Considering Structural Properties of Inter-organizational Network Fragments during Business-IT Alignment

Novica Zarvić

University of Twente, Department of Computer Science, IS Group
P.O. Box 217, 7500 AE Enschede, The Netherlands
n.zarvic@ewi.utwente.nl

Abstract. Value exchange models can be used to reason about possible networked business constellations. Such inter-organizational business settings are determined in most cases solely from a financial point of view, i.e. by assessing the economic sustainability of the constellation. In this paper we discuss also other criteria that are relevant and should additionally be considered, namely the *structural properties of the inter-organizational constellation* itself. The multitude of possible inter-organizational business constellations – and underlying systems constellations respectively – makes it a necessary requirement to split such constellations into recurring structural patterns, which we call fragments. The structural properties are helping the designer to reason about quality related issues of the inter-organizational network, and may have an influence on design choices to be made. The paper suggests to design new e-business constellations not only on the basis of financial criteria, but to consider also quality issues of the inter-organizational network.

1 Introduction

A value web is a networked business constellation that is representing the inter-organizational business setting by considering value exchanges between independent business actors. e^3-*value* models [1, 2][1], as a graphical representation form of value webs, are based on the principle of economic reciprocity. This means that for every value exchange, something of value is expected in return. The e^3-*value* approach itself was introduced to a time when many dot-coms failed financially, because many companies wished for a 'slice of the cake' without assessing the economic sustainability of the business idea as a whole. However, we believe that it would make much sense to consider next to the financial criteria also other criteria. We suggest therefore to consider also the structural properties of the inter-organizational setting.

We are mainly focusing on IT-enabled value webs, which means that we have to consider multiple perspectives during development and implementation, which

[1] We assume that the reader is familiar with the e^3-*value* approach for e-business exploration.

in turn need to be well aligned. The business perspective is represented by an e^3*-value* model, which is put into operation by information systems (IS). This IS may provide functionality only inside a single company, or an IS may cross organizational boundaries. In the latter case it is called an IOS (inter-organizational system) and is used by at least two organizations. IOS, as a special class of IS, are from a technical point of view enabling inter-organizational business settings as present in value webs. The multitude of possible inter-organizational business constellations – and underlying systems constellations respectively – makes it a necessary requirement to split such constellations into recurring structural patterns, which we call fragments, and to investigate their structural properties.

For doing so, we focus in this paper on a structure and architecture-based perspective on the inter-organizational setting. Most definitions of *architecture* contain at least two essential parts, which are *components* and *relationships* between the components (cf. [3, 4]). This resulted in the following argumentation:

> *E-business constellations are complex inter-organizational business constellations, which often represent a composition of multiple architecture styles. Every* e^3-value *model, representing such constellations, can be decomposed into recurring fragments. We consider three fragment styles from literature and every conceivable fourth fragment style can be decomposed to one of the three. Each fragment style has its own properties that can influence certain quality attributes, which in turn can have positive/negative effects on the business.*

An architecture-based systems view seems very important, because it will enable us to consider structural properties of the enabling systems landscape. This holds for both levels, the business level and the IS level. We discuss several quality aspects of networks, such as *complexity* or *reliability*.

2 Review of existing architecture-based IOS classifications

Many different ways of classifying systems in inter-organizational settings are conceivable. We could for instance classify systems according to the functional area of application, e.g. payment systems, ordering catalogues. Other possible criteria for classification could be ownership, systems role, geographical application areas, access restrictions, architecture, etc. For instance, Johnston and Vitale [5] suggested an IOS classification framework that is based on four dimensions (business purpose, relationships with participants, information function, and improvement focus). The framework has a tree-like structure. Note that its main usage is not the provision of an IOS classification, but it rather represents a platform for studying essential factors and improvement possibilities (cf. [6]). However, we are interested in structure and architecture-based classifications (cf. [3, 4]). In the following subsection we discuss three such classification approaches, namely the classification by Barrett [7], Hong [8], and Kumar and van Dissel [9].

2.1 Barrett and Konsynski's IOS classification

IOS, as a term used in the IS research commmunity, stems from work done by Barrett and Konsynski in the early 1980s. They were initially talking about "inter-organization information sharing systems", and further were also the first to provide a classification scheme for such IS [7]. They defined five levels of participation for individual firms in IOS.

