

MODELS AND SOLUTIONS FOR THE IMPLEMENTATION OF DISTRIBUTED SYSTEMS

Ghencea Adrian

Titu Maiorescu University, Faculty of Economics

Vătuțiu Teodora

Titu Maiorescu University, Faculty of Economics

Țarcă Naiana

University of Oradea, Faculty of Economics

Software applications may have different degrees of complexity depending on the problems they try to solve and can integrate very complex elements that bring together functionality that sometimes are competing or conflicting. We can take for example a mobile communications system. Functionalities of such a system are difficult to understand, and they add to the non-functional requirements such as the use in practice, performance, cost, durability and security. The transition from local computer networks to cover large networks that allow millions of machines around the world at speeds exceeding one gigabit per second allowed universal access to data and design of applications that require simultaneous use of computing power of several interconnected systems. The result of these technologies has enabled the evolution from centralized to distributed systems that connect a large number of computers. To enable the exploitation of the advantages of distributed systems one had developed software and communications tools that have enabled the implementation of distributed processing of complex solutions. The objective of this document is to present all the hardware, software and communication tools, closely related to the possibility of their application in integrated social and economic level as a result of globalization and the evolution of e-society. These objectives and national priorities are based on current needs and realities of Romanian society, while being consistent with the requirements of Romania's European orientation towards the knowledge society, strengthening the information society, the target goal representing the accomplishment of e-Romania, with its strategic e-government component. Achieving this objective repositions Romania and gives an advantage for sustainable growth, positive international image, rapid convergence in Europe, inclusion and strengthening areas of high competence, in line with Europe 2020, launched by the European Council in June 2010.

Keywords: information society, databases, distributed systems, e-society, implementation of distributed systems

Cod JEL: O33, M15, L86

Introduction

The concept of "information society" is a very generous program that covers all sectors of government. The main goal is to create a society that includes all citizens' access to public services provided in electronic form, by increasing the capacity to use information society services, reforming the government operational models and increasing operational efficiency through appropriate use of information and communication technologies and the increase of competitiveness of the business through advanced use of ICT - Information and communications technologies.

The complexity of software applications is an essential property that derives from: the complexity of the problem domain, managing the development process, the degree of flexibility allowed by the software and problems that characterize the behavior of discrete systems.

Since the last decades of the previous century growth rate has been calculated mainly by tricks, by dividing tasks in the computer system, by introducing devices interrupt requests from the input / output or direct memory access (Lungu, Vătuțiu and Fodor, 2006: 45). Then specialized systems for digital imaging systems appeared that sought to compensate for insufficient speed through parallel processing, allocating each pixel in an image line of each unit of computing - a processor dedicated local operations from the image. This is how the first configurations of parallel computing and the first parallel algorithms appeared. One aspect of distributed processing which has been subject to attention in recent years refers to an environment in which they can engage cycles and spare storage space of the CPU tens, hundreds and thousands of networked systems to work on a specific problem that involves a very high processing capacity. Developing these processing models was limited however by the lack of attractive broadband connections and problems, combined with the real challenges of security, management and standardization. A distributed system can be defined as a group of independent computers that the user perceives as a coherent unit.

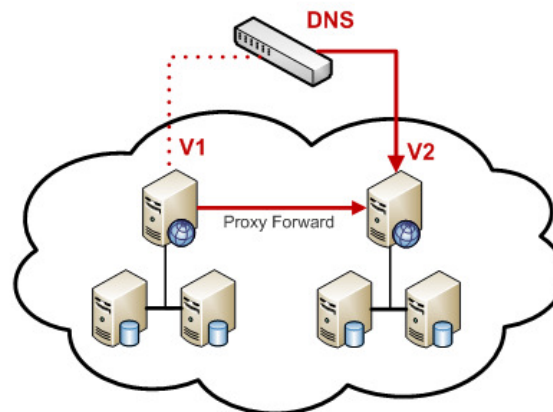


Fig. 1 Distributed storage and query of arbitrary data

A distributed system must not be built only because it can be built. There are many different types of distributed processing systems and there are quite a few challenges to overcome to successfully design such a system. Distributed processing systems aim to connect users and resources in a transparent, open and scalable way (Vătuțiu and Popeangă, 2006: 134). It would be ideal if this system would show more tolerance for errors and also if it would be more powerful than any combination of independent computer systems.

