Journal of Ambient Intelligence and Humanized Computing (2019) 10:239–253
https://doi.org/10.1007/s12652-017-0643-9

ORIGINAL RESEARCH

The interaction type approach to relationships management

Giancarlo Nota[1] · Rossella Aiello[1]

Received: 6 June 2017 / Accepted: 27 November 2017 / Published online: 19 December 2017

Abstract

One important paradigm proposed in literature by the scientific community that advocates the viable system approach is the structure-system dichotomy for the capability to simplify the investigation of a phenomenon and recognize its scope and implications. The purpose of this paper is to provide a contribution to the debate which concerns the structure-system approach introducing the concept of interaction type as a bridge linking the relationship concept to that of interaction. The interaction type is first introduced in its simplest form using an UML class diagram that points out its role when the observer views changes from structure to system and from system to structure. Then, the recursive nature of interaction types is discussed in order to show how complex relationships, that involve more parties at several interaction levels, can be better represented and used. Finally, a model for the representation of transactions is proposed in which a transaction type can be interpreted as a particular case of interaction type and a transaction as an ordered set of interactions. Two case studies are presented in the paper; the first concerns an example of fund transfer and highlights how the interaction type hierarchy can be applied. The second case study is from the port community system of South Italy and shows how a clearance to enter or leave national waters can be described as "departure from" and "arrival to" interaction types.

Keywords Viable system approach · Interaction type · Relationships management

1 Introduction

The problem of relationship management has received a great deal of attention from researchers and practitioners in several sectors of economy (Allen et al. 2009), public administration (Roberts 2015; Schellong 2009; Stromback and Kiousis 2011), social networks (Kilduff and Tsai 2006; Scott 2013) and many others (Waters and Bortree 2012). In general, the motivations behind this growing interest depend on the context in which the relationship management is pursued. Seeking long-lasting relationships with customers to achieve a competitive advantage in a given market (Lambert 2014), or involving citizens in the decision process on how public resources are used (Aiello et al. 2016), provide examples of the motivations that encourage the implementation of systems aiming at the relationship management.

Given such a variety of settings and applications, it is certainly not surprising that a universally accepted definition of relationships upon which to build a general theory of relationship management does not exist. Within each application domain, specialized models, languages and theories are developed so that they work in a satisfactory way with respect to the chosen domain but cannot be used when the application domain changes. Nevertheless, common concepts and characteristics connected to relationship management could be used to build a model that exploits the new concept of interaction type introduced in this paper.

Two approaches emerge from the analysis of the current scientific literature that can be called structural and behavioural, when the study of the relationship concept is taken into account. In the structural approach, first the definition of relationship is provided, then the concept of interaction is qualified in terms of "activation" of a relationship (orientation: from structure to behaviour). A scholar that adopt the structural approach to the study of a system focuses his attention first on the system structural properties; then the system is studied looking at its dynamic properties. In

✉ Giancarlo Nota
nota@unisa.it

Rossella Aiello
raiello@unisa.it

[1] Department of Scienze Aziendali, Management and Innovation Systems, University of Salerno, Fisciano, Italy

the behavioural approach, it is the opposite, the concept of interaction, usually described as an action that involves two or more entities, is initially stated and then the concept of relationship emerges as recurring interactions, mutual commitment and other properties (orientation: from behaviour to structure).

Examples of a structural approach to the definition of relationship can be found in Golinelli (2010) and Barile (2013). In his Viable System Approach, Golinelli defines a relation as a logical or physical connection between components of a structure, while an interaction represents the activation of a structural relation when resources and/or data are really exchanged between components in order to achieve a common goal.

The theories and models defined in the Relational Marketing (Gummesson 2012; Håkansson and Shenota 1995; Holmlund and Törnroos 1997) or Customer Relation Management (Chen and Popovich 2003; Zablah et al. 2004; Buttle and Maklan 2015) are representative of the behavioural approach to the definition of relationship. The initial focus is on the interaction, with the relationship then being perceived as a succession of long-lasting, repeated interactions equipped with characterizing properties. In these studies, it is the relationship that "inherits" the interaction properties from which the relationship is conceived.