The *level 1* (remote I/O node) indicates that a firm hosts no IOS, but only accesses systems that are run, operated and maintained by other companies in the network. A *level 2* (application processing node) indicates the presence of a single system application used for a specific purpose, like for instance an order processing system. *Level 3* (multi-participant exchange node) participants are responsible for a network interlinking itself and any number of lower level participants with whom it has a business relationship. On *level 4* (network control node), the participants develop and share a network with diverse software applications that might be used by many different types of lower participants. Last, a *level 5* (integration network node) participant is defined as a data communication and processing utility, which can integrate any number of lower level participants and applications in real time. Note that each succeeding level may interlink with any combination of lower levels. Further, participants may restrict their sharing with respect to the applications. This means that a level 3 participant in system A may reduce its participation to level 1 in system B.

Relation to e^3-*value* models. Barrett and Konsynski's classification focusses on IOS from a participation perspective. Even the previously mentioned essential parts of architectures, i.e. components and relationships between those, can be found in this classification, its usage with respect to e^3-*value* is of limited nature, because the business perspective is not taken into account. An assignment to one of the levels can only happen, if we have sufficient knowledge about the underlying IOS, which is not always the case during the planning phase.

2.2 Hong's IOS framework

Another approach to an architecture-based classification of IOS is provided by Hong [6, 8]. His approach is based on what he calls the "value activity linkage". Value activities have previously been described in the context of value chains and value systems by Porter [10], who suggests that a company performs a set of value activities which as a whole determine the company's economic profit. Hong's IOS framework uses the concept of *horizontal* and *vertical* linkages (cf. figure 1 below) and argues that IOS are also often linked to firms in other value chains. Now, depending on the horizontal and vertical orientation of linkages, IOS are categorized as (i) horizontal IOS, (ii) vertical IOS, and (iii) cross IOS.

Relation to e^3-*value* models. Hong, in contrast to Barrett and Konsynski, discusses inter-organizational constellation from a business perspective by using Porter's value chain concept. The consideration of a business perspective

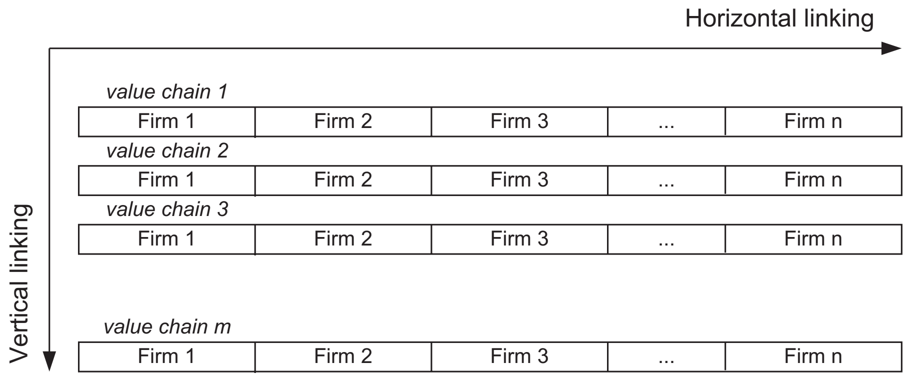


Fig. 1. IOS framework according to Hong [8].

and relating an IS (IOS) perspective to it, makes the framework at first view a possible candidate to build upon. But he neglects the fact that recent developments in information technology have allowed companies to create other forms of inter-organizational business cooperations [11], than the traditional value chain approach with its focus on industrial production processes. e^3-*value* models only show the business constellation under consideration. In case a business actor in a value model participates in other businesses, this has to be represented by other, additional e^3-*value* models. This can be explained by the fact that value activities, as a whole, need to lead to a positive economic profit when considering Porter's value chain concept. Value activities, which cause economic loss, are not problematic, if the margin is positive. In e^3-*value* models it is obligatory that all value activities lead to positive economic profit, simply because a company might be involved in more than the represented business constellation. The framework shown in figure 1 provides a grid-like structure, allowing to map all conceivable inter-organizational business settings to it, but it includes also firms that might not be involved in a particular business setting.

