International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 3 (2011) pp. 586-593

Database Technology Analysis for the Support or Instant Information Retrieval

Milena Janakova[1]

[1] Department of Informatics, Silesian University in Opava, School of Business Administration in Karvina,
Univerzitní nam. 1934, 733 40 Karvina, Czech Republic
mija@opf.slu.cz

***Abstract*: This paper is centered on analyzing selected segments of database technology on the basis of practical resolution for the Oracle database system. The subject of investigation is the architecture of a database system. This solution is compared with the analysis of architecture adopted from the UNIX operating system. The links of two products are not random. Our modern information world requires competitive and fast access to information. This information is often managed with database systems in a hosting operating system environment. Both products must reflect the need of its users. One of the many preferences is an easy start and stop of the application. The given architecture influences available solutions. The analysis of the adopted architecture is realized by Petri Nets. The reason for selecting this method is monitoring similarities in both products with the possibility of a classic analysis.**

***Keywords*:** analysis, database and operating systems, information, knowledge, Petri Nets**.**

I. Introduction

Information needs are growing with links to navigation in the increasingly complex and coherent world. Internet and network applications bring new challenges, chances, expectations, and changes. These realities affect all organizations, firms, and societies. Within this context we must also think about very small companies, middle-sized firms, and of course, individuals and their families.

Information has a very important influence in all aspects of everyday life. Information is indispensable for our activities, decision-making, and provides an optimal combination of steps for adopting a resolution. Information is a goods and it offers the benefit of competitive advantage. Competitive advantage is a reality that differentiates firms and organizations, regardless of size. Competitive advantage is based on uniqueness, idea, innovation, communication, and an element of surprise.

The question is "where should we source unique, new, meaningful ideas as well as surprise?" One of the various solutions is inspiration by a given process or access. The advantage is the ability to share interest about selected topics with anyone located anywhere via the Internet in a multicultural environment. Another way of inspiration is intentionally seeking information for future processing, acquiring new knowledge and skills for innovative implementation of all information technologies. A good starting point is to use a known method for the analysis based on the object and multidimensional approach.

II. Wide Range of Data and Information

Whether this way will in any manner be linked to inspiration with the intent to seek information, a positive influence on the diversity of approach will exist in all societies. This influence uses growing information needs for navigating in this world, and the meaningful use of information. [3] A great deal of information is monitored for future expectations with respect to defined objectives and responsibilities. People's perception of information depends on quality, visual and graphic effects, ways of presentation, and timeliness.

Besides the availability of data on this topic, it is also important to consider the amount of information that data brings. The amount of information, "I(a)," of the given data "a" is based on the likelihood of occurrence for data "p(a)". If the data appears only sometimes, then it is more important to its occurrence:

$$p(a) > p(b) \Rightarrow I(a) < I(b) \text{ [13]} \qquad (1)$$

The benefit of information is not only regarded for future decision supporting. [18] Information and data can be sent to further recipients for analysis, confrontation, and moment of surprise. These activities are realized by available tools and kits with computer support. Both data and information are types of knowledge or mediums used to attain knowledge. [4] Users can work with much information, but the amount of processed data is not a measure of quality and success. Aspects of quality are important for data. For example, "the volume of digital data has increased over the previous year by 62% to 800 billion gigabytes", but only 25% of stored information is unique. [16]

Simply, we must work with relevant information. Relevant information can be obtained from an optimal application from a database system area. The optimal implementation of database technology uses knowledge and skills.

III. Maximum Possibilities and Knowledge

Large amounts of information must be processed by optimal hardware and software. It is estimated that in 2007 all common

computers in the world managed to process 6.4 × 1018 instructions per second. [7] In accordance with these needs, there is an increasing number of installed servers and applications such as database systems. For example, Google controls more than 2% of the world's servers, and it is estimated that they now own more than one million servers. [7]

The knowledge spectrum for useful implementation is wide. Every user has experience with storage, processing and other data and information distribution. Knowledge and skills are needed for various areas of business activities with the support of information technology. The Oracle database system offers a wide range of menus and submenus like links to web pages or classic SQL commands. [17] The key to understanding and orientation is dividing interests into specialized areas. This access is not unique. The same method is used in information technology products such as operating systems with layers.