Even though the research studies on the concept of relationship and interaction clarify the properties of two basic relationship management concepts, there are important aspects that have yet to receive sufficient attention in current literature. Assuming that there exists a relationship among two or more entities, there are generally many ways in which interactions reveal themselves as relationship activations. What is the structure of these interactions? What are the consequences on structure and processes of involved entities when we introduce a new interaction type among them? How does the comprehension of interaction nature contribute to the improvement of relationship management systems?

In this paper, the concept of interaction type as a structural notion that provides the link between the concepts of relationship and interaction is proposed. The purpose of the paper is twofold. On the one hand, a metamodel centered on the interaction type concept to reason on both is developed: the structural properties of relationships and the behavioral properties of interactions between active entities. We will show how the definition of interaction types linked to a relationship is the logical step that allows for a better understanding of the dynamic behavior that emerges from the structure of a system. On the other hand, we propose a method for the development of a relationship management software where a top down strategy guides the development. The method starts from the metamodel of interaction type, adds information to transform the metamodel into the model appropriate for the chosen application domain and adds further technical details to transform the model into code for the software application under development.

The literature review in Sect. 2 points out some research lines in the field of relationship management. In Sect. 3, we present our metamodel of interaction type discussing three views, represented as unified modeling language (UML) class diagrams that focus on different properties of interaction types. The main case study of this paper is presented in Sect. 4. It shows a real application of our metamodel to the design and realization of a port community system. After some methodological consideration, in which the implications of the interaction type metamodel are discussed in Sect. 5, the conclusion summarizes the main results of our research along with its limitations and suggests potential research areas to explore starting from the findings of this work.

2 Literature review

The work presented in this paper wants to contribute to the debate on the structure-system dichotomy that animates the viable system approach (VSA) community. The VSA is a scientific approach to business theory whose basis can be traced back to the works of Golinelli (2000), Barile (2000) and Barile (2013). According to VSA, every entity (a business or an individual) can be considered a system of many parts or structures made up of a group of interlinked sub-components, with the aim of realizing a common goal. The VSA comprises 10 fundamental concepts (FCs) that concern structure and dynamics of social systems. These fundamental concepts, together with several key related principles, can be found in Barile and Polese (2010), Barile et al. (2012) and in Polese and Di Nauta (2013).

For the purposes of this research we shall focus on the FC "Structures and systems":

> Every organization is characterized by a structure constituted of a set of individual elements with assigned roles, activities, and tasks that are performed in compliance with rules and constraints. From the structure a system can emerge by the activation of relationships into dynamic interactions with external supra-systems and internal sub-systems (Golinelli and Gatti 2001).

In particular, our research starts from the work presented in (Barile and Saviano 2011) that discusses the structure-system dichotomy as a fundamental aspect of system thinking. Their proposal contemplates a dual perspective—static and dynamic—of observing reality: the structure-system paradigm. In the definitions provided of relation and structure on the one hand and of interaction and of system on the other, a fundamental characteristic has been identified that consists in envisaging that every phenomenon can be interpreted from both a static and dynamic perspective.

The literature reviewed in this section is mainly from system theory and business relationship research. Articles from system theory provide a robust reference to interpret in a general setting the new concept of interaction type; articles from business relationship research provide useful references especially as far as the nature of interactions among parties is concerned. Most of the research work reviewed below discusses both the relation and interaction concepts since they are intertwined; therefore, the division between structural orientation papers and behavioural orientation papers might appear arbitrary. However, it allows to emphasize how the interaction type acts as a bridge between the concepts or relation and interactions and how the model can be explained in terms of structural and behavioural properties.

In the following, we will use the term relationship since it is widely used in several fields of social sciences limiting the use of the term relation to the discussion of artificial systems built by human beings.

2.1 Structural approach: from relation to interaction

To give a common and unified relationship definition is not simple because very often there is an implicit assumption that the reader understands what is meant by this term (Zolkiewski 2004). In everyday language, a relationship describes a connection or association between two or more people or things (Foo et al. 2008; Gummesson 2012). Relations can be observed among parts of an artificial system (Bruegge and Dutoit 2010) and relationships between individuals (person-to-person, interpersonal), between an individual and a firm or group of people (person-to-firm, firm-to-person), and between firms (firm-to-firm, interfirm) (Palmatier 2008).