2.3 Kumar and van Dissel's IOS typology

The last architecture-based classification approach that we investigate and relate to the value web context is the IOS typology by Kumar and van Dissel [9]. Their typology consists of three types and is based on previous work by Thompson [12], who investigated among others inter-organizational interdependence. They view IOS simply as the technologies designed and implemented to operationalize the relationships between the partners [9, 8]. Kumar and van Dissel argue that the structurability of relationships influence the degree to which the relationship can be embedded in IOS. As a result the three types of interdependence and their level of structure are supposed to be reflected in the design of IOS. It is clear that a multi-actor business network heavily depends on several factors. Questions like

(a) 'Who provides the goods and/or services?', (b) 'Which way does the good follow towards the end consumer?' or (c) 'Is their supply secured?' indicate that coordination is a key requirement. More precisely, they are determining the structure, i.e. architecture, of inter-organizational constellations after sufficient exploration of such questions. This architecture-based classification seems to be a good candidate for our further analysis.

In the following we look more detailed at the three IOS types, which are the *pooled information resource IOS*, *value/supply chain IOS*, and the *networked IOS*. The first type, pooled information resource IOS, is representing an inter-organizational sharing of common IT resources. It is originally represented as a directed graph (cf. Fig. 2). Data movement is directed towards a central entity (cf. [6]), explaining the direction of the arrows in the graph. Next, value/supply chain IOS support the relationships between customers and suppliers. They operationalize sequential interdependency between companies. In an industrial production setting the value/supply chain IOS type could provide an answer to previously stated interdependence question (b). The last type, networked IOS, operationalize and implement reciprocal interdependencies between a set of organizations. This type is also represented as a graph, where each node is connected with all other nodes.

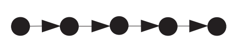
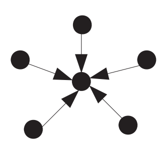
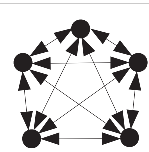
	Graph-based representation of architect. fragment styles	Support type and examples	Technology and lateral relationships
Chained architecture fragment		- Support of Porter's value system concept - e.g. EDI applications for IT-enabled supply chains	- Linked - First and last node bilateral, other nodes multilateral relations to two nodes
Pooled architecture fragment		- Marketplaces, shared databases, central production points - e.g. support usage of pooled resource, support of centralized production points	- Mediated and centralized - Nodes on the left have bilateral relations, node on the right multilateral relation to number of nodes on the right
Networked architecture fragment		- Networks - e.g. support of n-to-n communication between all participants	- Intensive - All nodes stand in multilateral relation to each other, number of each nodes' relationship is n-1

Fig. 2. System architecture fragments: based on Kumar and van Dissel's IOS typology.

We are solely interested in structural properties of the architecture, which was previously defined as components and relationships between them. These relationships represent system collaboration constellations, which might be the communication domain [13], where data movement is usually bidirectional. First, Thompson's patterns are referred to in the following as architecture fragments. An e^3*-value* model usually consists of a combination of architectural fragments, but there also exist many simple value web constellations that are made up of just one architectural fragment style. Historically, the *chained architecture fragment* appeared first in a networked context, because the purpose of the earliest software intensive-systems has been to support value chains [8], or more precisely value systems [10]. This style still can often be found in IOS that operationalize sequential supply chains. Next, the *pooled architecture fragment* can mostly be found, when looking at marketplaces where IT-supported business activities are centered around one business actor. A central database is a well-known example. We are of the opinion that next to the sharing character this architectural style can also be found in situations where a central actor is in need of several different resources from different firms in order to turn out a final product to be delivered to end consumers. So it is not representing a common repository, but rather a *centralized production point* like pointed out in figure 2. The last and most complex style is the networked architecture fragment. Its main characteristics is that all participants are in multilateral relationship to each other.

Relation to e^3*-value* models. This view seems more appropriate to the value web context than the previously described classification by Barrett and Konsynski, and later Hong. Barrett and Konsynski focussed on the level of participation in IOS, not on networked system collaborations. Hong on the other side considered system collaborations, but neglected the possibility of other constellations than provided by Porter's value chain. Volkoff et al. also state that advances in technology lead to a greater variety of inter-organizational constellations [11]. Our slightly adapted classification of Kumar and van Dissel's IOS typology allows for this, and certainly a value web might be built of a union of architecture fragments, other than the sequential value chain.

