

ARHITECTURA DISTRIBUITĂ A BAZELOR DE DATE ÎN ORACLE

Adrian Runceanu, *Universitatea
Constantin Brancusi, Tg-Jiu,
ROMANIA*

Mihaela Runceanu, *Colegiul
National Ecaterina Teodoroiu, Tg-
Jiu, ROMANIA*

Marian Popescu, *Universitatea
Constantin Brancusi, Tg-Jiu,
ROMANIA*

REZUMAT: În această lucrare prezentăm arhitectura distribuită a serverului *ORACLE*, împreună cu câteva facilități ale acestui SGBD ce pot fi utilizate de către administratorul bazei de date în monitorizarea activității și funcționării *Oracle Database*.

CUVINTE CHEIE: baze de date distribuite, server de baze de date, client, monitorizare.

1. BAZE DE DATE DISTRIBUITE

O bază de date distribuită este o mulțime de baze memorate pe mai multe calculatoare, care apare, din punct de vedere al aplicațiilor ca o singură bază de date.

Astfel o aplicație poate accesa simultan și modifica datele din câteva baze de date aflate în rețea. Fiecare bază de date Oracle din sistem este controlată de propriul server local Oracle, dar cooperează pentru a menține consistența bazei de date distribuite globale.

A DISTRIBUTED DATABASES ARCHITECTURE IN ORACLE

Adrian Runceanu, *Constantin
Brancusi University of Targu-Jiu,
ROMANIA*

Mihaela Runceanu, *Ecaterina
Teodoroiu High College of Tg-Jiu,
ROMANIA*

Marian Popescu, *Constantin
Brancusi University of Targu-Jiu,
ROMANIA*

ABSTRACT: In this paper we present distributed server Oracle architecture, along with some features of the DBMS that can be used by the database administrator to monitoring activity and operation of Oracle Database.

KEYWORDS: distributed databases, database server, client monitoring.

1. DISTRIBUTED DATABASES

A distributed database is a set of databases stored on multiple computers, which is, in terms of applications as a single database. Such an application can access and modify data simultaneously from several databases in the network. Each Oracle database system is controlled by its own local Oracle server but cooperates to maintain the consistency of the global distributed database.

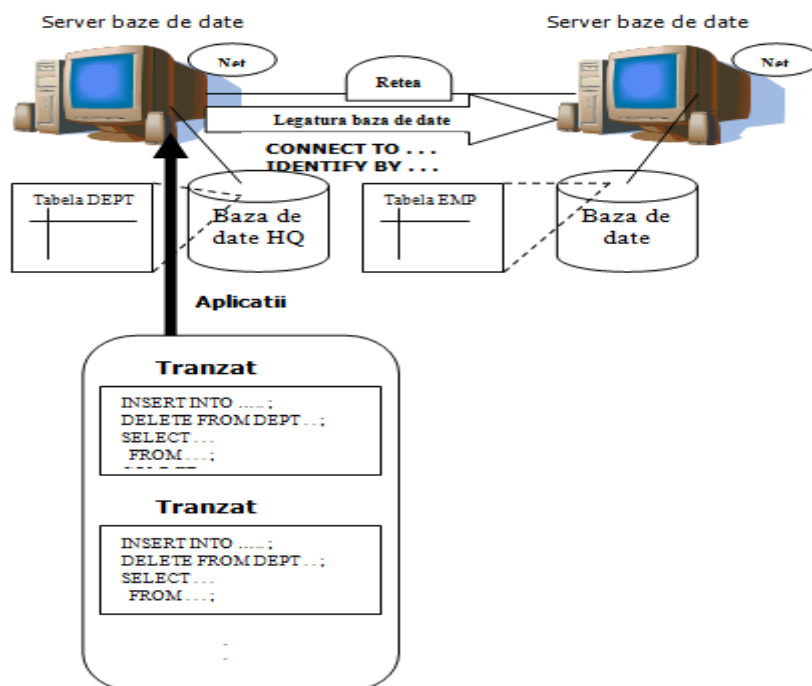


Fig. 1 Sistem reprezentativ de baze de date distribuite Oracle
 Fig. 1 Representative Oracle system of distributed database

Clienți și servere

Un *server* de baze de date este software-ul Oracle care gestionează baza de date, și un *client* este o aplicație care solicită informații de la server. Fiecare calculator dintr-un sistem este un nod. Un *nod* într-un sistem de baze de date distribuite se comportă ca un client, ca un server, sau ca amândouă, în funcție de situație. De exemplu, în figura 1, calculatorul care gestionează baza de date HQ se comportă ca un server când o instrucțiune se referă la datele sale locale (de exemplu, a doua instrucțiune din fiecare tranzație se referă la o interogare pe tabela locală DEPT), și se comportă ca un client atunci când o instrucțiune se referă la date de pe alte calculatoare (de exemplu, prima instrucțiune din fiecare tranzație se referă la tabela remote EMP din baza de date SALES).