For its administration, the Oracle database system has specialized web pages, offered by the Oracle Enterprise Manager. This environment is divided into a few pages with required information. These pages are Home, Performance, Availability, Server, Schema, Data Movement, Software, and Support. Important pages are dedicated to:

- general information like the host CPU, active sessions, diagnostic summary,
- setup, manage, and secure backup,
- storage, database configurations, scheduler, security, query optimizer,
- database objects, programs, materialized views, user-defined types, and
- software configuration, database software patching.

The maximum quality of both knowledge and skills draws from many realized exercises, experiments, and practical methods. The important aspect is time and intensity of the education process. Great helpers are video recordings and analyses using the models. This merger brings synergy in the form of visual map of existing solutions. These methods are also useful for further changes within database technology on the basis of actual needs. The installed products often do not respect specific user preferences and need to be adapted or adjusted, which lies in the ideal combination of available functions, commands, menus, and links. These options need to be addressed in the design of information technology products. In this context, user preferences are difficult to estimate: an "average" user does not exist [14], but regardless of the preferences, the product must properly respond to given inputs to prevent errors and uncontrolled situations.

The changes are necessitated by the development of user preferences, needs, and the existing environment in which they operate. Database systems are no exception and they also need new challenges, changes, and analysis to be able to offer an efficient, dynamic, and user-friendly environment.

IV. Database Technology Analysis

Database technology offers many of the procedures which are performed by processes. Every database system has needed processes and files that must be initiated. Processes of the database system are divided into user and server processes. Existing Grid technologies [1], [10] influence Oracle database architecture that is created by:

- instance

If a user runs an application, a user process makes a connection with an instance. An Oracle database system instance is created by the main memory structure called SGA (System Global Area), and processes running in the background. This background is made up of a few specific processes such as System monitor, Process monitor, Database writer, and Log writer. System monitor carries out required activities for database recovery after crashes. Process monitor is centered on monitoring running processes, and handles support for their activities. Database writer stores data to disk. Log writer stores records for rollback transactions. Instance controls work with data and users of the database system. The database must be associated with an instance.

- database files

After starting, Oracle database system reads an SPFILE or initial parametric file with the aim to define values of needed parameters, to allocate the SGA memory, and to start background processes. The instance must seek and open database control files. They have information about required names of databases and redo log files. This procedure is called mount database. The database is in the ready state at this point and it must be opened for the users. In next step, the database opens online data and redo log files. Users can carry out needed activities like storing, editing, and searching information.

- Oracle Client and Oracle NET

Needed utilities offer a connection to the database, establish network sessions, and transfer data between client and server. The important configuration files are sqlnet.ora, tnsnames.ora, and listener.ora.

Experienced users easily create a script for to start all components of the database system from the operating system. A model Oracle architecture is shown in Fig. 1. An analysis is displayed in Tab. 1 with an incidence matrix and reachable marking for first four markings.