The structural approach is a common one to the creation or understanding of artificial systems. Examples are: (1) the railway system where stations are parts and tracks are physical connections, interpreted as relations that connect the parts. The track activation, a train that runs from one station to another, is an interaction between the stations. (2) A Local Area Network (LAN) is made of computers and storage devices (the nodes of the network) and a communication channel among nodes is made of hardware (fiber or copper cables, network cards) and software (communication protocols). A point to point message from a computer to a storage device is a one to one interaction; a broadcast message from the sending computer to all the other computers in the network is a one to many interactions.

Starting from the assumption that an organization is considered as a physical structure provided with physical components interacting with each other with a specific purpose, Golinelli and Gatti (2001) state that a "relation represents a logical or physical connection between components of a structure". Great relevance for the organizational subsistence have the relations and interactions established by the organization with their suprasystems (Barile 2013; Golinelli 2010). The interaction is the activation of a structural relation when resources and/or data are really exchanged between the components and share their know-how in order to achieve a common goal (Golinelli and Gatti 2001). The interaction reflects a dynamic behaviour that transforms the structure into a system.

An important contribution that VSA can provide to relationship management in business is shown in Polese and Di Nauta (2013). The authors explore various perspective concerning how the concept of consonance, that is the structural compatibility between systems, and resonance, that is the virtuous interactions between parts, are essential to develop a harmonious and viable business behaviour.

Polese and Di Nauta also adopt the VSA: (a) to explore the theoretical connections between Service-Dominant Logic and Service Science from the point of view of relationships management; (b) to propose a model for the analysis of the relevance of a system. Relevance, stated in terms of influence and criticality, allows the top level management to decide the appropriateness of maintaining or implementing the underlying relationships with outside systems from the point of view of the survival prospects of a viable system.

2.2 Behavioural approach: from interaction to relation

The concept of relationship has many facets that led scholars to propose various definitions suitable to their research field. When we restrict our attention to business networks made of human being and organizations, the shared assumption is that a business relationship is based on repeated interactions between parties (Holmlund 2004). In other words, the studies that have a behavioural orientation first discuss the concept of interaction and then a relationship is defined in terms of interaction.

The studies based on the transactional marketing approach define an interaction as a mutual or reciprocal action between two or more parties (Gummesson 2012). This approach is focused on the products, with the interaction, usually called transaction or exchange, representing the transfer of goods, services, information or money between two parties. The exchanged items and the parties already existed and they remain unchanged during the transaction (Holmlund and Törnroos 1997; Ford et al. 2010).

The relationship marketing introduced in the late 1980s by the Nordic school (Grönroos 1994, 1999; Gummesson 2012) and IMP Group (Håkansson 1982; Ford et al. 2010; Holmlund and Törnroos 1997) challenged this traditional point of view, focusing on the concepts of the interaction, value, relationships and network. According to these schools of thought, the value for the business derives from the repeated interactions with the customers. An interaction is considered as a continuous process between actors over time; by means of successive interactions, actors, activities and resources may change and transform themselves.

Continuity, long-term and mutuality constitute some common characteristics that an interaction must have to become a relationship (Dwyer et al. 1987; Gadde and Mattsson 1987; Damkuviené and Virvilaité 2007). Trust, loyalty, commitment and satisfaction are other very common success variables that allow the parties to continue the business relationship (Damkuviené and Virvilaité 2007; Kleinaltenkamp et al. 2015).

2.3 The metamodels of interaction type

The metamodels presented in this section aim to introduce the new concept of interaction type as a bridge linking the relationship (structural concept) to that of interaction (systemic concept).

In order to describe our proposal, we first discuss the structural point of view. Considering the creation of a railway to connect two cities with the purpose of passenger transport, this requires the realization of an artificial system in which the components are the railway stations and the relation connecting them is a track. In such a system, a passenger train that travels from one station to another gives an activation of the relation between stations, that is, the track activation that can be interpreted as an interaction between stations. Several interactions are observable, each one with its proper description such as train number, passengers, departure time and other data.

Supposing that the designer wishes to realize a system that also allows freight transport. In this case, the freight train that runs along the track is the track activation. It is natural to observe that there are two different ways to use the underlying structure, the first for passengers and the second for freight transport, and that the need to introduce the freight transport service requires an important realization effort creating suitable structures for freight transport in both stations.