3 Inter-organizational Constellations from Multiple Perspectives

In this section we describe the relationship between an inter-organizational business constellation and an inter-organizational systems constellation. As already stated in the previous section, Kumar and van Dissel argue that the three types of interdependences identified by Thompson [12] and their level of structure are also likely to be reflected in the design of the IOS. In general the former refers to the business level and the latter to the systems level in an inter-organizational setting. These are clearly two distinct perspectives (or viewpoints) on the system that are hierarchically interrelated. Gordijn [14, p.35] states that the relationship between the IS level and the value-based business level is a "put into operation relation". In earlier research we analyzed documentation structures of several enterprise architecture frameworks and found that they also follow this argument. Take for instance the GRAAL framework or the Zachman framework. In both, the business considerations (diverse business models, organization charts, economic models, etc.) are located in the top row, whereas several system-related considerations (processes, data structures, software architectures, quality considerations, etc.) are placed in the rows below [15]. The relationship between these levels and concepts is discussed in the following by means of the systems hierarchy.

3.1 The Systems Hierarchy

System is a basic and fundamental concept in systems engineering, that enables us to reason about the world, or even parts of it. Remember that architecture definitions contain at least components and relationships between them. In our work the components might represent systems, subsystems, modules, etc. and the relationships are representing the way these components collaborate with each other, i.e. the relationship refers usually to the communication links. A subsystem is a system itself, and part of a larger suprasystem. This relationship can be represented by means of the system hierarchy [13].

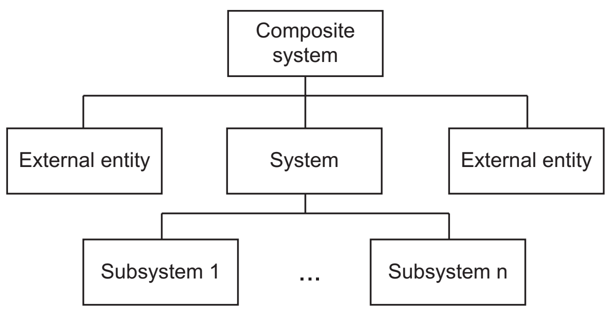


Fig. 3. The system hierarchy.

3.2 Composite systems, systems and subsystems in a value web context

In this subsection we show how the systems hierarchy with its concepts relates to the value web context. The top of figure 4 shows a simple e^3-*value* model. The model consists of three business actors, a seller, a wholesaler, and a producer, who exchange objects of value with each other. In fact the example represents a simple supply chain. Now, coming back to the system-subsystem line of reasoning, a value web is representing a(n inter-organizational) business system, which consists of subsystems, which in turn are represented by the business actors. These business actors themselves are systems and each business actor hosts several software-intensive systems, as can be seen at the bottom of figure 4.

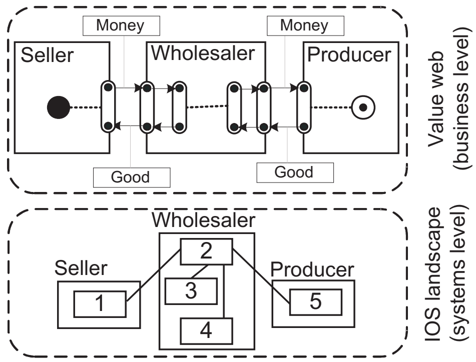


Fig. 4. Example of how an inter-organizational business constellation is enabled by underlying inter-organizational systems constellation

It is clear that, even for the simple value web constellation above, there are several different systems landscapes conceivable, which would operationalize the business requirements. Here we slightly disagree with Kumar and van Dissel, who argued that the three interdependence patterns are to be found *as such* in the underlying IOS [9, p. 286]. This is only true, if we view the set of all software-intensive systems inside a business actors as a composite system. But even this small example showed that we can not derive the IOS landscape on the basis of an e^3*-value* model. The inter-organizational business constellation on top consists of three nodes, which are the business actors.

The inter-organizational software systems constellations below consists of five nodes. In the example systems 1, 2, and 5 qualify as IOS, because they transcend organizational boundaries, whereas system 3 and 4 are IS, as far as they do not cross organizational borders. Now looking at the wholesaler, the sum of IOS 2 and IS 3 and IS 4, is the composite system. On the business level we are dealing with a chained/sequential architecture fragment (cf. 2), but on the systems perspective we are dealing with a combination of pooled architecture fragment (wholesaler) and chained architecture fragment. Clearly this requires, if possible, distinct analyses at both levels.