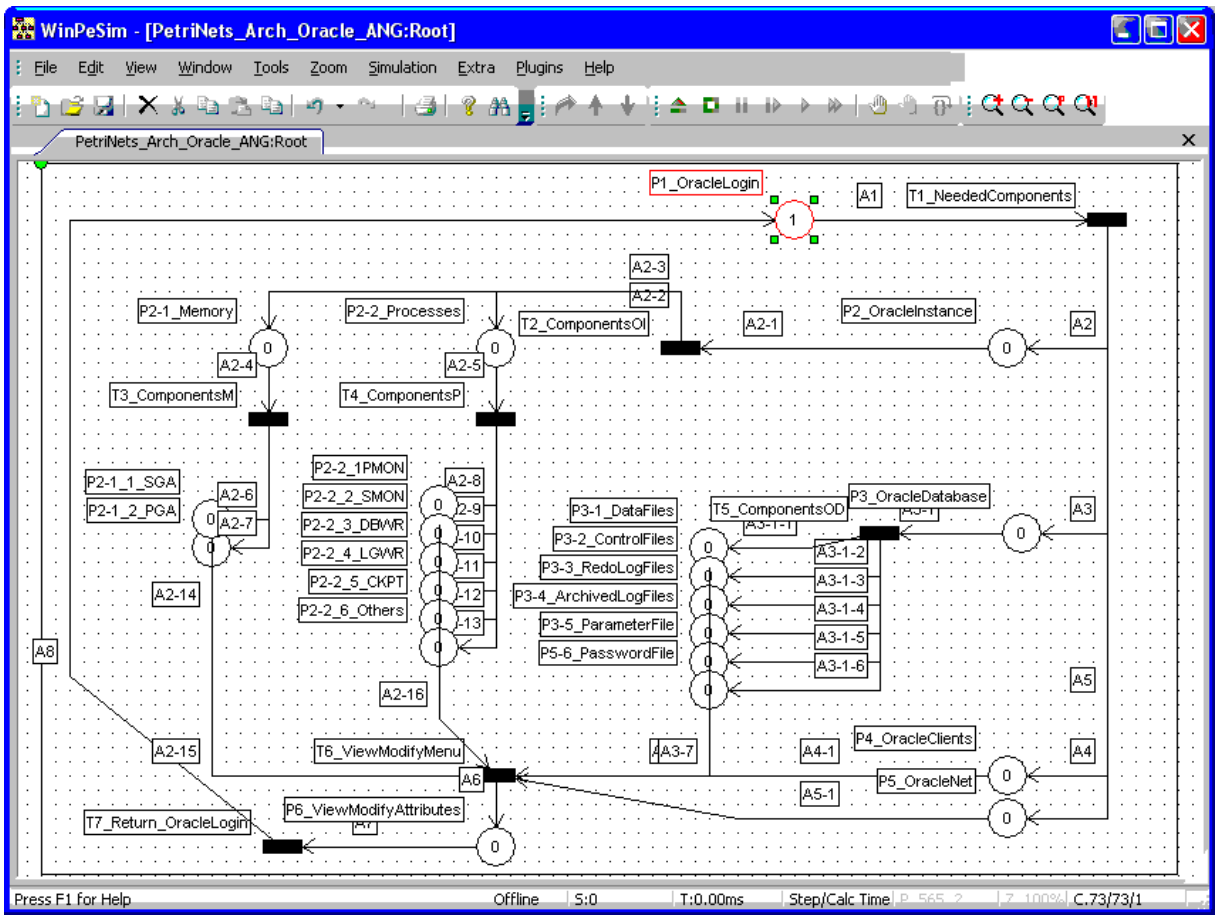

Figure 1. A model describing Oracle architecture

	Incidence matrix							t1→M1	t2→M2	t3→M3	t4→…
	t1	t2	t3	t4	t5	t6	t7	M0	M1	M2	M3
p1	-1	0	0	0	0	0	1	1	0	0	0
p2	1	-1	0	0	0	0	0	0	1	0	0
p2-1	0	1	-1	0	0	0	0	0	0	1	0
p2-1/1	0	0	1	0	0	-1	0	0	0	0	1
p2-1/2	0	0	1	0	0	-1	0	0	0	0	1
p2-2	0	1	0	-1	0	0	0	0	0	1	0
p2-2/1	0	0	0	1	0	-1	0	0	0	0	0
p2-2/2	0	0	0	1	0	-1	0	0	0	0	0
p2-2/3	0	0	0	1	0	-1	0	0	0	0	0
p2-2/4	0	0	0	1	0	-1	0	0	0	0	0
p2-2/5	0	0	0	1	0	-1	0	0	0	0	0
p2-2/6	0	0	0	1	0	-1	0	0	0	0	0
p3	1	0	0	0	-1	0	0	0	1	0	0
p3-1	0	0	0	0	1	-1	0	0	0	0	0
p3-2	0	0	0	0	1	-1	0	0	0	0	0
p3-3	0	0	0	0	1	-1	0	0	0	0	0
p3-4	0	0	0	0	1	-1	0	0	0	0	0
p3-5	0	0	0	0	1	-1	0	0	0	0	0
p3-6	0	0	0	0	1	-1	0	0	0	0	0
p4	1	0	0	0	0	-1	0	0	1	0	0
p5	1	0	0	0	0	-1	0	0	1	0	0
p6	0	0	0	0	0	1	-1	0	0	0	0

