



Conference on ENTERprise Information Systems / International Conference on Project
MANAGEMENT / Conference on Health and Social Care Information Systems and Technologies,
CENTERIS / ProjMAN / HCist 2015 October 7-9, 2015

Organizational Operating Systems, an approach

Carlos Páscoa^{a,c*}, José Tribolet^{b,c}

^aDepartment of Education, Portuguese Air Force Academy, Sintra, Portugal

^bDepartment of Information Systems and Computer Science, Technical University of Lisbon, Portugal

^cCODE – Center for Organizational Design & Engineering, INOV, Rua Alves Redol 9, Lisbon, Portugal

Abstract

Like working with the computer, managing organizational resources and using them optimally is also an exceedingly challenging job. Therefore organizations are also equipped with an Organizational Operating System whose purpose is to provide resources with a better, simpler, cleaner model of the organization and providing the means to successful management. Starting from the computer example and its operating system description and by comparison with the organizational configuration concepts, as an initial approach to this important organizational subject, we characterize the organizational operating system, using DEMO, by defining its basic operations, devices, basic utilities (resources), structure and basic concepts.

© 2015 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of SciKA - Association for Promotion and Dissemination of Scientific Knowledge

Keywords: Organizational Configuration, Flying the Organization, Computer, Operating System, Organizational Operating System, DEMO.

1. Introduction

Given the increasing organizational complexity, several perspectives have been defined by social scientists, management scientists and engineers that have come together in a shared effort to capture, analyze and understand the multitude of factors that affect the organizational world. Organizations are dynamic systems that run in complex environments and need to react to changes, by increasing its self-awareness and its ability to transform and adapt. Failure to adapt can lead to disruption. Adaptation mechanisms need to consider the wholeness of the organization to maintain its viability and performance.

* Corresponding author. Tel: +351-214-726-129.
E-mail address: cjpascoa@gmail.com

To manage this complexity, organizations need “something” that can tie all different aspects of the organization into a seamlessly running machine guaranteeing that all these parts come together in a viable and performative organization.

However, that “something” that glues the organization parts, being only an idea that needs clarification and research in order to be materialized into a tangible concept, can benefit from previous approaches and known concepts, theories and models.

Páscoa and Tribolet^{1,2,3} proposed the use of the flying aircraft organization concepts, based on the metaphor “Flying the Organization”, implementing, within the business organization, the same concepts in order to attain the aircraft agility in near-real time. An aircraft, like an organization, possesses the glue that ties its components together in a seamlessly viable organization. However, when we think of “something” that glues different parts together, that already belongs to the world culture, we think on the computer and its operating system. In fact, the computer has components and the operating system is the “something” that glues these components making the computer a viable and performative organization.

Tanenbaum⁴ states that “*A modern computer consists of one or more processors, some main memory, disks, printers, a keyboard, a mouse, a display, network interfaces, and various other input/output devices. All in all, a complex system*”. The author further states that “*managing all these components and using them optimally is an exceedingly challenging job. For this reason, computers are equipped with a layer of software called the operating system, whose job is to provide user programs with a better, simpler, cleaner, model of the computer and to handle managing all the resources*”.

The document is structured as follows: section 2 introduces the Computer and Operating Systems; section 3 presents the foundations for the Flying the Organization concept, its components and the need for near real time steering; section 4 outlines DEMO models and presents its theoretical foundations; section 5 proposes an approach to devising the organizational operating system; section 6 concludes and section 7 presents future research recommendations.

2. Computer and Operating Systems

According to Shelly & Vermaat⁵ a computer can be defined as an “electronic device, operating under the control of instructions stored in its own memory, that can accept data (input), process the data according to specified rules (process), produce results (output), and store the results (storage) for future use”. Input, process, output and storage are the four basic computer operations.

To perform operations, a computer needs two main utilities: i) hardware, a collection of electric, electronic and mechanical components; ii) software, also called a program, consists of a series of related instructions, organized for a common purpose, that tells the computer what tasks to perform and how to perform them. Furthermore, the computer is divided into four main components: input devices, processor, memory and peripherals (output, storage and communication devices).

As computers are used by humans, acting as users, an operating system (OS) is a software program that “acts as an intermediary between a user of a computer and the computer hardware”⁵.

Tanenbaum⁴ states that OS provide a clean abstract set of resources that helps humans to coop with complexity. Silberschatz & Galvin⁶ propose three main goals for the OS: i) execute user programs and make solving user problems easier; ii) make the computer system convenient to use; iii) use the computer hardware in an efficient manner. The authors consider users (people, machines, other computers) to be computer system components.