Changing focus from the structural to the behavioural point of view, we can make the following considerations. The researchers that study social sciences often approach the study of relationships starting from the interactions among people and then derive the main properties of a relationship. This is probably because, in contrast to the creation of artificial systems, organizations and people “already exist” and they come in contact due to mutual interests or common goals; therefore, the focus is on the interactions.

2.4 The interaction type as a bridge between relationship and interaction

The discussion above suggests that there is a need for the introduction of a new concept that can provide further insight when studying relationships and interactions; we shall call this concept interaction type. Our goal is to provide a formalized metamodel comprising the concepts of relationship, interaction type and interaction. The metamodel allows for a better understanding of structure and dynamics of a system and we will use it as a reference for the development of a relationships management software.

In Fig. 1, we show the UML class diagram that integrates the concepts of relationship between active entities, interaction type and interactions where:

- *active entity* is an organization, an individual or an automated component capable of performing a behaviour during the interaction with other active entities;
- *relationship* it represents a logical or physical connection between components of a structure. Through relationships communication becomes possible sustaining the interaction between active entities;
- *interaction* an interaction is a concrete action between two active entities in order to reach a goal;
- *interaction type* it is the structural element that gives form to one kind of interaction. This form qualifies one or more interactions in the sense that it provides external shape or configuration to the interactions of that kind.

The structure of an interaction type can be suitably represented considering the structure of messages between active entities. We shall adopt this interpretation in Sect. 3.2 to formalize the notion of simple interaction type.

A set of interaction types concerning active entities delimits either the kind of admissible actions between them or allows to circumscribe the attention to the kind of interactions in which we are interested. As far as our model is concerned, we focus on the message exchange due to its ability to convey data that represent the exchanges between active entities.

We consider two different times when we refer to a relationship: build time and activation time. At build time, the structure of the relationship is qualified as a set of interaction types. At activation time, the systemic behaviour emerges in terms of interactions between active entities. Such interactions are mediated by the interaction type in the sense that the communication that happens between active entities becomes possible when the structure of exchanged messages is compatible with that of interaction types.

On the one hand, approaching the diagram from the relationship side, we adopt the structural orientation. We can say that there is a relationship between two active entities