Table 1. Analysis for a database system model with an incidence matrix and reachable marking

The above-shown model build follows the defined places:

- P1_OracleLogin – place for standard login into system with all components; this place is the starting point.
- P2_OracleInstance – place displays the Oracle instance for correct work with database.
- P2-1_Memory, P2-2_Processes – places that have each instance (memory allocation and background processes).
- P3_OracleDatabase – places for storing needed information into database files; Oracle has a few files for various uses.
- P3-1_ – P3-6_ – items to specify database files.
- P4_OracleClients – accesses information for user processes.
- P5_OracleNet – accesses information to Oracle Net Server-Side and Client-Side.
- P6_ViewModifyAttributes – place for accessing available parameters to existing components, for example via Oracle Enterprise Manager.

Required transitions of the defined model are:

- T1_NeededComponents – Oracle database start defined database items. These items must be running to correct database activities.
- T2_ComponentsOI – database uses SPFILE or initial parametric file to define values of needed parameters, to allocate the SGA memory, and to start background processes.
- T3_ComponentsM – database allocates memory structures like PGA and SGA.
- T4_ComponentsP – databases create background processes for the Oracle database.
- T5_ComponentOD – database opens control, data, and redo log files. Other files are archived log files, password and parameter files.
- T6_ViewModifyMenu – accesses information about Oracle database.
- T7_Return_OracleLogin – returns to start point P1.

The validity of the defined model is verified by starting the given simulation. A route cycle is built from place P1 via specified transitions and places. Places P2 – P5 create standard components of the Oracle database system. Places P2-1 – P2-2 create items Oracle instance, places P3-1 – P3-6 make standard structure database files. The next route returns to place P1.

The available solution of the Oracle database system is very sophisticated and consistent. Users rely on stability and security of the database system. The architecture of every database system is one of the basic components of this stability and security. The question is “Is this unique solution presented without any room for improvement?” For the sake of comparison, an architectural analysis will be conducted of the UNIX operating system as openSolaris.

V. Operating System Analysis

The users of the UNIX operating system are working in the multiuser mode. Each mode is represented by a set of processes which the “init” process creates and oversees their status. The “init” process is the ancestor of every process in the operating system. The process is located in the operating system entity, created by calling the kernel fork. [2] Every process has assigned attributes such as UID, PID, PPID, C, TIME, TTY, TIME, and CMD. The UNIX operating system architecture is based upon:

- Needed processes

When the operating system (kernel) is started, the swapper process (sched, PID=0) is there for support. This process is a special process that is created automatically when the boot begins. Its purpose is to improve the maintenance of memory management. Afterward, the child of the “init” process (PID=1) is created via the call fork. The “init” process allows any user to access the system, creating a process that secures a standard input and output terminal. This process is called “getty” and cache memory on the established file systems), or lpsched (provides printer spooling).

- Important configuration files, scripts, and directory

The key file /etc/inittab and the rc* scripts (usually in /sbin, /etc) are located in the booting process. Init process successively reads each record of the /etc/inittab file and runs the command lines that correspond to the mode number of the operating system on which the operating system passes. Another init process runs the commands contained in the rc*. Names and locations of these scripts vary according to the implementation of the operating system. Other important files are rc.config, vfstab, or passwd, group from directory /etc.

These files contain information about computer settings, mount file systems, users of accounts and groups. Users easily run or stop an operating system from the console. To start an operating system, commands such as boot, init, or B are used. The operating system ensures the start-up of all needed processes for a given mode. The operating system is introduced into various modes of operation, i.e. single user, multiuser mode, halt, console, or user-defined modes. The most useful is the multiuser mode for standard activities of all users with the help of hosted applications like database systems. The benefit is running all needed processes via one command.

A model of UNIX architecture is viewed in Fig. 2. A classic analysis is displayed in Tab. 2 with an incidence matrix and reachable marking.