3.3 Fragments at business level: An example case

At the very beginning of the exploration phase of some new inter-organizational business ideas, the systems level will most probably be an unknown, so that at first sight only the value web level can be considered. For the identification of system fragments, the distinction of lateracy relationships is necessary. We distinguish bilateral and multilateral relationships, which are defined as follows.

Bilateral relation. A business system, and at the same time its composite IOS, is said to be in a bilateral relation, iff it is related to only one business system in the value web.

Multilateral relation. A business system, and at the same time its composite IOS, is said to be in a multilateral relation, iff it is related to more than one business system in the value web.

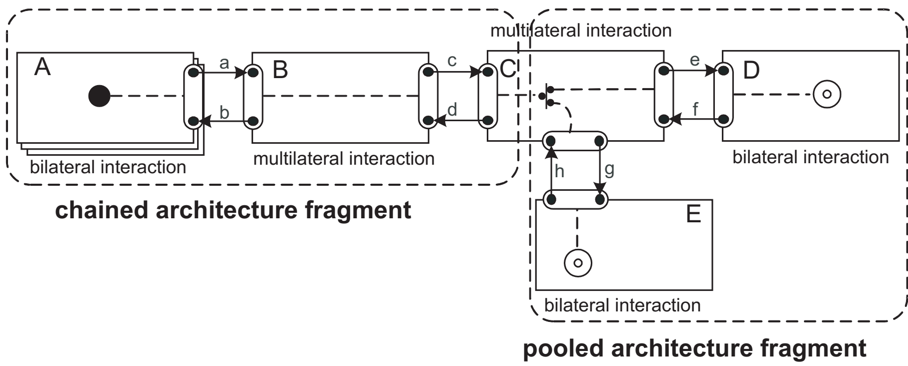


Fig. 5. Decomposing a value web into two architectural fragments.

In figure 5, a slightly more complex inter-organizational business constellation is represented by means of e^3-*value* . The value web consists of five business actors, but as a whole we can not identify the structure, as a structure present in Kumar and van Dissel's IOS typology. By splitting a value web into 'fragments', which ultimately explains choice of terminology, we get a mixture of one or more of the described IOS types. In figure 5, we can distinguish two architecture fragments, a chained/sequential and pooled architecture fragment. For doing so, we follow a simple 'splitting guideline', which says that the business actor node node with highest number of collaborations with other business actor nodes represents the decomposition point for splitting a value web into fragments. In graph-theory, this property is called the *degree of a node*. The degree of node C is 3, thus the highest in the value web of figure 5. The splitting node is to be considered, hence present, in both fragments.

4 Structural properties at IOS level: A case study

In the last subsection we described briefly, how to decompose complex inter-organizational constellations into recurring fragments that we know from literature. The same approach holds for the IOS level. In this section we focus on structural properties of the architecture fragments, and discuss possible success impacts. We do so by means of a recently performed case study in the area of non-governmental organizations (NGOs) [16, 17]. First, we describe the business case. Then, we shortly describe three basic structural properties from graph-theory, which are

(a) *degree of a node*,
(b) *size of a graph*, and
(c) *order of a graph*

We discuss, on the basis of these graph-properties, quality properties such as *complexity* and *reliability*.

4.1 The NGO case

The real-life example of Non-Governmental Organizations (NGOs) in the domain of international voluntary work [16, 17] is the basis for our further analysis. In this inter-organizational business constellation each NGO has the following administration tasks: It sends volunteers from its own country to projects in other countries and its own projects and it accepts volunteers from other countries in its own projects. The projects are provided by so-called partner organizations, who pay a fee for the volunteers sent to their projects. The volunteers pay a handling fee for being placed in one of the projects. The overall aim of this collaboration is to learn from other cultures and to help in local social development. Contact between the NGOs is maintained by a supranational umbrella organization that loosely coordinates the work of the NGOs. The following systems are present at each NGO: A general ledger system for the financial administration, a simple workflow management system (WFM) for performing the matching (i.e. placing each volunteer in a project), a project database of available projects, and a customer relationship management system (CRM) to manage information about the volunteers interested in voluntary work. Furthermore, each NGO has its own website. Since the NGOs are independent organizations, the implementations of these systems vary widely and do not provide compatible interfaces. In our example this represents the *old business constellation*, but one stakeholder in the network is studying the possibility to centralize the WFM systems at the umbrella organization and thereby taking over the matching process from the NGOs. This represents the *new business constellation*. The question to be solved is how this can be done such that the umbrella organization works profitable, while the NGOs perform better in terms of cost, quality, or both.

