

Clopenness of Systems: The Interwoven Nature of Ecosystems

Joost te Molder, Ben van Lier, Slinger Jansen

Utrecht University, The Netherlands {j.temolder, s.jansen}@cs.uu.nl
Centric, Gouda, The Netherlands, ben.van.lier@centric.eu

Abstract. Openness of a software product or software producing and end-user organization is perceived as a binary concept: organizations are perceived to be closed, with its negative connotations, such as being dictatorial, undemocratic, and opaque, whereas an open organization is positively considered to be transparent and favorable to deal with. The binary view of openness, however, is harmful for the software industry, since wrong decisions are made on these qualifications, calling for a better definition of the concepts of openness and closeness. In this paper a definition is given of the concepts, which led to the concept clopenness; a model is provided that assesses the clopenness of a software producing or end-user organization and its software products, and a case study is performed to show the use of the model and concepts. The results of the paper are that the use of the clopenness model provides insight into the openness of an organization, in turn enabling better founded business decisions.

Keywords: openness, closeness, organizational clopenness, software clopenness, clopenness assessment

1. Introduction

Openness is perceived as a binary concept; ‘open’ or ‘closed’. Open and closed are two states that are used by software vendors to label their software and corresponding organization. In addition, organizational and software openness and closeness have become popular buzzwords that attract the interest of various organizations and their customers. Typically, organizations make rhetoric use of the adjective open, use it as a marketing pitch to describe their organization, and attribute positive effects to this openness. One question that needs to be asked, however, is whether they are ‘truly’ open or closed? Open and closed are opposites, but we argue that these are more than simple opposites. To date there has been little agreement on what open exactly is, and what is meant by closed [12]. Consequently, how is openness or closeness perceived [11]? The debate between open and closed is challenging, vague and broad, covering different perspectives and viewpoints [4], [21], [23], [29], [8]. All actors, e.g. organizations, developers, end-users, competitors and other third parties, have their own opinions, beliefs and intentions in their particular realities, that tend to have philosophical intentioned descriptions and

meanings. These interpretations have shifted over time, hence we recognize a new perspective on the open vs. closed discussion. The paper is divided in five parts. The first section deals with the research approach, design, and related methods. The second section gives a brief overview of the origins and status quo of the concepts open and closed, attempts to show shortcomings and ideas of these concepts and outlines the theoretical concepts of the research. The third section describes how the concepts are combined and structured into a model, the subsection elaborates on evaluation of the model. The next section shows the use of the model and concepts in case studies. The paper ends with the conclusion, discussion and directives for further research.

2. Research Approach

Different methods were used in this study; the general method structuring this research can be characterized as design science research [24]. With a literature study the status quo of the phenomena open and closed (systems) in different scientific fields were studied. Based on the literature study we discuss the status quo and aim at revealing shortcomings and ideas in the field to construct a new perspective or artifact. Complementary, viewpoints are created on the found shortcomings and ideas, resulting in the development of a new perspective on the subject matter. A multiple case study design, as described by Yin [33], was chosen to allow more analytic benefits for this research than a single-case design, because of the complexity of the phenomena. A preliminary pilot case study and a few meetings preceded the agreement over the case study protocol. By means of four case studies existing of semi-structured interviews with experienced information technology managers, we evaluated the phenomena in a real-life context, i.e. in systems in the transport and logistics ecosystem. The results were drawn by using an interpretive study approach [20] on the collected data, and the use of various analytic techniques such as replication logic, explanation building and cross-case synthesis [33].