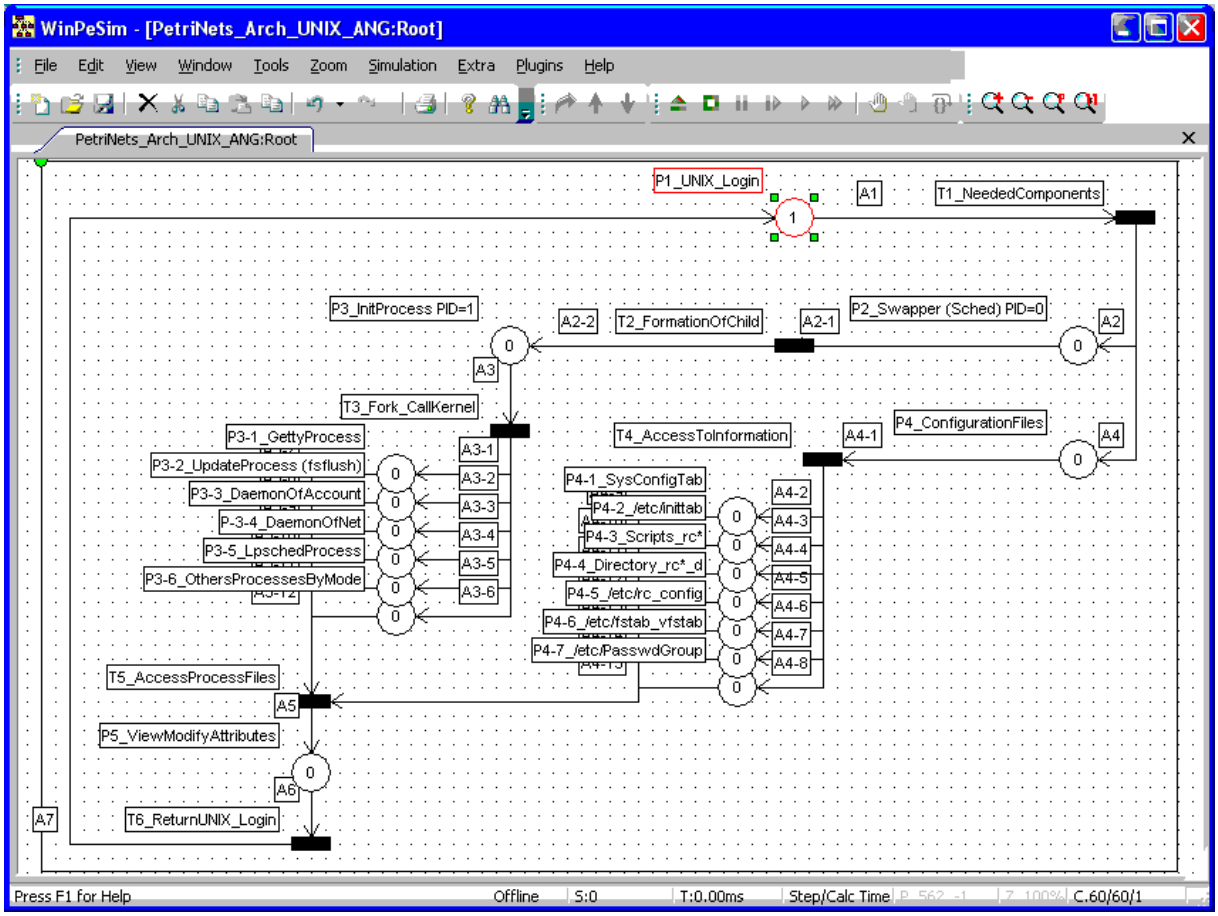

Figure 2. A model describing UNIX architecture

	Incidence matrix						t1→M1	t2→M2	t3→M3	t4→M4	t5→M5	t6→M0
	t1	t2	t3	t4	t5	t6	M0	M1	M2	M3	M4	M5
p1	-1	0	0	0	0	1	1	0	0	0	0	0
p2	1	-1	0	0	0	0	0	1	0	0	0	0
p3	0	1	-1	0	0	0	0	0	1	0	0	0
p3-1	0	0	1	0	-1	0	0	0	0	1	0	0
p3-2	0	0	1	0	-1	0	0	0	0	1	0	0
p3-3	0	1	1	0	-1	0	0	0	0	1	0	0
p3-4	0	0	1	0	-1	0	0	0	0	1	0	0
p3-5	0	0	1	0	-1	0	0	0	0	1	0	0
p3-6	0	0	1	0	-1	0	0	0	0	1	0	0
p4	1	0	0	-1	0	0	0	1	0	0	0	0
p4-1	0	0	0	1	-1	0	0	0	0	0	1	0
p4-2	0	0	0	1	-1	0	0	0	0	0	1	0
p4-3	0	0	0	1	-1	0	0	0	0	0	1	0
p4-4	0	0	0	1	-1	0	0	0	0	0	1	0
p4-5	0	0	0	1	-1	0	0	0	0	0	1	0
p4-6	0	0	0	1	-1	0	0	0	0	0	1	0
p4-7	0	0	0	1	-1	0	0	0	0	0	1	0
p5	0	0	0	0	1	-1	0	0	0	0	0	1

Table 2. Analysis for an operating system model with an incidence matrix and reachable marking

The above-shown model build follows the defined places:

- P1_UNIX_Login – place for standard login to a system with all components; this place is the starting point.
- P2_Swapper (Sched) PID=0 – place displays the running swapper process for booting the system. This process creates the next "init" process.
- P3_InitProcess PID=1 – places where the "init" process is created.
- P3-1 – P3-6 – items specifying needed processes of the multiuser mode.
- P4_ConfigurationFiles – place contains links to needed configuration and system files.
- P4-1 – P4-7 – accesses information about main configuration files and scripts like /etc/inittab, rc.config, vfstab, passwd, or group.
- P5_ViewModifyAttributes – place for accessing available parameters for existing components, for example via menu, or commands via terminal.

Required transitions of the defined model are:

- T1_NeededComponents – UNIX operating system start defined items. These items must be running for the operating system to function correctly.
- T2_FormationOfChild – method of creating a init process.
- T3_Fork_CallKernel – kernel call "fork" is used to create the next processes.
- T4_AccesssToInformation – the operating system accesses needed information for the operating system's optimal work.
- T5_AccessProcessFiles – operating system uses own processes to carry out defined activities for users. This information is for example PID, PPID, CMD.
- T6_ReturnUNIX_Login – returns to start point P1.