4.2 Graph properties

In order to be able to argue about quality issues that are due to the structural properties of the inter-organizational network, we describe in the following three properties from graph-theory. These basic graph properties are representing the basis for our argumentation.

Degree of node. The degree of a node is denoting the number of nodes it is connected with.

Size of graph. The size of a graph is denoting the total number of edges in the graph.

Order of graph. The order of a graph is denoting the total number of nodes in the graph.

4.3 Quality discussion of IOS fragments

Complexity. In a recent article, Rettig [18] describes that complexity is directly related to the number of systems that were added in an idiosyncratic way to the information technology landscape of companies. Further, she states that problems arise when companies try to integrate multiple data sources. We are dealing in the NGO case with the integration of multiple data sources, though it makes sense to investigate this quality property in our case. However, complexity cannot be found as a quality characteristic e.g. in ISO 9126 [19], because the standard focuses mainly on single software applications and many of the metrics in ISO 9126 are so-called *code metrics*. We are interested in *architectural metrics*, which measure the complexity of software components and their interconnections at architectural design level (cf. [20]), which is in a value web context the inter-organizational constellation at business level and IS level respectively.

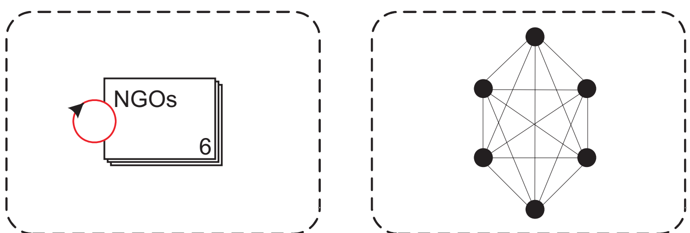


Fig. 6. Networked fragment: communication between NGOs.

The economic sustainability of the new NGO business constellation (cf. Sec. 4.1) was positively evaluated, but what remains is to analyze whether the changes may have some qualitative impact. Figure 6 shows on the left a small part of the overall value web constellation, namely the market segment NGOs, as it was in the old business constellation, where each NGO communicated directly with the other NGOs in the market segment. The communication took place via WFMs. In real life the NGO market segment has n = 23, but for reasons of space we represent the situation for n = 6. The communication between the NGOs is enabled by a *networked fragment*. The number of communication channels for the old business constellation is then $n(n-1)/2$.

Figure 7 shows the new business constellation, where the WFMs have been outsourced to a single WFM at the umbrella organization, so that one software

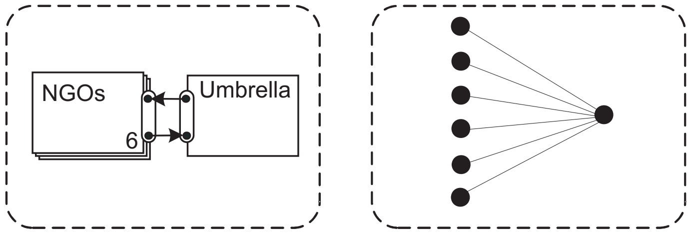


Fig. 7. Pooled fragment: communication between NGOs via umbrella organization.

system at each NGO communicates with the new centralized system. The edges on the left represent these system and the edge on the right represents the centralized WFM system. We are dealing now with a *pooled fragment*, where the number of communication channels is $n * m$, where $m = 1$. Now with respect to complexity, we can say that the pooled fragment is less complex than the networked fragment, because the number of communication channels has linear growth, opposed to the parabolic growth in the old constellation. Remember that we do not deal with 6, but with 23 NGOs, so that the size of the graph in the new business constellation is 23, but 253 in the old constellation. This represents a reduction of ca. 91% of the initial number of communication channels. The order of the graph on the other hand was increased by 1, which means that the pooled fragment has just one system more than the networked one. Summing up, even with the order of the graph of the pooled fragment slightly increased, we can say that the new constellation is much less complex, as the size of the graph – meaning the number of communication channels – drastically decreased.