1.Parallel processing systems

Parallel processing is the simultaneous execution of the same instructions (divided and especially adapted) on multiple processors in order to obtain faster results. The idea is based on the fact that the process of solving a problem can usually be divided into smaller instructions that can be achieved simultaneously if they are coordinated. A parallel processing system is a computer that is equipped with more than one processor for parallel processing (Lungu and Ghencea 2011:122). Also new multicore processors are parallel processing systems. There are several types of parallel processing computers. They are differentiated by the type of interconnection between processing (known as "processing elements"), between processors and memory. Besides the classification made by Flynn who consider the type of processor instructions executed, there is a classification that is based on how memory is constructed: parallel processing computers that have distributed memory have multiple processors which access all the available memory like an addressing global space; computers which processes in parallel and have distributed memory with multi-

processors, but each processor can only access local memory - no global space for memory access between them.

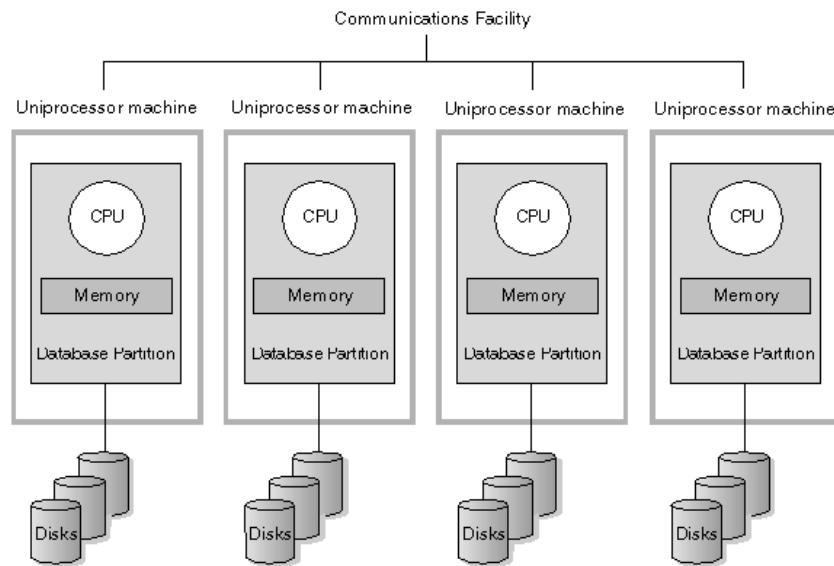


Fig. 2 Parallel processing systems

1.1.Implementation of parallel processing for different computer systems

Parallel processing, defining two or more processors that execute concurrent processes and acts as a single unit, allows one to run complex applications.

Most management systems relational databases are currently being upgraded to take advantage of parallel processing in heterogeneous systems, and to allow complex run of mission-critical applications. The optimal distribution of data is difficult in terms of technology; this process can be strongly dependent on the requirements for ensuring a good response to requests, and ensuring data integrity, continuous availability, interoperability, etc.

Management systems of modern databases use a series of abstract concepts and strategies to meet the requirements associated with current applications. The transaction, for example, can be used in the distributed data network to ensure the passage of groups of data and associated operations from a client post to server or from one server to another. Most of the producers of databases have monitors for the transaction process (TP - Transaction Processing) which represent advanced tools to manage distributed transactions in heterogeneous networks (Cristea 2007:44). The standard method of communication between TP monitors and database systems has been accepted as Protocol X / A as part of a group of standards X / Open. Currently, Sybase System 10 servers and Oracle 7 supports protocol X / A, while indirectly Informix Online adhere to this protocol through an ancillary product, Informix TP / XA.