3. Open and Closed Systems

Open and closed systems originate from the 1950s in physical sciences, or to be more specific; from thermodynamics. In thermodynamics, an open system is a system that exchanges heat and work with their environment [27], this is in contrast with a closed or isolated system, which exchanges neither heat nor work with their environment. In life sciences and biology, living systems are defined as open systems, “maintaining themselves in exchange of materials with environment, and in continuous building up and breaking down of their components” [27]. The term open system was further developed in systems theory. Bertalanffy observes that “we call a system closed if no materials enter or leave it. It is open if there is inflow and outflow, and therefore change of the component materials” [26]. In general systems theory, Boulding [5] emphasized that organizations are open systems, or self-maintaining structures. We note that technology played no role yet in Boulding’s general systems

theory. An open system is an organization that is capable of self-maintenance by throughput of environmental influences [5]. On the other hand, a closed system is one that assumes that the organization is only affected by internal influences. Environmental influences can be participants or individual firms and/or products that mutually depend on each other. An open system as opposed by Boulding is essentially closed in itself; this is expressed by the theorem ‘openness based on closeness’ [2]. For example, a human can be closed in the sense that thoughts are protected by a skull, but we are still open in the sense that we can communicate these thoughts with each other. Derivatives of open and closed systems are also found in economics [32], politics [17] and management science [7]. For example, the regime of Obama is characterized as open, while we recognize the contrary in the regime of Mubarak or Kim Jong-Il. These systems are based on the fundamental principles of Boulding or von Bertalanffy, which are previously described.

3.1 Theoretical Concepts

So far, open and closed systems have been viewed as opposites. We argue that they are more than simple opposites. Maxwell [16] suggests that something is not simply open or closed; it ranges from open to closed and encompasses varying degrees of openness. There is a continuum in between open and closed. A **continuum** has two endpoints or extremes, open and closed. Something can be placed anywhere in between those two points, but can never be the endpoint itself. Completely open or closed systems do not exist. Therefore, we consider the open versus closed discussion as a false dichotomy. A false dichotomy appears in a situation where only two options are discussed, in this case open and closed, when in fact there are more options. Crucial decisions with great impact are taken, with a naïve and myopic view of the concepts. The naïve and myopic view is also emphasized by West [28], when he tries to make sense of open standards. Standards are often characterized as either open or proprietary, two options, while there are black, white, and many shades of gray. To determine how open or closed something is, we suggest attaching a degree of openness or closeness to organizations or software. We do not want the impression that we lean toward open by using openness, or tend toward closed by using closeness. Therefore, we will use the term ‘**clopenness**’, that refers to both openness and closeness, which respectively are build upon open and closed. It also addresses the theorem of openness based on closeness and thereby ‘solves’ the false dichotomy. Complementary, we can say that the discussion should never be about open versus closed, but about the **degree of clopenness**. In mathematics, a clopen set is a subset of a topological space; both open and closed, the topological space is finite dimensional, a point on this space is determined by postulates of some kind [10]. Although the function in mathematics is different than the context of this article, we can semantically deduce that the topological space refers to a **continuum**. The degree on the continuum can be placed anywhere in between open and closed.

Cloopenness suggests that systems are always both open and closed, in which the system as a whole determines how the parts behave; the connections between the parts enable systems to ‘communicate’ with their environment [25]. The whole exists of parts such as software, a device, network and the environment. Parts can be

systems themselves but are also composed of other parts. Clopenness can refer to ‘clopenness in parts’ i.e. how the parts behave, and ‘clopenness of parts’ i.e. the connections between parts that compose the whole.

Open systems are popularized by the open-source community, but this can be misleading. The characterization source in ‘open-source’ says it all; it is only about the source code in this case, not about other organizational processes or software parts. Open standards, open formats and open source are often considered equivalent or mutually implicated [6]. In addition, software products characterized as closed source can have open components and vice versa. We can ask ourselves the following: is the Android OS open because the source code is available and allows modification? Is Microsoft Windows closed because the source code is not available? Source code is just a part of the whole, all that can be said about this is that the degree of source code clopenness differs between these two operating systems. We describe opening up of a specific part or resource of the organization, e.g. a certain process or intellectual property, or a part of the software, e.g. APIs or source code. This results in clopenness of a resource itself within software clopenness or organizational clopenness. We refer to these resources or parts as **organizational and software clopenness options**. A certain distinction is recognized, there are two types of clopenness: of/in an organization and of/in software. The **organization** is the whole that developed or uses (end-users) the software product. This can be a business unit, a group of people who work together, groups who work together, the whole organization, a consortium, or any other form of organizations, as long as the whole is focused on the development or the use of a specific software product. The organization also includes the organizational core components of governance, research and development, software product management, marketing and sales, and consulting and support services [22]. The **software product** is defined as “a packaged configuration of software components or a software-based service, with auxiliary materials, which is released for and traded in a specific market” [31].