Reliablity. When discussing reliability, we can refer to the previously described graph property degree of a node. Figure 8 confronts the two fragments with this property. It can easily be seen that each node in the networked fragment has the same degree, in real life $n - 1$, hence 22. In the pooled fragment we deal actually with a bipartite graph $G_{23,1}$, which means that the systems at NGOs have degree 1 and the WFM system at the umbrella organization has degree 23.

What do these numbers say about the reliability of the graph, i.e. the fragment under consideration? We refer again to the ISO 9126 standard and in contrast to the previously treated quality aspect complexity, we can find that, in Part 1 of the standard, *availability* is one of four subcharacteristics (maturity, fault tolerance, recoverability, availability) of *reliability*. Interesting to note is following fact. Part 2 of the ISO standard suggests several metrics for the subcharacteristics, but for characteristic reliability only metrics for the first three reliability subcharacteristics (maturity, fault tolerance, recoverability) are provided. However, *availability ratio* is mentioned as a metric under the reliability subcharacterstic *recoverability*. It is defined to be "the ratio of total available time of software to total observation time" [19]. The nodes represent systems

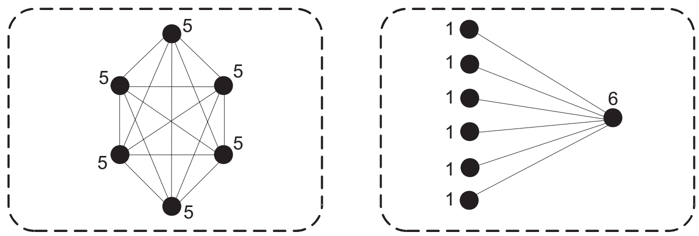


Fig. 8. Degree of node: comparing networked fragment with pooled fragment.

and are linked by edges, which in turn represent the communication channels that are needed for IT supported value object transfers. This means that the higher the degree of a node, the more systems communicate with this node. This means that the availability ratio might for each node be equal, which does not mean that the impact is equal. In figure 8 on the left (old constellation) all nodes have degree 5. Each system node is located at one actor, but when one of these nodes is not available for a certain period of time, this does not affect much the communication between the other nodes. The situation is completely different in the new business constellation on the right, where the system nodes at the NGOs all have degree 1, and the WFM system at the umbrella has degree 6. The NGOs communicate now via the centralized system on the right, meaning that they do not directly communicate anymore with each other. If one NGO system node is not available for a certain period of time, this will not effect the communication of the other NGO system nodes very much, but if the WFM system node with the highest degree is unavailable, then none of the nodes can communicate anymore. As a consequence, it is dangerous to solely consider the availability ratio of a single node, without to taking context of the fragment into account.

4.4 Graph-theoretical considerations

In figure 2 three types of system architecture fragments were considered. Graph-theoretically these are, in the undirected case, *paths* P_n, *stars* $K_{1,n}$ and *cliques* K_n. The reader may wonder whether the number of types discussed in this paper is not too restrictive. Another basic structure is the *bipartite graph*, where two sets of n_1, respectively n_2 nodes are considered, and edges only connect two nodes from different sets. The star $K_{1,n}$ is a special case of a bipartite graph.

We consider an arbitrary connected graph G, representing a communication structure. Suppose we have determined all cliques in G of order greater than 2. We then remove all the edges in these cliques, which gives a graph G^* without cliques of order greater than or equal to 3. G^* needs not be connected anymore. Now consider these nodes in G^* that have degree greater than 2. These are central nodes of stars. We now remove the edges of these stars to obtain a graph G^{**}. Graph G^{**} has only nodes of degree 0, 1, or 2 and therefore consists of single nodes, single edges and paths. The conclusion of this decomposition procedure is that the three types of fragments indeed suffice to describe any communication structure.

We should remark that reliability, as discussed in subsection 4.3, ties up with the graph-theoretical concept of *connectivity*. This concept is defined as the minimum number of nodes that have to be removed in order to disconnect the graph into two or more components. For cliques K_n for which disconnection is not possible, the connectivity may be defined to be $n-1$.

In figure 8 the networked fragment on the left has connectivity 5, whereas the pooled fragment has connectivity 1, the lowest possible value for a connected graph.