In terms of ensuring data integrity in systems client / server distributed management software vendors have addressed these databases strategies:

- The "two-phase commit" technique: all changes required by a transaction on a database are either committed (the execution of the transaction is completed) or void, with the return of the database to the previous state. This strategy is not suitable for complex heterogeneous networks in which the probability of failure in any node is large and mission-critical systems. It is used as a way to ensure that all servers hold identical copies of the database at any time;

- The data replication strategy is now a solution adopted by Oracle, Sybase and Informix. Replication is a process in which multiple servers hold many identical copies of a database. The replication strategy differs significantly from the database "two phase commit" in that it guarantees the identity of copies of distributed databases only at certain times or under certain conditions. The data replication technique used by the Oracle Server 7 is called "Table Snapshots" by which the central server (master) copies at certain moments of time only those parts of the database that have changed, then disseminating these changes in the network. The replication mechanism used by the server is Informix database with the method "snapshots" used by Oracle. Informix uses the file "log" to "backup" to yield data from database tables to be replicated.

Replication servers are only the beginning of a whole generation of software that implements the abstract concepts related to data sharing in heterogeneous environments with advanced management and parallel processing, optimizing transactions.

1.2 Distributed Parallel Processing in Neural Networks

This type of processing is done using:

- Processing units: by analogy with the human brain, they correspond to neurons, and collectively to concepts such as character, features of pictures or objects in a PPP scheme. Individual units do not interpret the detailed design; they can be represented only by groups of units. We obtain a robust architecture that is not dependent on the efficiency of individual units and which assigns responsibility;
- Connections and the rule of activation;
- Internal input data (usually spread);
- External input data;
- Results obtained from processing units.

1.3.Open Distributed Processing

Open Distributed Processing (PDD) is the ISO standardization effort in the field of distributed processing. PDD 10 746 is the set of standards produced by ISO / IEC and ITU-T X.900 products (Telecommunication International Union). The notion of distributed processing seems very technical and complicated. However, with the development of large coverage networks, distributed processing is increasingly used. PDD's goal is to facilitate the use of distributed processing in areas as diverse as the wide deployment technology (Tannenbaum and Steen 2006: 241).

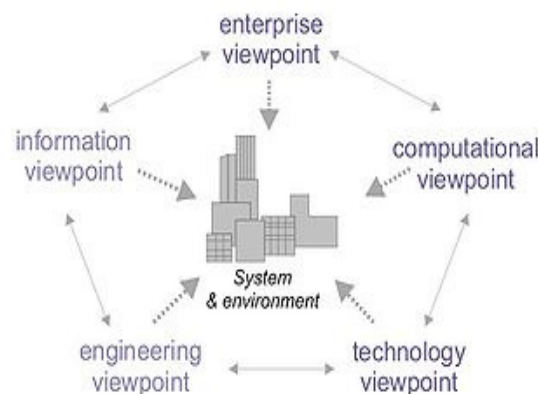


Fig. 3 Open distributed processing model

PDD is easy to use by programmers and operators because of its nature and means of distribution. In other words, both programming and use of distributed applications is presented as if that application would not be distributed. The way to get this perspective is transparent homogeneous PDD. Transparency provides users and developers with a consistent view of the network system as in the case when a message can traverse several different networks, without the user knowing the details of this process; distributed processing may involve different areas controlled by different authorities and equipment heterogeneous with very different hardware and software.

In general, the difference between distributed and network solutions is that when working in a network, the user is aware that the system runs on multiple machines, while in case of distributed processing the system appears as a single entity.

2. Implementation of Distributed Systems Solutions

2.1 Company Informix, symmetric multiprocessing

Informix Company has made a complete redesign of the system or introduced incentives for symmetric multiprocessing (SMP). Informix Online, is a dynamically scalable architecture (DSA). In situations in which the Informix server used version 6.0 of parallel processing, performance increased in comparison with older architectures. Informix Server version 6.0 of parallelism is allowed only on certain operations (create indexes, sorting, backup, data recovery) and runs on some platforms of symmetric multiprocessing (SMP). -Online/DSA Informix server (Dynamic Scalable Architecture). Version 7.0 has a scalable architecture that can manage scans, unification (join) the parallel sorting and querying databases.

2.2. Oracle 2.2 - parallel processing technology

Oracle 7 parallel Server creates a good balance between the need for better management and a multiprocessing system. Both Informix and Sybase disks require special partitioning in parallel processing, unlike Oracle, which is regarded as superior in this regard. The Oracle 7, Version 7.1 automatically maintains multiple copies of the same data on multiple servers, which eliminates the need for intensive disk-partitioning labor. Parallel query options in the Oracle Server 7 allow machinery working in symmetric multiprocessing (SMP), in clustered multiprocessing and massively parallel processing (MP) to execute a single application on multiple processing units, thus providing an almost linear scalability with each processor added to the configuration.