The degree of clopenness, and so a point on the continuum, is determined by how an option approaches the clopenness key attributes. These can be found in Table 1.

Table 1. Descriptions of clopenness key attributes

Key attribute	Description
Availability	<i>The degree to which an option</i> is available internal as well as external when it is required for use or reference [16][11][7].
Accessibility	Is accessible, can be consulted by as many interested people as possible, regardless their status towards the company or the field of endeavor they are in. Can be limited based on registration, subscription, religion, sexes, political persuasion or intellectual goal [16][30][11][7].
Transparency	Is transparent, in terms of no filtered or prohibited content regarding the accessibility mode. Clearly defined, understandable, comprehensible and unambiguous documentation guided by established well known common rules, policies and processes. If applicable, according to standards approved by recognized standard-setting organizations [30][7][15]. Including insights in stakeholders’ experiences and decision making processes based on a plausible frequency of information disclosure. May be limited based on encryption or other protection.

Reciprocity	Is able to be modified or updated in a reciprocal manner, i.e. enabled for interaction with the environment [16][7] based on policies about how to handle with the different stakeholders or other third parties. Not limited by technical or functional shortcomings, such as the lack of tools and features to communicate and share experiences.
Licensing	Is being restricted in the sense of requirements and restrictions [11][21][7] on the modification, extension, and integration of it [18]. Only applicable to software clopenness.

Based on the previous paragraphs the following definition of clopenness is created: *‘Clopenness can be divided in **organizational** and **software** clopenness, consisting of **options** characterized by a certain **degree of clopenness**, which is determined by the degree to which an option approaches the **clopenness key attributes**.’* In which it should always be attached to a context and situation, such as clopenness of an independent software vendor in the software development ecosystem or the clopenness of a transport organization in the transport and logistics ecosystem. In the next paragraph we will elaborate on how to apply this definition in order to assess clopenness.

4. Clopenness Assessment

Based on the previous paragraphs, the Clopenness Assessment Model (CAM) is created. It includes the clopenness types, options and key attributes to determine a degree of clopenness; these concepts together form the base of the CAM. The model is depicted in Fig. 1. In principle, clopenness of any context or situation in any kind of ecosystem with respect to software and organizations can be argued and described using this model and its techniques.