The validity of the defined model is verified by starting the given simulation. A route cycle is built from place P1 via specified transitions and places. Places P2 – P4 create standard components of the UNIX operating system. Places P3-1 – P3-6 create items of major processes generating a given mode, places P4-1 – P4-7 make important files and scripts. These files use the operating system to obtain necessary information. The next route returns to place P1.

The created model of UNIX architecture considers the multiuser mode. Only this mode is important from the user view because this mode is dedicated to standard operation with computer hardware sources and hosted applications. The next reason is that the aim is to run the operating systems in this mode as long as possible, without any outages or downtime. Despite these efforts, a break in the operating system's function occurs because of the necessity to update system software, install specialized software products, add new hardware, corrupt the file system, or hardware errors.

VI. Methodology of Research and Discussion

In the analysis of information technology products is a necessary quality model, evaluation and measurement support tools. [6] Analysis of selected database and operating systems use Petri Nets. Petri Nets were chosen to analyze Oracle database system and UNIX operating system. Petri Nets are primary used for simulation parallel systems and distributed database to describe state changes in a system with transitions. [15] The created model specifies important features of a given reality and they enable to better know the studied system. This analysis is centered on the description of parallelisms for adopted architecture of given products with possibilities of classic analysis. This is the reason for select Petri Nets to analyze on base simulation. Simulation requires an executable and hence a formal model of the software. A graphical visualization of the model is equally important to enhance the understanding of the software requirements. [12]

The default structure of Petri Nets is defined as <P, T, I, O, H>, where

- P is final set of places,
- T is final set of transitions,
- I are input functions,
- O are output functions,
- and H are inhibitory functions.