Conexiune directă și indirectă

Un client se poate conecta direct sau indirect la un server de bază de date. În figura 2, când o aplicație client execută prima și a treia instrucțiune pentru fiecare tranzație, clientul este conectat direct la baza de date

Clients and servers

A database server is the software that manages Oracle database and a client is an application that requests information from the server. Each computer in a system is a node. A node in a distributed database system acts as client or as server, or both, depending on the situation. For example, in Figure 1, the computer that manages the HQ database acts as a server when a statement refers to its local data (for example, the second statement of each transaction refers to a local query table DEPT), and acts as a client when a statement refers to data from other computers (for example, the first instruction of each transaction refers to the remote table EMP in the SALES database).

Direct and indirect connection

A client can connect directly or indirectly to a database server. In figure 2, where a client application run the first and third statement for each transaction, the customer is directly connected to intermediate HQ database and indirectly to the SALES database that contains data remotely.

intermediară HQ și indirect la baza de date SALES care conține date remote.

Rețeaua

Pentru a lega baze de date individuale dintr-un sistem distribuit de baze de date, este necesară o rețea. Vom încerca să explicăm în continuare câteva aspecte despre sistemul distribuit Oracle.

Net8

Toate bazele de date Oracle dintr-un sistem distribuit utilizează software-ul Oracle de rețea – Net8 - pentru a facilita comunicația inter-baze de date de-a lungul rețelei. Așa cum Net8 conectează clienții și server-ele care operează pe calculatoare diferite dintr-o rețea, tot așa permite server-elor de baze de date să comunice de-a lungul rețelei pentru a putea efectua tranzacții distribuite și remote în bazele de date distribuite.

Net8 prezintă în mod transparent conectivitatea necesară transmiterii cererilor SQL și primirea datelor pentru aplicațiile care utilizează sistemul. Net8 preiau instrucțiuni SQL de la un client și le împachetează pentru transmisia lor la un server Oracle pe un protocol de comunicație standard sau o interfață programabilă. Net8 de asemenea preia răspunsurile de la server și le împachetează pentru a le transmite înapoi la clientul căruia îi sunt trimise. Net8 execută toate procesele independent de sistemul de operare al rețelei.

Denumirile Oracle

Opțional, rețeaua Oracle poate utiliza denumirile Oracle pentru a furniza sistemului un serviciu global de directoare. Când o rețea Oracle gestionează un sistem de baze de date distribuite, se pot utiliza serverele de denumiri Oracle pentru a poziționa informații despre fiecare bază de date din sistem astfel încât să se ușureze configurația de acces la bazele de date distribuite.

Baze de date și legăturile bazelor de date

Fiecare bază de date dintr-un sistem distribuit este diferită de toate celelalte baze de date din sistem și are *denumirea sa globală*. Oracle formează o denumire

Network

To link the individual databases in a distributed database system, a network is required. We will try to further explain some aspects of distributed Oracle system.

Net8

All Oracle databases in a distributed system use Oracle software network - Net8 - to facilitate inter-database communication over the network. As Net8 connects the clients and servers that operate on different computers from a network, so allows servers database to communicate over the network to perform distributed and remote transactions in the distributed databases. Net8 presents transparently the necessary connectivity to transmit SQL statements and receive data for applications that use the system. Net8 take SQL statements from a client and pack them for their transmission to an Oracle server on a communication protocol standard or programmable interface. Net8 also takes replies from the server and pack them for transmission back to the client to whom they are sent. Net8 performs all processes independent of the network operating system.

Oracle Names

Optionally, the network Oracle can use the labels Oracle to provide a global service directory system. When an Oracle network manages a distributed database system, you can use servers labeled Oracle to position information about each database in the system configuration so to facilitate the access to distributed databases.

Databases and database links

Each database in a distributed system is different from all other databases in the system and has its own global name. Oracle forms a global database name by prefixing the domain name database with a single database. For example, Figure 2 shows a hierarchical representation of databases over a network.

globală a bazei de date prin prefixarea domeniului bazei de date împreună cu denumirea individuală a bazei de date. De exemplu, figura 2 ilustrează o reprezentare ierarhică a bazelor de date de-a lungul unei rețele.