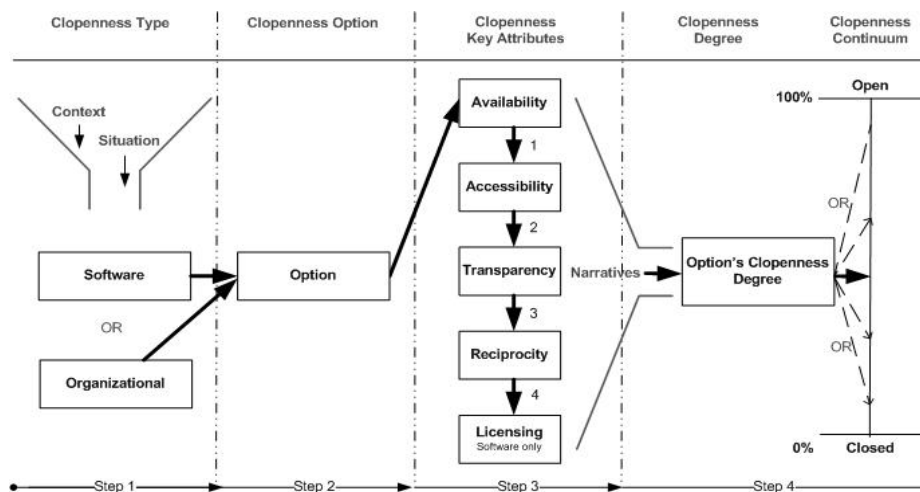


Fig. 1. Clopenness Assessment Model

In this section, we will describe how to use the model properly in order to determine the degree of clopenness. (*step 1*) First of all, any clopenness degree is context-dependent and situational and belongs to a software- or organizational-option. For example, the **context** can be a ‘software developer ecosystem’ in which the **situation** is the clopenness of the ISV’s software development process to the customer. (*step 2*) The software development process is a software **option** and is seen as a part of the ISV, which in turn is a system in the software developer ecosystem. The customer or end-user is part of this ecosystem. The most crucial in this system of systems are the connections between the parts that enable systems to ‘communicate’ with their environment i.e. customers. Another context can be a ‘transport and logistics ecosystem’ in which the situation can be the clopenness of a transportation organization’s monitoring process to the customer. We will elaborate on this context in the next section. (*step 3*) The most essential parts of the model are the **clopenness key attributes**, the descriptions of which are found in the last section. The descriptions should not be used mechanically, they can be openly used and interpreted in narratives. A narrative is a story in a constructive format, which describes a series of events. In this case, we create a ‘story’ about an option to be assessed referring to the clopenness key attributes and using the descriptions as guidelines or criteria for this story. (*step 4*) Finally, a **degree of clopenness** can be determined. However, the focus should not be on the exact number; the narratives are most essential, these give a rich background and context of the case. To give an exact degree of clopenness, one must value the derived criteria from the clopenness key attributes. From the narratives, we can interpret and assign a value to these criteria. Based on how these criteria are described in the narratives and thus how they are assessed these can be assigned a value of 0 or 1, where the 1 states a high degree of clopenness and 0 a low degree of clopenness. All values can be counted and divided by the total of the derived applicable criteria, resulting in a degree, i.e. percentage, of clopenness which can be placed alongside the continuum. For example, a clopenness score of 60% would mean it tends to be more closed than open. As noted before, it is about the rich background and context in the narratives which tells about the connections between the systems or parts, not about an exact score.

In the next section the CAM is used to shape and formulate clopenness in the transport and logistics ecosystem. We will elaborate on how clopenness influences the relations between systems in the transport and logistics ecosystem.

4.1 CAM Evaluation

The CAM is evaluated through four case studies, managed using the case study protocol and a set of criteria. The criteria consisted of issues such as the comprehensiveness and understandability of the model, the message it conveys, the descriptions and interpretations of the key attributes, abstraction level and the representation of the model. The Clopenness Assessment Model is overall positively evaluated and well received, therefore evaluated successfully. The OAM conveys the message of clopenness with respect to technology and organizations.

5. The Interwoven Nature of Clopenness in Ecosystems

The term business ecosystem was first coined by Moore [19] as an analogy on ecological ecosystems. The term business or commercial ecosystems was further developed by Iansiti and Levien [13], and it is seen as a network of organizations, i.e. businesses, suppliers and customers, which are most critical or closely intertwined with your organization. Jansen et al. [14] shifted this definition towards software ecosystems, which is defined as “a set of businesses functioning as a unit and interacting with a shared market for software and services, together with the relationships among them”. Hence, that a software ecosystem is part of a business ecosystem, these blend together, and what they have in common is the term ecosystem.

5.1 Ecosystems Case Study Context

The study is mainly focused on the business ecosystem in the transport and logistics sector and the actors involved in it in the Netherlands. In the case study the search is primarily aimed at criteria that determine the clopenness of software used in the ecosystem and whose software is able to communicate inside and outside the organization. The results from the study are used to shape and formulate clopenness in the transport and logistics ecosystem, both software- and businesswise. Note, the relation between organizational- and software- clopenness and the business model is not discussed in this paper. First we will outline the situation and context of the case studies, following with an example on how clopenness is interwoven within the transport and logistics ecosystems.