To provide comprehensive understanding about OS, basic concepts are provided such as: processes, address spaces, files, protection, shell (command interpreters) and system calls.

The monolithic basic OS has the following structure: i) a main program that invokes the requested service; ii) a set of service procedures that carry out the system calls; iii) a set of utility procedures that help the service procedures.

Silberschatz & Galvin⁶ divide the OS into three components: i) resource allocator that manages and allocates resources; ii) control program that controls the execution of user programs and operation of I/O devices; iii) kernel, the one program running at all times (all else being application programs).

3. Flying the Organization and the Organizational Configuration

Páscoa¹ and Páscoa & Tribolet^{2,3} proposed the metaphor^{7,8} “Flying the Organization” arguing that, like in an aircraft, organization’s failure to adapt can lead to organizational disaster. Today’s need to near real time reaction to constant changes in organizations, demands proactive thinking and agility.

Páscoa & Tribolet^{2,3} compare organizational concepts with aircraft concepts and identify some key factors like culture, mission planning, configuration selection, real time feedback and control, adjustment mechanisms, and debriefing (feedback) to successfully steer the organization. As aircraft are flown in determined configurations, the authors state that the near real time running organization needs to define several organizational configurations, in order to be able to react to adverse or favorable conditions, which can affect its flight plan in a negative or positive way.

Like an aircraft, an organization also has a set of components that have to be considered, in possible combinations to perform its mission and achieve its end state, which the authors claim to be an Organizational Configuration (OC) that can be defined as “a set of components that are combined in an optimal manner to perform the mission and achieve its end state” to clearly identify: i) WHAT is on the configuration; ii) WHO manages the configuration⁹; iii) WHEN is the configuration managed.

To relate components, the authors make the following statements: i) the aircraft configuration (the WHAT) serves as a basis for conducting the flight and evaluate anything that happens in terms of what is affected in the wholeness of the configuration; ii) the aircraft present and future flying conditions are constantly monitored and evaluated by the crew (the WHO) that takes the convenient actions to alter the configuration when something is affecting or is thought to affect it (the WHEN). For the objective of this paper it only matters to delve a little on the OC’s WHAT component.

Páscoa and Tribolet^{2,3}, using General Systems Theory (GST)^{10,11,12,13,14,15} and other models like those proposed by the Business Rules Group (BRG)¹⁶ and Lankhorst et al¹⁷, argue that an organization is a system being (the set of system elements that characterize its uniqueness (DNA, structure, organization and resources) in the universe of systems), becoming (the process of identifying the system’s future state, considering its ability to import energy, its ability to produce output, the cycles of events needed and its ability to maintaining dynamic homeostasis while minimizing entropy and therefore, assuring its existence within other systems) and behaving (the system’s capacity to monitor and stay on its path towards its future state in spite of changing conditions), that influences and is influenced by other systems.

The authors defend that resources can take many forms like human, material, infrastructure, finance, information, knowledge and wisdom. Like in an aircraft, each combination of resources is a configuration designed to attain an objective or future state. In this context, every resource is essential to obtaining the desired end-state.

4. Enterprise Ontology and DEMO

Generally speaking, an ontology is the metaphysical study of the nature of being and existence¹⁸. Dietz claims that the ontology of something is a conceptual model that satisfies the next requirements: coherent, comprehensive, consistent, concise, and essential¹⁹. The author distinguishes between world ontology and system ontology defining enterprise ontology “as the realization and implementation independent essence of an enterprise, in short, as the deep structure behind its observable surface structure”^{18,20}.

The Design and Engineering Methodology for Organizations (DEMO)^{21,22} has shown to be useful in a number of applications due to:

- None of the enterprise modeling techniques purpose a reduction of complexity as high as the one purposed by DEMO (over 90%). This topic is directly related with the concise and essence focus qualities of DEMO;
- DEMO methodology is based on a rigorous theory: the Psy-theory which combines the knowledge from ontological works, language/action perspective, logic and systems theories. This stands for coherence and consistence of the models;

- DEMO defines three notions that are considered relevant in governing the enterprise dynamics (competence, authority and responsibility). Most of these notions are absent or not clear defined in others enterprise modeling techniques;
- DEMO has been widely accepted in both scientific research and practical appliance. In fact DEMO practical application has been successful validated in several enterprises.

Focusing on the ontological production DEMO defines four models^{23,24,25}: Process, Action, State and Construction. The Construction Model (CM) specifies the construction of the organization system by the identified transaction kinds and the associated actor roles, as well as the information links between the actor roles and the information banks. The Process Model contains, for every type in the CM, the specific transaction pattern (basic pattern, standard pattern, cancellation pattern) of the transaction kind. The Action Model specifies the imperatively formulated business rules that serve as guidelines for the actors in dealing with their agenda and the State Model specifies the state space and the transition space of the production world by the object classes and fact types, the results types and the ontological coexistence rules that can be considered as declarative formulations of business rules.