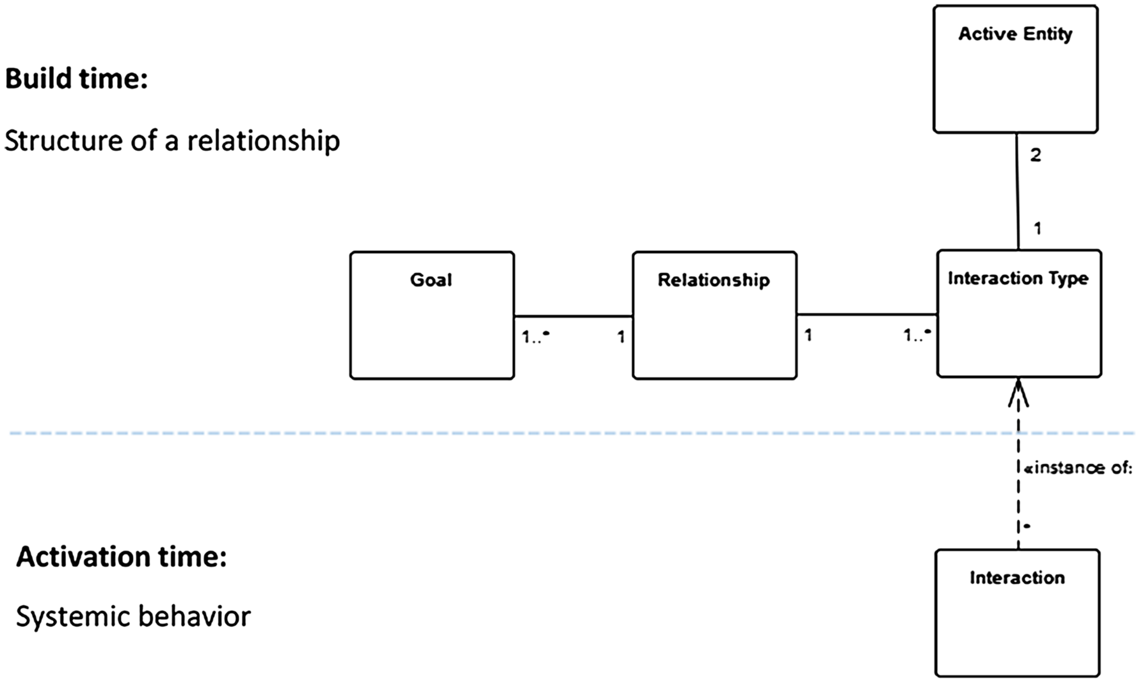


Fig. 1 Structure and behaviour of a relationship (UML class diagram)

when there exists at least one interaction type that connects them. Once the structure is built, the interactions can take place. On the other hand, if we approach the diagram from the interaction side, we can adopt the point of view of many marketing scholars that first focus their attention on the interactions. The behavior oriented relationship approach defines a relationship as repeated interactions with an organization where the interaction process consists of a multitude of exchanges and adaptation between active entities. The content of exchange can be products, money, social contact or information. From the analysis of a multitude of interactions, regular pattern of communications emerge; we synthesize them as interaction types and this set of interaction types provides structure to the relationship.

Example 1 This example shows the applicability of the model shown in Fig. 1, considering the problem of a purchase and consequent payment made by bank transfer. This problem is representative of a wider class of similar problems in B2B and B2C settings. Supposing that we have four active entities C1 and C2, where C1 is a customer of C2, and B1 and B2 are the banks of C1 and C2 respectively and focusing on the relationships in Fig. 2a, at level 0, we state that a relationship between C1 and C2 exists whose purpose is a payment from C1 to C2. At level 1, there are the relationships between clients and their reference banks, and finally, at level 2 there is the relation between the banks B1 and B2. Note that we use the term relation here because the interactions between B1 and B2 taken into consideration

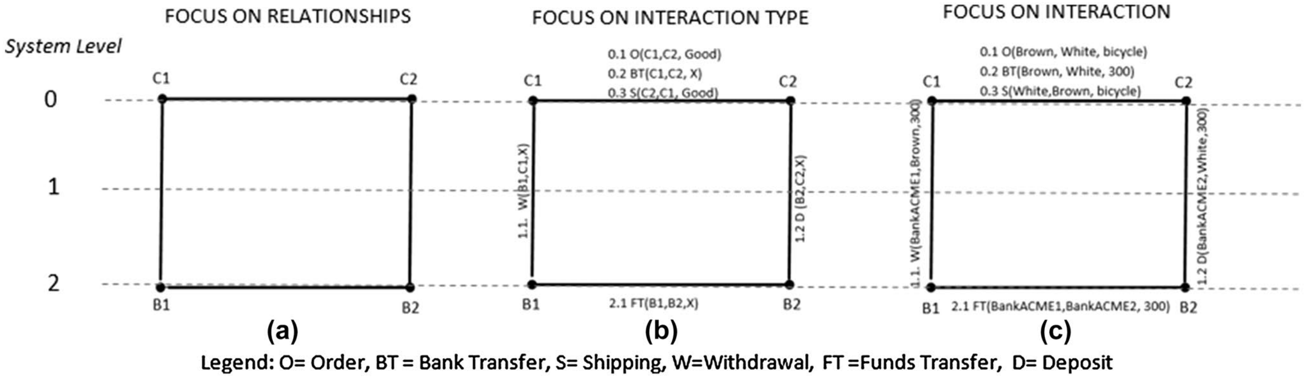

FOCUS ON RELATIONSHIPS
FOCUS ON INTERACTION TYPE
FOCUS ON INTERACTION
System Level
0
1
2
C1
C2
B1
B2
0.1 O(C1,C2, Good)
0.2 BT(C1,C2, X)
0.3 S(C2,C1, Good)
1.1. W(B1,C1,X)
1.2 D (B2,C2,X)
2.1 FT(B1,B2,X)
0.1 O(Brown, White, bicycle)
0.2 BT(Brown, White, 300)
0.3 S(White,Brown, bicycle)
1.1. W(BankACME1,Brown,300)
1.2 D(BankACME2,White,300)
2.1 FT(BankACME1,BankACME2, 300)
(a)
(b)
(c)
Legend: O= Order, BT = Bank Transfer, S= Shipping, W=Withdrawal, FT =Funds Transfer, D= Deposit