The transport and logistics sector is a highly dynamic sector where the profit margins are under pressure. In this ecosystem, a shift from closed to more open systems is noticed; the environment, i.e. customers, demanding organizations to be more open about various processes. Transport organizations do not know how to control clopenness, they are ‘victims’ of the supply chain. However, they are willing to participate in this shift; giving or providing clopenness, but also expecting customers to be more open to improve their own processes. We need to enclose the remark that one of the four case studies was performed in the industry’s association Transport and Logistics Netherlands (TLN) to provide an additional rich background for the other case studies. The carefully selected organizations for the case studies are in a range from small to medium to large transport organizations, whose services are mainly freight by road through Europe. The carried goods vary from a low to high value, e.g. paper, raw materials, shoes or electronics. The revenues come mainly from the distribution of these goods. The organizations have a transport management system (TMS), which supports order-intake, planning, dispatching, monitoring and transaction. Upon this system a fleet management system (FMS) supports real-time monitoring of the fleet and communicates with the TMS. In one of three organizations, the TMS and FMS are from the same supplier. Communication between systems and environment is established with use of electronic data

interchange (EDI) and other data interchange techniques or standards, e-mail, a web portal, or via traditional channels such as the telephone.

Software is developed in a software ecosystem within a business ecosystem, as without business there is no software and no software without business or innovation. Clopenness is intertwined in the connections between the different actors in the ecosystem and influences the activities and transactions generated at these connections. The transport and fleet management software is developed for the Microsoft Windows ecosystem or platform. Bosch [3] identifies such an ecosystem as an operating system centric software ecosystem, in which “third party developers build applications that offer value for customers”. Microsoft is owner of the platform and is therefore seen as the keystone of this ecosystem that retains a keystone strategy. The key interest of the keystone is improving or regulating the health of the ecosystem, and where the intertwined organizations, i.e. ecosystem participants, benefit from this ecosystem and are dependent on this specific ecosystem [13]. One of these participants is the independent software vendor (ISV), which is characterized as a niche player. A niche player usually produces a function that customers require, its existence is based upon the fact that keystones cannot produce all possible functionalities [3]. The ISV handles a disciple or follower strategy, because the developed software is committed to only one ecosystem [9]. The ISV produces transport and fleet management software, which supports the activities and transactions of a certain organization; in this case a transport organization. Complementary, Bosch [3] noted that the ecosystem consists of “the set of software solutions that enable, support and automate the activities and transactions by the actors in the associated social or business ecosystem and the organizations that provide these solutions”. Clopenness is an essential interwoven concept in this software ecosystem and is involved in for example the relationship between the ISV and customer; they are each other’s dependants. However, this specific situation will not be discussed in this paper. As said, business and software ecosystems are a blend, the software development ecosystem is intertwined with the transport and logistics business ecosystem. End-users or customers license the solutions brought forth by the ISV. After all, the customers are the actors using the functionalities of the software solutions that are built on the Microsoft platform. The transport organization can benefit from how they control clopenness with the help of software. Building on the definition of a software ecosystem by Bosch [3] we can shape the end-user or transport ecosystem as follows: the set of actors or systems, i.e. businesses, suppliers and customers, of which the interrelated activities and transactions in this transport and logistics commercial ecosystem are enabled, supported, and automated by a set of software solutions and other social connections between the actors such as exchange of information and knowledge; a system of systems [1]. Within this definition, clopenness plays an essential role. We view the concept of clopenness both software- and businesswise of a system, a transportation organization, situated in the transport and logistics ecosystem. The interwoven nature of clopenness in this ecosystem, in specific in the latter (present) situation and context, is expressed in the following section, using the CAM, described by means of the narrative method.

5.2 Case Study: Transport and Logistics Ecosystem

The following example situation, the monitoring process of a transport organization, was present in each of the cases. Cross-case synthesis and replication logic were applied to compose the example.