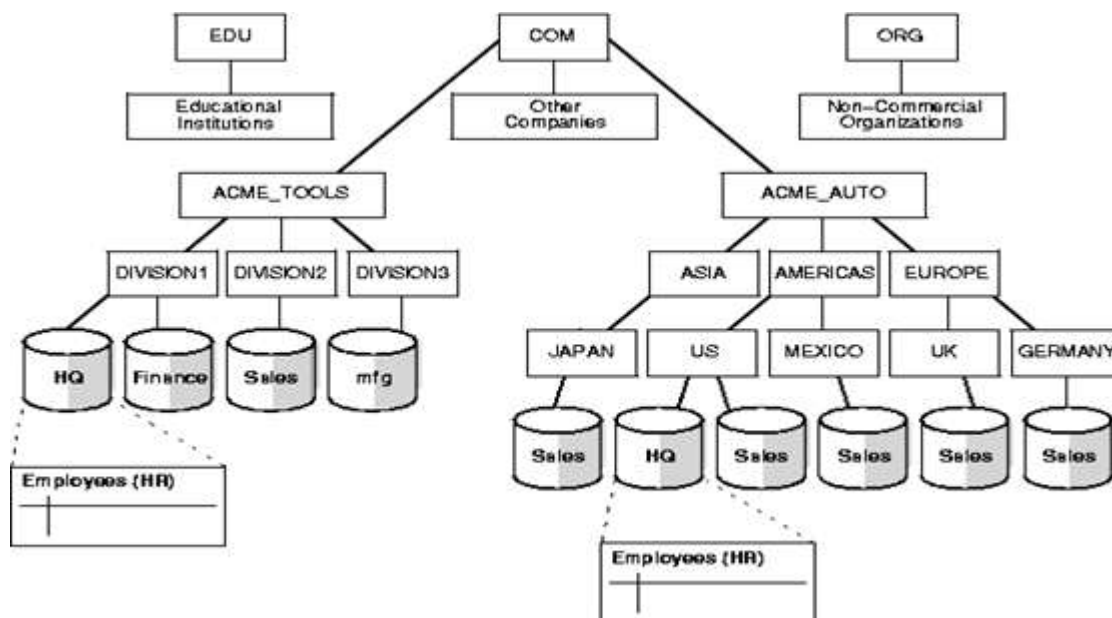


Fig. 2. Reprezentare ierarhică a bazelor de date de-a lungul unei rețele
 Fig. 2. Hierarchical representation of the database over a network

În timp ce câteva baze de date pot avea același nume individual, fiecare baza de date trebuie să aibă o denumire globală unică. De exemplu, domeniile de rețea US.AMERICAS.ACME_AUTO.COM și UK.EUROPE.ACME_AUTO.COM conțin fiecare o bază de date VÂNZĂRI.

VANZARI.US.AMERICAS.ACME_AUTO.COM
VANZARI.UK.EUROPE.ACME_AUTO.COM

Pentru a facilita interogările aplicațiilor într-un sistem de baze de date distribuit, utilizatorii Oracle folosesc legături ale bazelor de date. O legătură a bazei de date definește o cale unidirecțională de comunicație de la o bază de date Oracle la altă bază de date.

Legăturile bazelor de date sunt transparente utilizatorului unui sistem de baze de date

While some databases may have the same name individually, each database must have a unique global name. For example, network domains and US.AMERICAS.ACME_AUTO.COM and UK.EUROPE.ACME_AUTO.COM each contain a SALES database.

VANZARI.US.AMERICAS.ACME_AUTO.COM
VANZARI.UK.EUROPE.ACME_AUTO.COM

Database connection

A database link defines a one-way communication path from an Oracle database to another database. Database links are transparent for the user of the Oracle distributed database system as the database link name is the same as the global name of the database that connects. For example, the following SQL statement creates a database

distribuit Oracle, deoarece numele legăturii bazei de date este același ca și denumirea globală a bazei de date către care face legătură.

De exemplu, următoarea instrucțiune SQL creează o legătură de bază de date în baza de date locală care descrie calea către baza de date îndepărtată:

```
VANZARI.US.AMERICAS.ACME_AUT  
O.COM  
CREATE DATABASE LINK  
vanzari.us.americas.acme_auto.com . . . ;
```

După crearea legăturii bazei de date, aplicațiile conectate la baza de date locală pot accesa datele din baza de date îndepărtată

```
VANZARI.US.AMERICAS.ACME_AUT  
O.COM.
```

Oracle permite câteva tipuri de legături de baze de date.