2.3 Sybase - Build Monumentum

Sybase company made a major step in designing database systems from the new generation by launching a product with "multithreading" work features. Sybase Build Monumentum runs in several ways to control the execution on both platforms -Windows NT and UNIX platforms, while adding to the existing architectural features for working with object-oriented database. The multiprocessing on distributed networks is done however only from the "client" point of view. Build Monumentum has an administrator who manages the "multithreading" processes even if the host operating system normally does not support this.

2.4 - IBM DB2 Parallel Edition

IBM developed its system of relational database management, DB2/6000, in a parallel multi-processor architecture with hardware ranging from local networks and systems to single processor RISC System/6000 IBM Power Parallel Systems SPX. Parallel Server DB2 Parallel Edition database can efficiently handle very large databases using strategies of sharing data and applications on their parallel execution.

The DB2/6000 management system extends to support a "shared nothing" architecture selected for two reasons:

- It is a scalable architecture to the level of hundreds of processors;

- Provides high portability because it requires only one communication link between processors, so it can be ported on any platform.

In the "shared nothing" architecture with the parallel implementation of the management system DB2 in a network of machines RISC/6000 IBM, database storage is done in a network of processors that provide buffers, lock structures, files log and separate records for each process. This prevents competition on cache structure resulting from the fact that all processors share the same set of resources.

In the architecture with multiple processing nodes, for large databases, data placement becomes a complex problem and system administration can be difficult. The implementation of parallel systems using a DB2 requires a definition language and management tools necessary for data partitioning. DB2/6000 provides two important features for partitioning large database tables: the use of partitioning keys and groups of nodes for a database that contains several tables; the user who develops the application can define a partitioning key for each table.

Conclusions

The internal administration of distributed databases is demanding and generally difficult, because one has to ensure that:

- Distribution is transparent (invisible and unobtrusive) - users must be able to interact with the system as if it were a non-distributed one (monolithic);
- Transactions must also have a transparent structure (invisible and unobtrusive). The course of each transaction must maintain database integrity, despite the multiplicity of partitions. For this they are usually divided into several transactions, each of them working with only one partition.

These requirements lead to the harmonization of the smart development strategy for Romania, in line with the priorities of "Europe 2020". Europeans should have the chance to use services on networks that are equal to or better than those available in other countries. In this direction have been made following steps:

- In 2010: adoption of a Recommendation to encourage investment in next generation access networks (NGA).
- In 2011: Monitoring of the implementation of the NGA Recommendation by national regulatory authorities and Monitoring of NGA deployment and broadband competition in the Member States.
- In 2012: guidance on pricing and/or costing methodologies in national regulatory measures.

Bibliography

1. Cristea Valentin, *Algoritmi de prelucrare paralela*, Bucuresti, Matrix Rom, 2005
2. Frishberg Leo, *Architecture and User Experience (Part 8: A Form of Software Architecture?)*, Computer-Human Interaction Forum of Oregon, 2009
3. Lungu Ion, Ghencea Adrian, *Distributed databases. Security, consistency and their replication*, 16th IBIMA Conference Kuala Lumpur, Malaysia, 2011
4. Lungu I., Vătuțiu T, Fodor A. G., *Fragmentation solutions used in the projection of the distributed database systems*, Proceedings of the 6th International Conference "ELEKTRO 2006", pp. 44-48, Edis-Zilina University Publishers, 2006
5. Tannenbaum Andrew S., Steen Van Maarten, *Distributed systems: Principles and paradigms*, New Jersey, Pearson Prentice Hall, 2006
6. Vătuțiu Teodora, Popeangă Vasile, *Project risk management for migration from client/server distributed Oracle Database architecture to Web oriented centralized database architecture of The Ministry of Public Finance's level*, „Universitaria SIMPRO 2006”- Petroșani, 2006, ISSN 1842-4449, pag. 134.
7. http://ec.europa.eu/information_society/newsroom/cf/fiche-dae.cfm?action_id=203, accessed April 20, 2011