1. Availability. The monitoring process is an activity performed by the transport company and services the company and customer with monitoring information about the transport itself. The process provides a continuous stream of real-time information about the fleet and its transported goods.

2. Accessibility. The process can be accessed by a customer through a web-portal by means of a certain password or code, pushed or pulled through EDI or telephone. Making the process and its information available externally has become obligatory; customers will be dissatisfied when the transport company decides to make this information undisclosed. Please note the goods are property of the customer, they have the right to know what is happening with their property. The process can only be accessed timely through a web portal, for the period of shipment and aftercare. Not any interested person can access the monitoring process, because it is confidential information intended for customers only. Meaning that the status towards the company is taken into account, no competitor or other third party can access the process, regardless of the field of endeavor they are in or any other reasons. Making this process accessible for others than customers would create an unnecessarily dangerous situation; confidential information will be exposed, in the worst case leading to a high cargo theft risk because any interested person could track and trace the trucks. The effects of increased accessibility are clear; lost revenues and irreparable relationships with customers. Making the process not even accessible for customers will dissatisfy them at least, they tend to look for other companies providing the informational needs they desire from a company.

3. Transparency. Usually, information is omitted regarding the accessibility mode. In this case, content is filtered based on the customer who accesses the provided information from the monitoring process. The customer can only see his shipment information, i.e. no other shipments nor the goods of other parties inside the same truck or the load capacity in use. If the latter information is not omitted by the accessibility mode, then customers may notice that the truck isn't even full while the company said so; resulting in low trust. Also, valuables from other customers are exposed, which should obviously not be the case. Another effect caused by exposing more information is that the customer may come to know that a truck drives the same route with two shipments, of which only one is his own, and because of this insight the customer might propose the transportation company to split the distribution costs. When that happens, the transportation company has no advantage anymore, because they can't book the miles twice and invoice two customers with this. Therefore, transparency should be carefully managed. Another example of this occurs in dedicated drives, where the truck drives a fixed route for a customer a few days per week. The transport company hides information such as the truckload and real-time truck tracking, e.g. tracking on a visual map. They know there can be other customers on the same fixed route, for which they can carry some goods. That could save the company costs and generates additional revenues; the company can raise twice the price. The transport company does this in such way that the customer does not notice,

by omitting information. After accessing the right information, shipment information itself can be limited by certain factors. The information is not encrypted because it is already protected by means of the accessibility mode. It can be beneficial for the company to expose additional shipment information which meets the informational needs of the customer, by providing a few of the essential value propositions: estimated time of arrival, information on delivery, information on department, and proof of delivery. Additional information could be status codes such as loading, driving and unloading, but also reason codes; e.g. if you did not deliver then why did you not deliver, if the goods are damaged then why are they damaged, if the goods are late then why are they late, e.g. road closed, traffic jams, etc. The information is clear, comprehensible and unambiguous for the customer and, using the reason codes, also provides insights in the decision making process. Also, this information is not send in raw data, but is clearly documented and represented. A customer has the right to receive this information, after all the goods are his property. If information is made less clear or fails to give proper insights in the decisions, the customer will most probably drop its shipper. Experiences from other parties, negative or positive, are not presented. This is the companies' knowledge, "you will not give a customer exact details on this". Negative publicity is not convenient when profit margins are under pressure. Furthermore, the frequency of information disclosure to customers is well suited to the informational needs of the customers; they can be informed at all times, on all purposes, and through various channels.

4. Reciprocity. Transparency is one side of the story, the other side of the story is that synergistic advantages can be achieved by involving the customer in the process. Communications between the customer and company are enabled through various channels, such as EDI, telephone or a web portal. However, this is no guarantee that anything is actually being done reciprocally. Customers only want information, they mostly do not provide information as input for the transport companies processes. However, this information is gained via-via through the truck drivers, so they have sufficient information in order to modify or update the shipment information. The feedback for the customer is given at a satisfactory frequency and includes status and reason codes. However, customers could give more feedback, for example about the availability of unloading docks, enabling truck drivers to rearrange their schedule and deliver another shipment in the same city first when the unloading docks are full. The feedback frequency is limited by willingness of the customers, there are few agreements on this issue.