Fig. 2 The interaction type as a bridge between a relationship and an interaction

occur through software components only. Once the relation/relationships connecting the four active entities are identified, we can further refine our analysis on the structural properties of the system designed for bank transfers. Fig. 2b shows the interaction types associated to each relationship. The dyadic relationships (C1,C2), finalized to the purchase and payment of a good, can be expressed as a set of interaction types linked to the corresponding relationship at the appropriate interaction level starting from level 0. We have:

- 0.1 Order(C1,C2,Good)
- 0.2 BankTranfer(C1,C2,X)
- 1.1 Withdrawal(B1,C1,X)
- 2.1 Fund Tranfer(B1,B2,X)
- 1.2 Deposit(B2,C2,X)
- 0.3 Shipping(C2,C1,Good)

Finally, we can shift our attention to the interactions. Considering a single interaction episode, the value of data involved in the interaction are an instance of the schema described by the interaction type. For example, given the interaction type BankTransfer, then BankTransfer (Brown, White, 300) provides the instance data that represents the single interaction as a payment of €300; this payment is made from Brown to White by bank transfer to buy a bicycle. The semantics of the model of Fig. 1 also states that:

(a) several interaction episodes can be observed as particular instances of an interaction type;
(b) a set of interaction types can be linked to a relationship; this is the case of Order, Bank Transfer and Shipping that qualify the relationship (C1,C2).

Even in this simplified representation of a bank transfer, under the assumption that a payment is supported by a remote software application, it is worthwhile observing that:

- at level 0, active entities are persons or organizations. The interaction is among human beings;
- at level 1, C1 is a person and the behaviour of B2 is covered by a software component. The interaction is between a human being and a software component;
- at level 2, the interaction happens between two software components.

The capability of representing interactions among different kinds of active entities confirms the generality of our model even in the case when messages flow between organizational levels.

In general, the goal of a relationship can be further decomposed into smaller low-level goals. The achievement of these low-level goals allows for the achievement of the top-level one. The recursive interaction types introduced in the next section can represent the hierarchical structure of goals.

2.5 The recursive nature of interaction types

An interaction type can be eventually decomposed into one or more elements that are interaction types too. Figure 3 develops this aspect. The model is analogous to a well-known design pattern of Software Engineering called Composite that aggregates objects into a tree structure to represent parts as well as the whole hierarchy of objects (Gamma et al. 1994). However, the recursion on an interaction type has its peculiar characteristics that we describe in this section. According to the model, an interaction type can be simple or aggregate; an aggregate interaction type, at a given level, is the composition of interaction types at a lower level. The last level of the hierarchy contains only simple interaction types.

Since our proposal focuses on the messages exchanged in the context in which a relationship operates we define:

- *interaction type* it is either a simple interaction type or an aggregate interaction type;
- *simple interaction type* it represents the structure of an atomic interaction between two active entities in terms of exchanged messages:

$$itName = \{activeEntity_1, activeEntity_2, goal, \\ messageStructure, constraints\}$$

  where *goal* is something that you are trying to do or achieve and *constraints* acts as a guard for the activation of an interaction of this type;
- *aggregate interaction type*: it is a set of two or more interaction types:

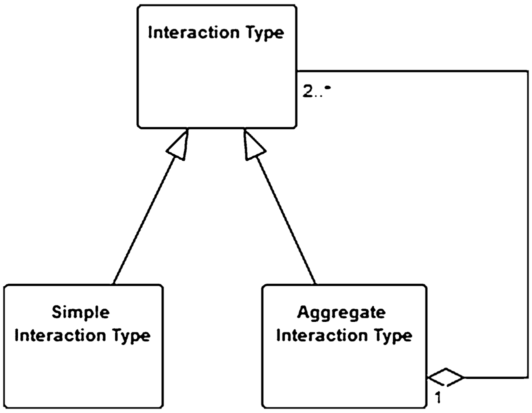


Fig. 3 The hierarchical nature of an interaction type

$$aitName = \{activeEntity_1, activeEntity_2, goal, setOfInteractionTypes, constraints\}$$

where *goal* expresses the goal of the whole set of interactions in *setOfInteractionTypes* and *constraints* can be applied to the elements of *setOfInteractionTypes*.