The Actor Transaction Diagram (ATD) expresses the interaction structure of an organization consisting of the transaction kinds in which the identified actor roles participate as initiator or executor. The Transaction Result Table (TRT) is the result of the step of Transaction Pattern Synthesis where from all the production facts obtained in the step of Coordination-Actors-Production Analysis, the corresponding result types are devised, as well as transactions responsible for their production.

5. The Organizational Operating System

Like working with the computer, managing organizational resources and using them optimally is also an exceedingly challenging job. Therefore organizations are also equipped with an Organizational Operating System (OOS) whose purpose is to provide resources with a better, simpler, cleaner model of the organization and providing the means to successful management.

Starting from the computer and its OS description (in section 2) and by comparison with the OC concepts (in section 3), as an initial approach to this important organizational subject, we characterize the OOS by defining its basic operations, devices, basic utilities (resources), structure and basic concepts.

Some principles must be defined to allow the transition between a computer OS and the organizational OS:

- *1st principle.* The organization is a system within systems. As a system it receives input from other systems, processes that input and turns it into output for exchanging with external systems. The ability to understand and foresee these dynamic changes causes the need to store them in the system's memory. These are organizational basic operations.
- *2nd principle.* The organization, like the computer, has devices that are used to perform the basic operations. These are input devices that allow the system to get information about other systems, processors that allow computing, memory that allows computing and organizational knowledge and output devices to transmit the added value generated.
- *3rd principle.* Hardware and software exists in the OOS in the form of resources.
- *4th principle.* The organization is structured in kernel programs that run continuously, process programs that allocate resources and control programs that monitor other programs execution.
- *5th principle.* The OOS basic concepts include processes, address spaces, files, protection and system calls.

Figure 1 shows the Organizational OOS with principles and its characterization.

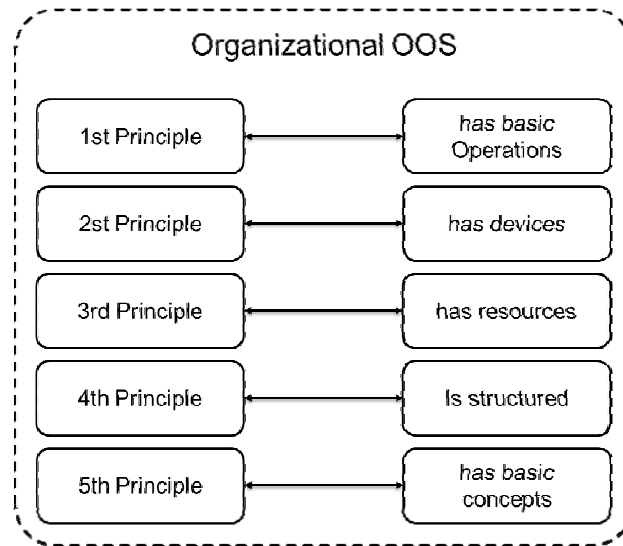


Fig. 1. Organizational OOS – principles and characterization.

The principles are further explained in the next sections.

5.1. OOS Basic Operations

We argue that, like the computer organizations, as complex entities, need abstraction to coop with its processes and, also like the computer, organizations perform four basic operations:

- Input, to scan or to accept the actions that come from the external environment;
- Process, to actually digest and process these actions, for instance in the accomplishment of the business processes;
- Output, that signifies, in the case of organizations, value creation that is normally handed to an external value taker;
- Storage, as the recording of several artifacts that, for example, drive organizational memory.

To model the OOS basic operations we use DEMO models explained in section 4. Figure 2 shows the Actor Transaction Diagram (ATD).