5. Licensing. In this monitoring process of the type organizational clopenness, licensing is not applicable. It is therefore not taken into account as it cannot cause effects in this case.

In this section we have shown how the CAM is applied and works in the present situation; the transport and logistics ecosystems. All narratives and decisions in it are taken within the defined concept of clopenness. From the case studies we have derived conclusions about clopenness in general and the interwoven nature of it in ecosystems which are described in the next section.

6. Discussion & Conclusion

Discussion. The model is not the ‘holy grail’, and it is noted that some external factors may come into play such as trust, frankness and honesty. It is not clear how these factors may affect the different elements in the model, such as the clopenness key attributes. Several interviewees noted that trust management is likely to be closely related to clopenness of systems and their benevolence. Regarding the CAM usage, the guidelines should not be used mechanically, so it could be practical to create guidelines for specific assessment situations or ecosystems.

Conclusion. Completely open or closed systems, software, or organizations do not exist; openness is considered a false dichotomy. The holistic view of systems should be addressed; i.e. the functional relation between the parts and the whole, discussing about clopenness as a property of these relations. Trying to create awareness of clopenness, understandable for everybody, seems easy. However, the underlying ideas are complex. This paper highlights the complexity of the phenomenon clopenness, elucidating this phenomenon by the construction of the Clopenness Assessment Model (CAM). The latter model is evaluated positively throughout the case studies. Essential in this were the assumptions or beliefs of the interviewees about the model and its message. It is agreed that the CAM largely conveys or represents the genuine message of clopenness with respect to organizations and software. Thereby, the model stimulates and provokes discussion about the ‘open vs. closed dichotomy’. Furthermore, the interwoven nature of clopenness in the transport and logistics ecosystem has also been investigated. Keypoint is that it is about understanding the situation and the decisions made within the defined phenomenon clopenness. One cannot just open up or close everything in their system, since generally the environment cannot accept that. The CAM, and in particular the generated stories, help to understand this. Relationships with the environment, e.g. transport organization - customer, can only be efficient or useful when they are reciprocal. The customer opens up the complementary process insights and participates in the process. A high degree of clopenness in the right parts of these systems would be helpful to be responsive and pro-active. Also, this ecosystem will face a rise of inter-company-networks in the near future, which should be able to communicate based on a high clopenness degree of the communicational processes in order to profit from synergistic advantages. Clopenness should never be the goal, but be used as an appliance.

Future research. In general, the principles of the general systems theory can provide us with a conceptual framework for future studies on various systems, such as information systems and complementary ecosystems, in which the main point is that the connections between the parts make the whole greater than the sum of its parts. Currently, the link between clopenness and business models is being investigated, evaluating the effects of clopenness changes on the business model of an organization.