Denumirea schemei obiect

Pentru a rezolva referiri la schemele obiect (un proces numit rezoluția numelui) Oracle formează nume de obiecte utilizând abordarea ierarhică.

De exemplu, cu o singură bază de date Oracle garantează că fiecare schemă are un nume unic, și împreună cu schema, fiecare obiect are un nume unic. Ca rezultat, numele schemei obiect este întotdeauna unic împreună cu baza de date. Mai mult, Oracle poate rezolva ușor referințele unei aplicații la numele local al obiectului.

Într-o bază de date distribuită, o schemă obiect cum ar fi o tabelă este accesibilă la toate aplicațiile din sistem. Oracle extinde modelul de denumiri ierarhice ale denumirilor globale de baze de date pentru a crea efectiv denumiri globale de obiecte și a rezolva referirile la obiectele schemei într-un sistem distribuit de baze de date. De exemplu, o interogare se poate referi la o tabelă îndepărtată prin specificarea denumirii complete, incluzând și baza de date în care se află :

connection in the local database that describes the path to the remote database:

```
VANZARI.US.AMERICAS.ACME_AUT  
O.COM
```

CREATE DATABASE LINK

```
vanzari.us.americas.acme_auto.com . . . ;  
After creating the database link, the applications connected to the local database can access data from the remote database  
VANZARI.US.AMERICAS.ACME_AUT  
O.COM.
```

Oracle allows some types of database links.

Name of object scheme

To resolve the references to object schemes (a process called name resolution) Oracle forms object names using the hierarchical approach.

For example, a single Oracle database ensures that each scheme has a unique name, and together with the scheme, each object has a unique name. As a result, the object schema name is always unique together with the database. Moreover, Oracle can easily solve the local references of an application to the local name of the object.

In a distributed database, a schema object such as a table is accessible to all applications in the system. Oracle extends the hierarchical naming model of global database names to effectively create global object names and resolve references to objects in a distributed system scheme database. For example, a query can refer to a remote table by specifying the full name, including the database :

```
SELECT * FROM  
scott.emp@vanzari.us.americas.acme_aut  
o.com;
```

To complete the application, the database server uses the default local database link that connects to remote SALES database.

The names of global databases and global names of objects

**SELECT * FROM
scott.emp@vanzari.us.americas.acme_aut
o.com;**

Pentru a completa cererea, server-ul bazei de date locale utilizează implicit o legătură de baze de date care se conectează la baza de date îndepărtată VANZARI.

Denumirile globale ale bazelor de date și denumirile globale ale obiectelor

Într-un sistem distribuit de baze de date, fiecare bază de date trebuie să aibă un nume global unic. Numele global al bazei de date identifică fiecare baza de date din sistem.

Numele global al bazei de date constă din doua componente:

- Un nume al bazei de date de opt caractere sau mai puțin (de exemplu VANZARI)
- Un nume de domeniu care conține baza de date respectivă.

Numele de domeniu al denumirii globale a bazei de date trebuie să urmeze convențiile standard ale Internet-ului. Nivelele în denumirea domeniului trebuie separate prin puncte și ordinea numelor de domeniu este de la frunză la rădăcină, de la stânga la dreapta. Numele bazei de date și numele domeniului sunt determinate de inițializarea parametrilor DB_NAME și DB_DOMAIN.

O legătură de bază de date trebuie să dea același nume ca și numele global al bazei de date îndepărtate la care face referire. Când se setează parametrul de inițializare GLOBAL_NAME la TRUE, atunci Oracle asigură că numele legăturii bazei de date este același cu numele bazei de date globale a bazei de date îndepărtate.

Oracle utilizează următoarea denumire a schemei globale obiect:

<schema>.<schema_obiect>@<numele_global_al_bazei_de_date>

unde

<schema> - o schema este o colecție de structuri logice de date, sau scheme de obiecte. O schemă este proprietate a unui singur utilizator al bazei de date și are același nume ca și utilizatorul.

In a distributed database system, each database must have a unique global name. Global database name identifies each database in the system.

Global database name consists of two components:

- A database name of eight characters or less (eg sale)
- A domain name that contains the relevant database.

The domain name of the global database name must follow standard Internet conventions.