Marking the Petri Nets is the multi-set over set of places:

$$\sum_{p \in P} M(p)´p = M(p1)´p1 + M(p2)´p2 + \dots + M(p/P/)´p/P/ \text{[8]} \quad (2)$$

Classic analysis of Petri Nets for models is realized by matrix representation with incidence matrix and set of reachable markings. The definition incidence matrix C is:

$$C = O^T\text{-}I^T \text{ [8]} \quad (3)$$

Benefit is easily demonstrated by offered activities and their confrontation. Realized own models are created in simulating program of Petri Nets HPSim. [11] For correct analysis of real situations, object access is used. A created class of objects is defined by attributes and accessible methods. Net Objects are defined in the HPSim environment program (Workspace section). Net Objects are Transition, Place, and Arc. The model creator must draw needed places for available windows, transitions as ways to change through the menu, commands, or buttons. Places and transitions must be linked via oriented edges. Places are drawn as circles, transitions are black rectangles, and edges are arcs.

An analysis of a selected database system using Petri Nets aims to specify the characteristics of a good user product and a detailed look on the existing similarity between the adopted solutions of other products. General characteristics of a good user of the product [5] can be divided into the following areas:

- efficiency and effectiveness,
- easy handling and user-friendliness,
- user resiliency,
- credibility and support for contextual help,
- adaptability to different users, and
- stability of the main features and requirements.

The characteristics of a user-friendly product are different for each user and firm. The quality of product is influenced by orientation activities, company size, and levels of information technology integration. Important characteristics are time, finance, and human resources. The difficulties are also SMEs. But this situation is not present a reason for resignation from the excellent use of information technology products such as the Oracle database system, UNIX operating system, or Business Intelligence products.

These users are work hard and specialize in very unique challenges with a need to succeed in a competitive market environment. They:

- are used to getting information quickly,
- are familiar with multitasking,
- prefer graphics over text information,
- work better if they have a network.

The above-mentioned characteristics are also relevant for young people – the network-oriented ("net-centered") generation, which is defined by M. Prensky [9]. Linking the ideas of good user products and dynamic characteristics of SMEs, rules can considered for information technology products. One of these rules is a dedicated number of realization steps to any given activity. In our exercise, it is the starting or stopping of selected products like database and operating systems. Of course, it is advantageous for every user to start the application in one step - for example, from a menu or script. All other application dependencies (necessary files and processes) must ensure their own processes.

On the basis of the realized analysis, the Oracle database system has a little problem. The database system must be started via three processes by hand. The useful script to run all these components must edited with text editors. The reason for this solution can be complexity and breadth of activities. The presented idea is not, however, justified by the UNIX operating system, with regard to experiences.

Operating systems are also complicated. They ensure effective management of available hardware resources by multitasking. Managing processes and processors is one of the complicated areas. Multitasking allows you to switch to another program without interrupting work in progress. Multitasking facilitates the implementation of activities that, for reasons of principle, must be conducted in parallel with other activities of the computer. Operating systems do not have a problem with starting in one step with a special command. The specialized processes “swapper” and “init” start all the needed processes for all users in the multiuser mode. The required information is read from defined files and scripts.

Conclusion

Modern information technology products must reflect the needs of all users. Regardless of preferences, activity orientation, company size, time, and finance, public rules must be defined for information technology products. These rules respect the requirements of good user products and the young generation. A natural requirement is a start or stop application in one step (transition) by specialized processes with competences and rights to do this activity. With regard to adopted solutions from available applications, we can consider this requirement as viable.

A good example is the UNIX operating system with defined processes “swapper” and “init.” The swapper process is created automatically when booting the system. The init process is created by the swapper process with the kernel call - fork. This process starts all the needed processes of the operating system and hosted applications. On the other hand, sophisticated product like the Oracle database system have two possible ways – starting the database system via three processes by hand, or creating a script to run all the needed components. This solution is quite impractical in our dynamic and global information society.

References

[1] A. Abraham, F. Xhafa, “Computational models and heuristic methods for Grid scheduling problems”, Future Generation Computer Systems 26 (2010) pp. 608-621, Elsevier 2009, [on-line], [cit. January 12, 2011]. From: http://www.softcomputing.net/fgcs2010_f.pdf.