References

1. Ackoff, R. L.: Towards a System of Systems Concepts. *Management Science*, **17**, **11** (1971) 661-671
2. Blom, T., & Haas, B.: De Ondraaglijke Lichtheid Van Systemen; Over De Grondslagen Van Het Luhmanniaanse Denken. *Tijdschrift voor Sociologie*, **17** (1996) 187-204
3. Bosch, J.: From Software Product Lines to Software Ecosystems. 13th International Software Product Line Conference, (2009) 1-10
4. Boudreau, K.: *how Open should an Open System be? Essays on Mobile Computing*. (2006)
5. Boulding, K.: General Systems Theory-the Skeleton of Science. *Management Science*, **2** (1956) 197-208
6. Cerri, D., & Fuggetta, A.: Open Standards, Open Formats, and Open Source. *Journal of Systems and Software*, **80** (2007) 1930-1937
7. Chesbrough, H. W.: *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business School Press, New York (2003)
8. Eisenmann, T., Parker, G., Van Alstyne, M.: Opening platforms: How, When and Why?. In: Gawer, A. (ed.) *Platforms, Markets and Innovation*. Edward Elgar, Northampton, MA, US (2009)
9. Hagel III, J., Seely Brown, J., Davidson, L.: Shaping Strategy in a World of Constant Disruption. *Harvard Business Review*, (2008) 1-10
10. Hocking, J. G., & Young, G. S.: *Topological Spaces and Functions*. In: *Anonymous Topology*, pp. 5. Dover Publications, New York (1988)
11. Hodgkinson-Williams, C., & Gray, E.: Degrees of Openness: The Emergence of Open Educational Resources at the University of Cape Town. *Emergence conference*, (2008)
12. Hylén, J.: *Open Educational Resources: Opportunities and Challenges*. OECD's Centre for Educational Research and Innovation, (2006)
13. Iansiti, M., & Levien, R.: Strategy as Ecology. *Harvard Business Review*, (2004) 1-11
14. Jansen, S., Finkelstein, A., Brinkkemper, S.: A Sense of Community: A Research Agenda for Software Ecosystems. 31st International Conference on Software Engineering, New and Emerging Research Track, (2009)
15. Laursen, K., & Salter, A.: Searching High and Low: What Types of Firms use Universities as a Source of Innovation. *Research Policy*, **33** (2004) 1201-1215
16. Maxwell, E.: Open Standards, Open Source, Open Innovation: Harnessing the Benefits of Openness. *Innovations*, (2006) 119-176
17. McNair, B.: *Glasnost, perestroika and the soviet media*. Routledge, New York (1991)

18. Mohsen, A., & Jansen, S.: Evaluating Architectural Openness in Mobile Software Platforms. Proceedings of the Fourth European Conference on Software Architecture: Companion Volume., (2010) 85-92
19. Moore, J. F.: Predators and Prey: A New Ecology of Competition. Harvard Business Review, (1993) 75-86
20. Orlikowski, W., & Baroudi, J.: Studying Information Technology in Organizations: Research Approaches and Assumptions. Information Systems Research, **2**, **1** (1991) 1-28
21. Pontin, J.: On Openness. Technology review, **112** (2009) 14
22. Riehle, D.: The Commercial Open Source Business Model. AMCIS 2009 Proceedings, (2009) 18-30
23. Rosen, L.: *Defining Open Standards*. **2010** (2004) 9
24. Takeda, H., Veerkamp, P., Tomiyama, T. et al.: Modeling Design Processes. AI Magazine Winter, (1990) 37-48
25. van Lier, B., & Hardjono, T. W.: Luhmann Meets the Matrix. Exchanging and Sharing Information in Network-Centric Environments. The 14th World Multi-Conference on Systemics, Cybernetics and Informatics, **3** (2010) 300-305
26. von Bertalanffy, L.: An Outline of General System Theory. The British Journal for the Philosophy of Science, **1** (1950) 134-165
27. von Bertalanffy, L.: The Theory of Open Systems in Physics and Biology. Science, **111** (1950) 23-29
28. West, J.: The Economic Realities of Open Standards: Black, White and Many Shades of Gray. In: Greenstein, S. and Stango, V. (eds.) *Standards and Public Policy*, pp. 87-122. Cambridge University Press, Cambridge (2007)
29. West, J.: How Open is Open enough? Melding Proprietary and Open Source Platform Strategies. Research policy, **32** (2003) 1259-1285
30. West, J., & O'Mahony, S.: The Role of Participation Architecture in Growing Sponsored Open Source Communities. Industry & Innovation, **15**, **2** (2008) 145-168
31. Xu, L., & Brinkkemper, S.: Concepts of Product Software: Paving the Road for Urgently Needed Research. First International Workshop on Philosophical Foundations of Information Systems Engineering, (2005)
32. Yannikaya, H.: Trade Openness and Economic Growth: A Cross-Country Empirical Investigation. Journal of Development Economics, **72** (2003) 57-89
33. Yin, R.: Case study research: Design and methods, Vol. 5. Sage Publishing, London, New Delhi (2003)