5 Summary and Conclusions

In this paper we have looked at structural properties of inter-organizational constellations that should be taken into account next to the assessment of economic sustainability. An inter-organizational business constellation might promise high revenues in the future, but this revenues might be negatively effected by the structure of the network itself. Therefore it is crucial to investigate whether a certain fragment or constellation respectively, may be for instance more fault tolerant than another. It is also advisable to keep the level of complexity (number of systems, number of communication channels) low, because this enables designers and maintainers to perform faster changes than at very complex constellations. Our results are applicable to all graph-based presentations of inter-organizational settings, including a business perspective and a systems perspective. In future research we aim to investigate how the described properties might influence previous positive assessment of economic sustainability. How do low fault tolerance or availability have a negative influence on financial figures?, is one question to be answered in this context.

Acknowledgment

The author is grateful to Roel Wieringa and Pascal van Eck for the discussions that lead to this work. He is also thankful to Cornelis Hoede for reviewing earlier versions of this paper. This research is supported by the Netherlands Organisation for Scientific Research (NWO), project 638.003.407 (Value-based Business-IT Alignment).

References

1. Gordijn, J., Akkermans, H.: Designing and Evaluating E-Business Models. IEEE Intelligent Systems **16**(4) (2001) 11–17
2. Gordijn, J., Akkermans, H.: Value-based requirements engineering: exploring innovative e-commerce ideas. Requirements Engineering Journal **8**(2) (2003) 114–134

3. L. Bass, P. Clements, and R. Kazman: Software Architecture in Practice. Addison Wesley (1998)
4. ANSI/IEEE Std 1471 :: ISO/IEC 42010: Recommended Practice for Architectural Description of Software-Intensive Systems. `http://www.iso-architecture.org/ieee-1471/ieee-1471-faq.html` (2000) last visited on 22-11-2007.
5. H.R. Johnston and M.R. Vitale: Creating Competitive Advantage with Interorganizational Information Systems. MIS Quarterly **12**(2) (June 1988) 153–165
6. I.B. Hong and C. Kim: Toward a New Framework for Interorganizational Systems: A Network Configuration Perspective. In: Proceedings of 31st Annual Hawaii International Conference on System Sciences YEAR =
7. S. Barrett and B. Konsynski: Inter-Organization Information Sharing Systems. MIS Quarterly **6**(Special Issue) (December 1982) 93–105
8. I. B. Hong: A new framework for interorganizational systems based on the linkage of participants' roles. Information & Management **39** (2002) 261–270
9. K. Kumar and H. van Dissel: Sustainable Collaboration: Managing Conflict and Cooperation in Interorganizational Systems. MIS Quarterly **20**(3) (1996) 279–300
10. M. E. Porter: Competitive Advantage. Free Press, New York (1985)
11. O. Volkoff, Y.E. Chan, and E.F.P. Newson: Leading the development and implementation of collaborative interorganizational systems. Information & Management **35** (1999) 63–75
12. J. D. Thompson: Organizations in Action. McGraw - Hill, New York (1967)
13. R. Wieringa: Design Methods for Reactive Systems. Morgan Kaufmann, San Francisco (2003)
14. J. Gordijn: Value-based Requirements Engineering: Exploring innovative e-Commerce ideas. PhD thesis, Free University of Amsterdam (2002)
15. N. Zarvić and R. Wieringa: An Integrated Enterprise Architecture Framework for Business-IT Alignment. In: Proceedings of Workshop of Business/IT Alignment and Interoperability (BUSITAL'06) at CAiSE'06. (2006) 262–270
16. P. van Eck, R. Wieringa, and J. Gordijn: Risk-Driven Conceptual Modeling of Outsourcing Decisions. In P. Atzeni et al., ed.: Proceedings of the 23rd International Conference on Conceptual Modeling (ER 2004). LNCS 3288, Berlin Heidelberg, Springer Verlag (2004) 709–723
17. N. Zarvić, R. Wieringa, and P. van Eck: Checking the Alignment of Value-based Business Models and IT Functionality (Accepted for the Requirements Engineering Track of the 23rd Annual ACM Symposium on Applied Computing – SAC 2008). (2008)
18. C. Rettig: The Trouble with Enterprise Software. MIT Sloan Management Review **49**(1) (Fall 2007) 21–27
19. ISO/IEC 9126: Information Technology - Software quality characteristics and metrics (ISO/IEC 9126-1 and ISO/IEC 9126-2) (June 1995)
20. J. Zhao: On Assessing the Complexity of Software Architectures. In: Proceedings of the 3rd International Software Architecture Workshop, Orlando, Florida, USA, ACM (November 1998) 163–166