The levels in domain name must be separated by dots and the order domain names is from leaf to root, from left to right. Database name and domain name are determined by the initialization parameter DB_NAME and DB_DOMAIN.

A database link should give the same name as the global name of the remote database to which it refers. When set to TRUE GLOBAL_NAME initialization parameter, then Oracle ensures that the database link name is the same global database name of the remote database. Oracle uses the following global object naming scheme:

**<schema>.<schema_obiect> @
<numele_global_al_bazei_de_date>**

Where

<schema> - a schema is a collection of logical data structures, or objects schemes. A schema is owned by a single user database and has the same name as the user.

<schema_obiect> - a schema object is a logical data structure such as a table, view, synonym, procedure, package or a database connection.

<numele_global_al_bazei_de_date> - the name that uniquely identifies the remote database. This name must be the same as the concatenation parameters for initialization of the remote database, DB_NAME and DB_DOMAIN.

<schema_obiect> - o schemă obiect este o structură logică de date cum ar fi o tabelă, vedere, sinonim, procedură, pachet sau o legătură a bazei de date.

<numele_global_al_bazei_de_date> - numele care identifică unic baza de date îndepărtată. Acest nume trebuie să fie același ca și concatenarea parametrilor de inițializare a bazei de date îndepărtate, DB_NAME și DB_DOMAIN.

2. Monitorizarea performanțelor bazelor de date în Oracle

Fiind prima bază de date proiectată pentru calcul în grid, *Oracle Database* aduce îmbunătățiri semnificative față de predecesoarele sale printr-o performanță superioară, scalabilitate, disponibilitate mărită. De asemenea securitatea a fost îmbunătățită iar utilizarea pe dispozitive de stocare și servere standardizate a devenit mai ușoară și la un pret mai redus. *Oracle Database* se poate instala pe diferite servere de la cele mici și calculatoare personale până la servere mari și clustere.

Monitorizarea funcționării Oracle cuprinde o paletă largă de activități, de la urmărirea transferului de date cu hard disk-ul și administrarea memoriei utilizate, până la controlarea lucrului în rețea. Este de asemenea treaba administratorului de baze de date să constate atunci când codul furnizat de programatori nu funcționează în limitele de performanță impuse și să ceară o revizuire a acestuia.

Memoria ocupată de Oracle se împarte într-o zonă comună denumită *System Global Area (SGA)*, care la rândul ei este împărțită în mai multe buffere, și mai multe zone aferente altor procese. Dimensiunea memoriei ocupate de Oracle precum și împărțirea acesteia între diferitele buffere aparținând SGA are o influență mare asupra performanțelor sistemului. Nu există o formulă prestabilită pentru dimensionarea lor optimă, aceasta tinând de aplicația implementată. Până la apariția Oracle 10g

2. PERFORMANCE MONITORING IN ORACLE DATABASES

As the first database designed for grid computing, Oracle Database brings significant improvements to its predecessors through superior performance, scalability, increased availability. Also the security has been improved and the using on storage devices and has become easier and at a lower price. Oracle Database can be installed on different servers to small and personal computers to large servers and clusters.

Monitoring Oracle operation includes a wide range of activities, from tracking data transfer hard disk and memory management used to control networking. It is also the job of the database administrator to find when the code provided by developers doesn't work within the performance requirements and to request a review of it.

Memory usage of Oracle is divided into a common area called the System Global Area (SGA), which in turn is divided into several buffers, and several other processes related areas. Memory size occupied by Oracle and its division between the various SGA buffers belonging has a large influence on system performance. There is no predetermined formula for their optimal size, it depends of the implemented application. Until the advent of Oracle 10g sizing was done by the database administrator. *Sga_max_size* initialization parameter expressed the maximum size that could be occupied by SGA. By the Oracle 10g the system is self-regulating. Sizing SGA and their various caches belonging is automatic. *Sga_target* initialization parameter is set dynamically and is lower than specified by the administrator *sga_max_size*.

dimensionarea se făcea de administratorul bazei de date. Parametrul de initializare *sga_max_size* exprima dimensiunea maximă care putea fi ocupată de SGA. La Oracle 10g sistemul se autoreglează. Dimensionarea SGA și a diferitelor cache-uri aparținând acestora se face automat. Parametrul de initializare *sga_target* se reglează dinamic și este mai mic decât *sga_max_size* specificat de administrator.

