual machine for access to the distributed DBMS. The metastable represents the usual table in LUA in which admissible operations over value are presented.

One of the global classes, responsible for the operation of information systems in the framework of the task is Alias class that provides interaction with the environment as follows: it encapsulates an object reference to the CORBA skeletons HSystem.DriverManager and HSystem.Connection. Using CORBA object references of the classes there is a transaction in SQL DBMS with which the currently set direct connection. This mechanism is implemented in the framework of Operation AHas: query (), which by means of delegated LQLAlias class for further registration in the LUA interpreter. After the end of the considered process the unique reference to an object to become available in the language formation of inquiries.

3.8. Creation of the environment of performance transactions. The constructed runtime environment of transactions is intended for storage of running states of each local transaction in the conditions of a distributed query to DBMS which at the time of execution of a distributed query to an information system of real time monitors operation of each operation in the interpreter. As soon as in the conditions of search the instruction of accomplishment of a local request to RDBMS meets, the environment of accomplishment places complete subject to the local transaction in the container. If there are errors during the execution of a distributed query - rolls back all transactions are in the container. In the absence of error situations - records every transaction in your database, for later execution.

3.9. In case of successful implementation of each earlier considered stage (item 1-p.8) forming a global object of the class LQLGlobal which is an internal class of the global LQL environment is made. Using dynamic conversion to the ILQLGlobal interface, the pointer on the recorded object returns to the web application. If the ORB initialization phase, the machine initialization LUA error, the information model built throws an exception, which sets the rules saying that from now on will not operate the library, respectively, web-application specifies a null pointer.

The Results

The article deals with the problems of transaction management in heterogeneous distributed IT environments and offers means of building transaction coordinator based on LUA scripting language in a Web-service.

The following results were obtained:

1. The architecture of web-service applications for multiprocessor and cluster solutions that are scalable and provides high performance when inter-module interaction in the DBMS.
2. Hierarchical model of access to data and internal information exchange in a web-service interface that provides rapid generation of data based on queries to the DBMS information environment.
3. The concept of the transaction was extended to information systems in general, not just to DBMS as a part of such systems.
4. This solution provides enhanced operational efficiency allowing several applications to work jointly.

Conclusion

Thus, the LUA scripting language, in context of creation of LTM transaction coordinator using the modified query language LQL, has sufficient tools available in modern scenarios programming languages: mathematical transactions completion, managing structures, iterator and standard libraries for string objects handling, return and data collection. All this absolutely meets requirements of creating the considered information system.

References

- [1] V.L. Burkovskiy, A.N. Dorofeev, S.V. Semin, Simulation and algorithmization of control of heterogeneous databases in the distributed information systems [Text] // Voronezh, VSTU, (2003) – p.71.
- [2] S.A. Rikov, V.L. Burkovskij, A.A. Golikov Control of the heterogeneous distributed objects of information systems of real time // Voronezh: VSTU, (2012) – p.193.
- [3] Rikov S.A. Mathematical and software information systems of real time with the heterogeneous environment of the distributed DBMS – Voronezh: VSTU, (2011) – p.185.
- [4] Lokshin, M.V. Technique of receiving time estimates of execution of requests in the parallel DBMS with data replication [Text] // Control systems and information technologies. – (2014). -№ 4. – p.41-44.
- [5] Ritter D. The Middelware Muddle. DBMS magazine, May, (1998), p. 15.
- [6] Rykov SA, AA Golikov, Burkovsky VL Implementation LQL query language for interaction with a homogeneous medium // Electrotechnical complexes and control systems, Voronezh, H.2, (2010).-p.53-55.
- [7] Golikov AA Rykov SA Lomov EO, Burkovsky VL "Building a Distributed Transaction Coordinator in a heterogeneous environment based on LUA» // Journal of Vilnius Gediminas Technical University, v.5, №4,(2009).- p.202-203.
- [8] Ozhan G., Dogac A., Kilic E., Ozcan F., Nural S., Dengi C., Halici U., Aspinar B., Koksall P., Mancuhan S., Evrendilec C. Making Oracle7, Sybase and Adabas Interoperable through CORBA: MIND Project. In Proc. Of European Oracle User Group Conference, Amsterdam, April (1996) p. 1047-1058.
- [9] Pitoura E., Bukhres O., Elmagarmid A. Object Orientation in Multidatabase Systems. Report CSD-TR-93-084, Department of Computer Science, Purdue University.
- [10] Pons J., Vilarem J. Mixed concurrency control: Dialing with heterogeneity in distributed database systems. In Proceedings of the Fourteenth International VLDB Conference, August (1988), Los Angeles p.144.
- [11] Lokshin MV Study of parallel execution of SQL-queries with advanced data caching // Modern Information Problems in Economics and Safety: Proceedings of the XVIII International Open Science Conference, USA, Mississippi, Lorman, "Science Book Publishing House", in January (2013), p. 69-72.

تكنولوجيا بناء منسق للمعاملات الموزعة في شروط خدمة الويب

علي محمد صالح احمد
جامعة ديالى
مركز الحاسبة والانترنت
dr.alisaleh80@gmail.com

فiras محمد اسود
جامعة ديالى
كلية التربية الاساسية
altae13@yahoo.com

المستخلص :

هدف البحث هو إدارة المعاملات في بيئات المعلومات الموزعة غير المتجانسة ويعتبر أداة البناء المنسق على أساس لغة البرمجة **LUA** في ظروف او شروط خدمة الانترنت.

لغة البرمجة **LUA** في سياق إنشاء منسق المعاملات **LTM** باستخدام لغة الاستعلام المعدلة **LQL**، والتي لديها أدوات كافية متاحة في سيناريوهات لغات البرمجة الحديثة ، وعند تحديد مكتبة لمنسق المعاملات **LTM** بنيت في ظروف **LUA machine**، وذلك باستخدام تحديث البرمجة **LQL**.

ان أولويات نشاط الشركات تتغير يوماً بعد يوم حيث يتم إعادة تنظيم الشركات، وكسر أرضية جديدة، وتوسيع قدرات التطبيقات، على سبيل المثال، من خلال تطوير خدمات الويب الجديدة. وفي ظل هذه التغييرات، تواجه المنظمة مشكلة توفير موارد فعالة ومفيدة من حيث التكلفة لبنائها بما يتفق مع الأولويات. يتطلب التوصل إلى حل يسمح بالوصول إلى البنية الإعلامية في الوقت والمكان تكون هناك حاجة إليها، من خلال تنفيذ وتحسين الوصول إلى المعلومات، بغض النظر عن هيكلها المادي. وبالتالي، هناك حاجة إلى تكامل المعلومات عبر البنية الموزعة، سواء داخل خادم واحد أو في بيئة تكنولوجيا المعلومات غير المتجانسة، لعدد من الأنظمة المستقلة أو مزيج من عدة حلول.

الكلمات المفتاحية: لغة **LQL**، لغة **LUA**، نظام ادارة قاعدة البيانات، **CORBA**، **IS**، خدمة الويب.