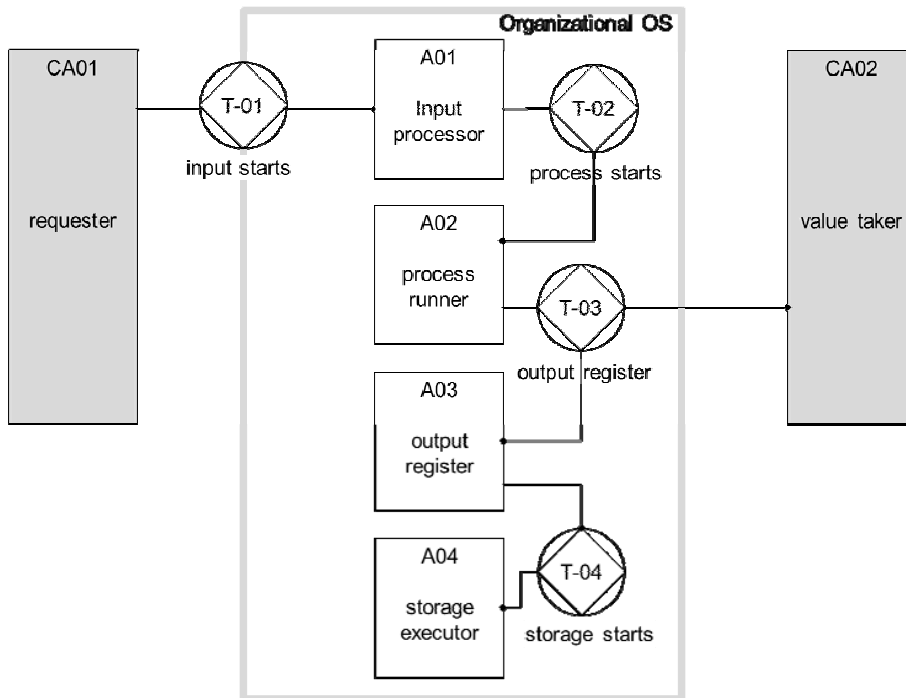


Fig. 2. ATD of the Organizational Operating System basic operations.

The requester initiates an action that serves as input to the organization. The input is then processed and turned into an output that produces value and is delivered to the value taker. To maintain organizational knowledge (memory), there is a storage process.

Table 1 shows the TRT.

Table 1. TRT of the Organizational Operating System basic operations.

Transaction kind	Result Type
T-01 input start	R01 input collection has been started
T-02 process start	R02 process running has been started
T-03 output register	R03 output registering has been started
T-04 storage start	R04 storage has been started

5.2. OOS Devices

According to the 2nd principle the organization has devices that perform the basic operations. The devices considered are:

- Input, that allows to get information about other systems. One example is communication and its channels.
- Processors, which allow computing and running and completing basic operations on the basic utilities using structure and basic concepts.
- Memory, that allows basic computing and long term organizational knowledge.
- Output devices, to transmit the added value generated and communicate to the exterior.

5.3. *OOS Basic Utilities*

According to the 3rd principle, when compared to the OC we propose to set the OOS basic utilities, composed by hardware and software, with the following match:

- Hardware: material and infrastructure since those are tangible resources.
- Software: human, finance, information, knowledge and wisdom, since those are tangible and intangible resources.

We propose that humans are a kind of application that runs in the OOS. When a certain application runs in a computer operating system, it suffers before an adaptation process to guarantee its compatibility with the operating system and other applications. A human application that is set to be running on a determined OOS suffers also, through the process of acculturation, an adaptation process to guarantee compatibility with the system and with other applications.

5.4. *OOS Structure*

According to the 4th principle, we claim that the OOS structure includes several programs:

- Kernel programs that run continuously, providing organizational boot strap and basic register settings. Examples are the nuclear definitions for words, nouns and basic forms of joining those together to form phrases.
- Process programs that allocate resources and produce the output.
- Control programs that monitor other programs execution and provide overall feedback to the kernel program.

5.5. *OOS Basic Concepts*

According to the 5th principle, OOS possesses the following basic concepts:

- Processes, used to run the business processes, for example.
- Address spaces, that change from program to program that, in the organization corresponds to different entities within the same organization.
- Files, used to store organizational OOS memory.
- Protection as a way to safeguard different processes and files.
- System calls, used by processes to check, for example, basic utilities availability.

6. **Conclusion**

As a first approach to understand the organizational operating system, we have done research about computer operating systems. Within the organizational configuration concepts, we identified the computer operating system characteristics and made the bridge to the organizational operating system identifying: i) five principles to facilitate its understanding; ii) its basic operations, devices, basic utilities (resources), structure and basic concepts.

We further characterized the organizational operating system basic operations using DEMO's actor transaction diagram and the transaction result table.

Naturally, we regard the paper as an initial approach to the subject of operating systems in organizations. Further research needs to be conducted to better understand the concept and it works within the organizational environment.

7. **Future Work**

Future research includes:

- - Use of DEMO should be extended to represent each OOS concept and extend its comprehension and relation.

- Another challenge in the scientific domain is to dive into the relations between concepts not only to formally characterize them but to also explore the unveiled potential.
- Concepts like basic concepts include processes, address spaces, files, protection and system calls should be further delve into using DEMO to understand the small characteristics that can actually make OOS complexity a little bit more understood.