2.1. Monitorizarea de la distanță a bazelor de date Oracle

Una dintre soluțiile Oracle de monitorizare activă de la distanță este denumită RPM (*Remote Preemptive Monitor*) datorită utilizării tehnicilor sistemelor expert care formează un strat de protecție “preemptivă” în scopul asigurării disponibilității continue a tuturor bazelor de date Oracle folosite în aplicații critice [22]. Soluția RPM reprezintă un adevărat sistem informatic de asistare a deciziei, DSS (*Decision Support System*) ce folosește tehnologia sistemelor expert pentru a automatiza componentele bine structurate ale monitorizării active, cu mențiunea că, totuși, este necesară prezenta unui expert uman pentru a diagnostica și rezolva o problemă ce deranjează buna funcționare a tuturor aspectelor specifice bazelor de date Oracle (server, disc și rețea). Componentele soluțiilor RPM sunt prezentate în figura 3.

- ✓ Componenta RPM *AllertPack* informează administratorul despre problemele apărute la baza de date Oracle înainte să fie deranjați utilizatorii finali. Ca exemple, se pot menționa alertele serverului de tipul *CPU Overload Alert*, *RAM Memory Overload Alert* și *Network Latency Alert* sau alertele interne ale bazei de date Oracle (*Replication backlog alert*, *Broken Job alert*, *Hot File Reads alert* etc.).
- ✓ Componenta RPM *ServerPack* monitorizează sistemul de operare pentru a avertiza asupra problemelor apărute la resursele serverului ce pot afecta

2.1. REMOTE MONITORING OF ORACLE DATABASE

One of Oracle solutions remote active monitoring is called RPM (*Remote Preemptive Monitor*) through the use of expert systems techniques to form a protective layer of “preemptive” in order to ensure continuous availability of all Oracle databases used in critical applications [22]. RPM is a real solution to assist decision system, DSS (*Decision Support System*) that uses expert system technology to automate the monitoring components of structured assets, provided that, however, the human expert is required to diagnose and solve a problem that disturbs the proper functioning of all aspects of Oracle database specific (server, disk and network). Components of RPM solutions are presented in Figure 3.

- ✓ Component RPM *AllertPack* inform the administrator about problems with the Oracle database before being disturbed the end users. As examples, we can mention the type of alert server *CPU Overload Alert*, *Alert and RAM Memory Overload Alert* *Network Latency* or internal alerts Oracle Database (*Replication Backlog alert*, *Broken Job alert*, alert, etc. *Hot File reads*.).
- ✓ *ServerPack* RPM component monitors the operating system to warn of problems with server resources that can affect disponibilitatea Oracle system.
- ✓ *AuditPack* RPM component allows for quick and accurate capacity planning details or server disk resources.
- ✓ *ManagementPack* RPM component provides a complete and thorough planning the details of important capabilities, including the total size of the database and increasing trends in all key tables of the database.

- disponibilitatea sistemului Oracle.
- ✓ Componenta *RPM AuditPack* permite obținerea rapidă și precisă a detaliilor de planificare a capacității resurselor discului sau serverului.
 - ✓ Componenta *RPM ManagementPack* asigură un raport complet și amănunțit asupra detaliilor de planificare a capacităților importante, inclusiv dimensiunea totală a bazei de date și tendințele crescătoare ale tuturor tabelor-cheie ale bazei de date.

Componenta *WISE Monitoring* asigură un instrument inteligent de analiză previzională pentru identificarea tendințelor ascunse de performanță și pentru a justifica reconfigurările imediate ale bazei de date.

Component Monitoring WISE provides intelligent analysis tool to identify trends hidden forecasting performance and to justify immediate configuration of database.