[2] A. Silberschatz, P. B. Galvin, G. Gagne, „Operating System Concepts Essentials“, John Wiley & Sons: 2010, pp. 99-114, ISBN 978-0-470-88920-6.

[3] “Attributes area information”, [on-line], [cit. February 8, 2011]. From: http://informatika.topsid.com/index.php?war=informace.

[4] “Difference Between Data and Information”, [on-line], [cit. February 10, 2011]. From: http://www.differencebetween.net/language/difference-between-data-and-information/.

[5] “General characteristics of a good user of the product”, Programming Technology, [on-line], [cit. December 12, 2010]. From: http://odkazy.internetshopping.cz/internet/technologie_programovani/otazky.htm.

[6] “International standard ISO/IEC 25045”, Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Evaluation module for recoverability, [on-line], [cit. December 10, 2010]. From: http://webstore.iec.ch/preview/info_isoiec25045%7Bed1.0%7Den.pdf.

[7] J. Černý, „Is world information overload?”, [on-line], [cit. February 15, 2011]. From: http://pctuning.tyden.cz/index.php?option=com_content&view=article&id=20154&catid=1&Itemid=57.

[8] M. Kochaníčková, “Petri Nets”, Palacky University, Faculty of Science, Department of Computer Science, Olomouc 2008, pp. 15, 35-41, without ISBN.

[9] M. Prensky, “The Reformers Are Leaving Our Schools in the 20th Century”, [on-line], [cit. January 9, 2011]. From: http://www.marcprensky.com/blog/.

[10] “Oracle Database Concepts”, [on-line], [cit. May 9, 2010]. From: http://www.oracle.com/technology/documentation.

[11] “Petri Nets Tools Database Quick Overview”, [on-line], [cit. May 9, 2010]. From http://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/quick.html

[12] R. Gold, “Petri Nets in Software Engineering”, [on-line], [cit. August 10, 2010]. From: http://www.haw-ingolstadt.de/fileadmin/daten/allgemein/dokumente/Working_Paper/ABWP_05.pdf.

[13] “Theory information”, [on-line], [cit. February 2, 2011]. From: http://wp.soulwasted.net/msz/pds/teorie-informace-mnozstvi-informace-ve-zprave-entropie-zdroj-diskretnich-zprav-kodovani-pro-potlaceni-redundance.

[14] “What bring WebExpo 2010 conference”, Good site - User experience, [on-line], [cit. January 12, 2011]. From: http://blog.dobryweb.cz/konference-webexpo/.

[15] B. B. Sarkar, N. Chaki, „Transaction Management for Distributed Database using Petri Nets“, *International Journal of Computer Information Systems and Industrial Management Applications* (IJCISIM), ISSN: 2150-7988 Vol. 2 (2010), pp. 069-076. [on-line], [cit. April 5, 2011]. From: http://www.mirlabs.org/ijcisim.

[16] M. Fišer, “Firms and households are at data chaos, if they do not change the approach to information management”, *EMC magaz in*. June 2010.

[17] S. Supavetch, S. Chunithipaisan, „The SQL-Based Geospatial Web Processing Service“, *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*, ISSN: 2150-7988 Vol. 3 (2011), pp. 119-126. [on-line], [cit. April 8, 2011]. From: http://www.mirlabs.org/ijcisim.

[18] Y. Dai, T. Kakkonen, E. Sutinen, „MinEDec: a Decision-Support Model That Combines Text-Mining Technologies with Two Competitive Intelligence Analysis Methods“, *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*, ISSN: 2150-7988 Vol. 3 (2011), pp. 165-173. [on-line], [cit. April 5, 2011]. From: http://www.mirlabs.org/ijcisim.

Author Biography

Milena Janakova is a assistant professor, researcher, and project manager at Silesian University in Opava (School of Business Administration in Karvina), Czech Republic. Janakova received her Ph.D. in Managerial science from the Technical University of Ostrava, Czech Republic in 2001. Research interests are related to database and operating systems, Business Intelligence, Customer Relationship Management, and their applications in education and SMEs.