References

1. Páscoa, C., Organizational and Design Engineering of the Operational and Support Dimensions of an Organization: The Portuguese Air Force Case Study. Unpublished Doctoral thesis. Technical University of Lisboa, Instituto Superior Técnico, Lisboa, 2012.
2. Páscoa, C.; Tribolet, J., Organizational and Design Engineering of the Operational and Support Components of an Organization: the Portuguese Air Force Case Study, Springer Lecture Notes in Business Information Processing (LNPIB) Series on Minutes of the Practice-driven Research on Enterprise Transformation (PRET) Conference on Enterprise Engineering held in the University of Delft, The Netherlands, November 2010.
3. Páscoa, C., & Tribolet, J., Maintaining Organizational Viability and Performance: The Organizational Configuration Map. In G. Jamil, A. Malheiro, & F. Ribeiro (Eds.) Rethinking the Conceptual Base for New Practical Applications in Information Value and Quality (pp. 266-283). Hershey, PA: Information Science Reference. doi:10.4018/978-1-4666-4562-2.ch012, 2014.
4. Tanenbaum, A., Modern Operating Systems, 3rd Edition, Prentice Hall, 2007.
5. Shelly G.; Vermaat, M., Microsoft Office Introductory, Course Technology, Cengage Learning, Boston, USA, 2011.
6. Silberschatz, A; Galvin, P., Operating System Concepts, Addison-Wesley, 1994.
7. Knowles, M.; Moon, R., Introducing Metaphor, Routledge, London, UK, 2006.
8. Lakoff, G., & Johnson, M., Metaphors we live by. Chicago: The University of Chicago Press, 1980.
9. Páscoa, C.; Aveiro, D.; Tribolet, J., Organizational Configuration Actor Role Modeling Using DEMO: Springer Lecture Notes in Business Information Processing (LNPIB) Series on Minutes of the Practice-driven Research on Enterprise Transformation (PRET-4) Conference on Enterprise Engineering held in Poland, June 2012.
10. Skyttner, Lars, General Systems Theory: Problems, Perspectives, Practice, World Scientific Publishing Company; 2 edition, London, UK, 2006.
11. Boulding, K., General Systems Theory — The Skeleton of Science, Management Science, No. 2, 1956.
12. Boulding, K., General Systems as a Point of View in A. Mesarovic's, Views on General Systems Theory, John Wiley, New York, USA, 1964.
13. Bowler, D., General Systems Thinking, North Holland, New York, 1981.
14. Churchman, W., The Design of Inquiring Systems: Basic Concepts of Systems and Organizations, Basic Books, New York, 1979.
15. von Bertalanffy, L., General Systems Theory, Main Currents in Modern Thought, Vol. 71, No. 75, 1955.
16. Business Rules Group (BRG), The Business Motivation Model, available at: <http://www.businessrulesgroup.org/bmm.shtml>, 2007.
17. Lankhorst, M. (2005), Enterprise Architecture at Work - Modelling, Communication and Analysis, Springer-Verlag Berlin Heidelberg, Germany.
18. Dietz, J. L. G., Enterprise Ontology: Theory and Methodology. Delft: Springer, 2006.
19. Aveiro, D.; Rito, A.; Tribolet, J., Towards a G.O.D. Organization for Organizational Self-Awareness, Advances in Enterprise Engineering IV, Lecture Notes in Business Information Processing Volume 49, Part 1, 16-30, Springer Berlin Heidelberg, Germany, 2010.
20. Dietz, J. L. G., The Atoms, Molecules and Fibers of Organizations, Data & Knowledge Engineering Journal - Special issue, Volume 47 Issue 3, December 2003, Elsevier Science Publishers B. V. Amsterdam, The Netherlands, 2003.
21. Dietz, J. L. G., Architecture - Building strategy into design, Academic Service – Sdu Uitgevers bv, 2008.
22. Dietz, J. L. G., Is it $\phi\tau\psi$ or bullshit, symposium on Methodologies for Enterprise Engineering, farewell lecture, Delft University of Technology, Delft, The Netherlands, 2009.
23. Dietz, J. L. G., DEMO-3 Way of Working, in www.demo.nl, 2009a.
24. Dietz, J. L. G., DEMO-3 Models and Representations, in www.demo.nl, 2009b.
25. Dietz, J. L. G.; Habing, N., A Meta Ontology for Organizations, On the Move to Meaningful Internet Systems: OTM 2004 Workshops, Lecture Notes in Computer Science, 2004, Volume 3292:533-543, Springer Berlin / Heidelberg, Germany, 2004