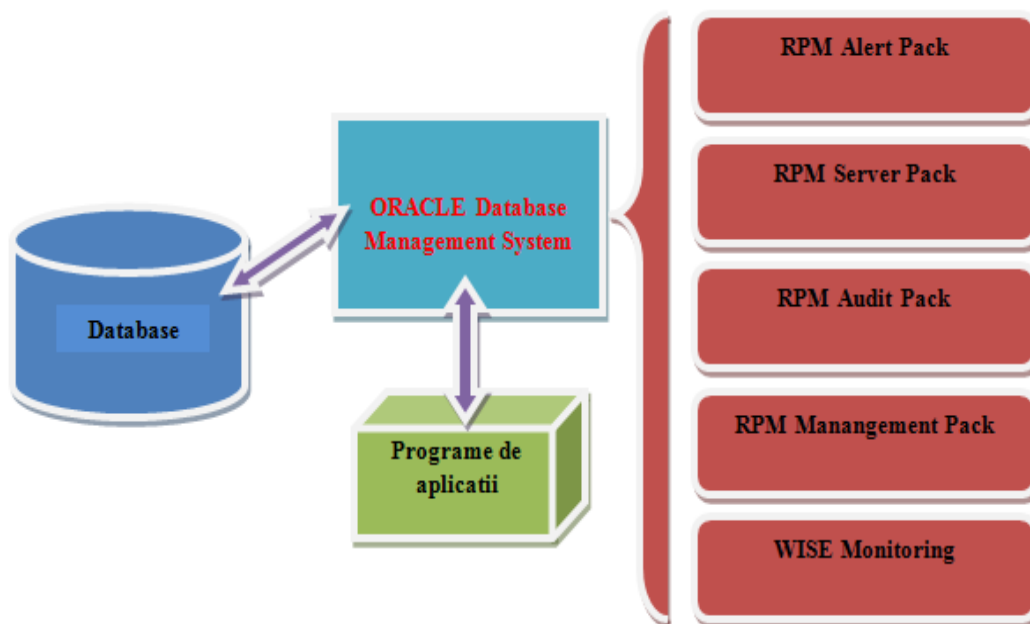


Fig. 3. Componentele RPM ORACLE

Fig. 3. ORACLE RPM components

3. CONCLUZIE

În această lucrare am trecut în revistă componentele arhitecturii ORACLE pentru baze de date distribuite și instrumentele de care dispune administratorul bazei de date (*Enterprise Manager, RPM Oracle*), pentru localizarea și semnalizarea posibilele probleme. Astfel, dezvoltatorii de software

3. CONCLUSION

In this paper we reviewed the components of Oracle architecture for distributed databases and tools available to the database administrator (*Enterprise Manager, Oracle RPM*) for location and signal potential problems. Thus, software developers can optimize the code so that the application

pot să optimizeze codul pentru ca aplicația să funcționeze la parametri optimi. could work better.

BIBLIOGRAFIE

- [1] Runceanu A., Marian Popescu, EvalTool for Evaluate the Partitioning Scheme of Distributed Databases (2009), International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE 2009) Conference, Conference Proceedings book, December 4-12, 2009, University of Brigeport, USA, Volume 1 - Technological Developments in Networking, Education and Automation (Elleithy, Sobh, Iskander, Karim, Mahmood), SCSS 09. (<http://cisse2009.org/cisse2009.aspx>) published in Volume 1 – CISSE 2009 Proceedings: Technological Developments in Networking, Education and Automation, ISBN: 978-90-481-9150-5
- [2] Runceanu A., Towards vertical fragmentation in distributed databases, Proceedings International Symposium on Systems Theory, Automation, Robotics, Computers, Informatics, Electronics and Instrumentation - SINTES 13, Craiova, 18-20 octombrie 2007, Vol II, pg. 80-85, Editura Universitaria, ISBN 978-973-742-816-5.
- [3] Runceanu A., Runceanu M., Dinca A., About fragmentation in distributed object-oriented databases, The XIIth Edition of the International Scientific Conference of the Engineering Faculty, 23-24 november 2007, Annals of Constatin Brancusi University of Targu-Jiu, nr.3/2007, pg.267-272.
- [4] Runceanu A., Popescu M. An implementation for database partitioning, Proceedings of 9th International Carpathian Control Conference - ICC 2008 (may 25-28), Sinaia, ISBN 978-973-746-897-0, pg. 559-563.
- [5] Runceanu A., Fragmentation in distributed databases, International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE

REFERENCES

- [1] Runceanu A., Marian Popescu, EvalTool for Evaluate the Partitioning Scheme of Distributed Databases (2009), International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE 2009) Conference, Conference Proceedings book, December 4-12, 2009, University of Brigeport, USA, Volume 1 - Technological Developments in Networking, Education and Automation (Elleithy, Sobh, Iskander, Karim, Mahmood), SCSS 09. (<http://cisse2009.org/cisse2009.aspx>) published in Volume 1 – CISSE 2009 Proceedings: Technological Developments in Networking, Education and Automation, ISBN: 978-90-481-9150-5
- [2] Runceanu A., Towards vertical fragmentation in distributed databases, Proceedings International Symposium on Systems Theory, Automation, Robotics, Computers, Informatics, Electronics and Instrumentation - SINTES 13, Craiova, 18-20 octombrie 2007, Vol II, pg. 80-85, Editura Universitaria, ISBN 978-973-742-816-5.
- [3] Runceanu A., Runceanu M., Dinca A., About fragmentation in distributed object-oriented databases, The XIIth Edition of the International Scientific Conference of the Engineering Faculty, 23-24 november 2007, Annals of Constatin Brancusi University of Targu-Jiu, nr.3/2007, pg.267-272.
- [4] Runceanu A., Popescu M. An implementation for database partitioning, Proceedings of 9th International Carpathian Control Conference - ICC 2008 (may 25-28), Sinaia, ISBN 978-973-746-897-0, pg. 559-563.
- [5] Runceanu A., Fragmentation in distributed databases, International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE 2007) Conference, Conference Proceedings

- 2007) Conference, Conference Proceedings book, December 3-12, 2007, University of Brigeport, USA, publish in *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, Springer Science, ISBN 978-1-4020-8734-9 (Print) 978-1-4020-8735-6 (Online), DOI 10.1007/978-1-4020-8735-6_12. (<http://cisse2007.org/cisse2007.aspx>)
- [6] ***, [_ebook_-_pdf_-_programming_oracle_9i_admin_guide](#).
- [7] ***, 9i Database Concept.
- [8] ***, [Introduction_to_oracle_9i_PL-SQL_Student_Guide](#).
- [9] ***, eBook[1].Prentice_Hall_PTROracle_DBA_Guide_to_Data_Warehousing.
- [10] ***, [Oracle9i_DBA_Fundamentals_II_-_Volume_1_-_Student_Guide](#).
- [11] ***, ORACLE_9i_Sample_schemas.
- [12] ***, [TestKing_1z0-031_-_Oracle_9i_Database_Fundamentals_I_v4](#).
- [13] ***, Oracle 8 Distributed Database Systems.
- [14] ***, Oracle 9i - Mcgraw Hill - The Complete Reference.
- [15] ***, Oracle Press - Oracle 9i SQL Reference (2001) - (a90125) - by Lorentz.
- [16] ***, O'Reilly - Oracle - SQL Programming.
- [17] ***, Oracle - Introduction to SQL.
- [18] ***, Oracle 11gR1, Oracle Corporation.
- [19] ***, <http://www.oracle.com/technology/products/database/oracle11g/index.html>.
- [20] ***, [http://www.oracle.com/technology/deploy/ilm/pdf/ILM_for Business_11g.pdf](http://www.oracle.com/technology/deploy/ilm/pdf/ILM_for_Business_11g.pdf).
- [21] ***, <http://www.oracle.com/technology/products/bi/db/11g/pdf/partitioning-11g-datasheet.pdf>.
- [22] <http://www.oracle.com>
- book, December 3-12, 2007, University of Brigeport, USA, publish in *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, Springer Science, ISBN 978-1-4020-8734-9 (Print) 978-1-4020-8735-6 (Online), DOI 10.1007/978-1-4020-8735-6_12. (<http://cisse2007.org/cisse2007.aspx>)
- [6] ***, [_ebook_-_pdf_-_programming_oracle_9i_admin_guide](#).
- [7] ***, 9i Database Concept.
- [8] ***, [Introduction_to_oracle_9i_PL-SQL_Student_Guide](#).
- [9] ***, eBook[1].Prentice_Hall_PTROracle_DBA_Guide_to_Data_Warehousing.
- [10] ***, [Oracle9i_DBA_Fundamentals_II_-_Volume_1_-_Student_Guide](#).
- [11] ***, ORACLE_9i_Sample_schemas.
- [12] ***, [TestKing_1z0-031_-_Oracle_9i_Database_Fundamentals_I_v4](#).
- [13] ***, Oracle 8 Distributed Database Systems.
- [14] ***, Oracle 9i - Mcgraw Hill - The Complete Reference.
- [15] ***, Oracle Press - Oracle 9i SQL Reference (2001) - (a90125) - by Lorentz.
- [16] ***, O'Reilly - Oracle - SQL Programming.
- [17] ***, Oracle - Introduction to SQL.
- [18] ***, Oracle 11gR1, Oracle Corporation.
- [19] ***, <http://www.oracle.com/technology/products/database/oracle11g/index.html>.
- [20] ***, [http://www.oracle.com/technology/deploy/ilm/pdf/ILM for Business_11g.pdf](http://www.oracle.com/technology/deploy/ilm/pdf/ILM_for_Business_11g.pdf).
- [21] ***, <http://www.oracle.com/technology/products/bi/db/11g/pdf/partitioning-11g-datasheet.pdf>.
- [22] <http://www.oracle.com>