The *messageStructure* is analogous to that of a data record that we can find in a database management system in which the metadata and their corresponding types describe the record structure. Often, in software applications, *messageStructure* is expressed in the XML format known as XSD file that describes the structure of an XML document (see the case study of Sect. 4). If we consider the railway system, an example of admissible constraint is: "the train is operating on the track and the line is electrified". A common use of *constraints* in the applications concerns how many elements in *setOfInteractionTypes* are selected to extend the hierarchy at a lower level; in particular: (1) all the elements have to be selected; (2) only the elements that verify particular conditions are selected.

In order to show a practical application of the *constraints* construct, consider a variant of example1 in which the purchase is online and the payment can be made by choosing one among the following modalities: bank transfer, cash on delivery or prepaid card. Figure 4 shows the hierarchy generated from the aggregate interaction types OnlinePurchase and Payment. The bold lines indicate the activation of the interaction types when the chosen payment modality is bank transfer according to the constraints defined at level 0 and level 1. The names chosen for the interaction types in Fig. 4 evoke the corresponding goals.

Level 0—OnlinePurchase:

setOfInteractionTypes = {Order,Payment,Shipping}

contraints = select ALL.

Level 1—Payment:

setOfInteractionTypes ={Bank Transfer, Cash on delivery, Prepaid Card}

contraints = select(Bank Transfer, Cash on delivery, Prepaid Card).

2.6 Representing transactions with interaction types

The model of interactive and transactional environment presented in this section puts together the two previous ones and introduces the aspects concerning the representation of a transaction as a particular case of the aggregate interaction type construct. In Fig. 5, we show a unified schema in which the set of transactions is a subset of the interactions that occurs between active entities operating within an interactive environment. In other words, there are interactions that are not transactions.

- *transaction type* it is a particular case of aggregate interaction type whose components are arranged in a precedence graph:

$$ttName = \{ActiveEntity_1, ActiveEntity_2, goal, precedenceGraph, constraints\}$$

  where *precedenceGraph* is a directed graph G=(N,E) in which the set of nodes N={IT_1,IT_2,…, IT_n} are interaction types, and E={e_1,e_2,…, e_m} are directed edges. Each edge in the graph is of the form ($IT_j \rightarrow IT_k$), $1 \le j \le$ n, $1 \le k \le$ n, and there is one node in the graph for each interaction type. Furthermore, in G we distinguish a "beginTransaction" node and a "terminateTransaction" node.
- *Transaction* it is a numbered instance of transaction type composed by two or more interactions, one of which initiates the transaction (an instance of beginTransaction) and one that terminates it (an instance of terminateTransaction). The remaining interactions occur in the middle of [begin, end] transactions.

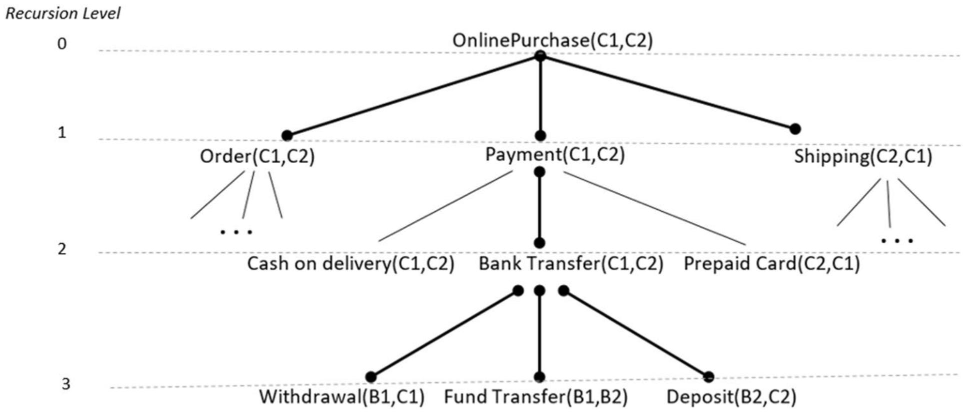


Fig. 4 Constraints driven recursion

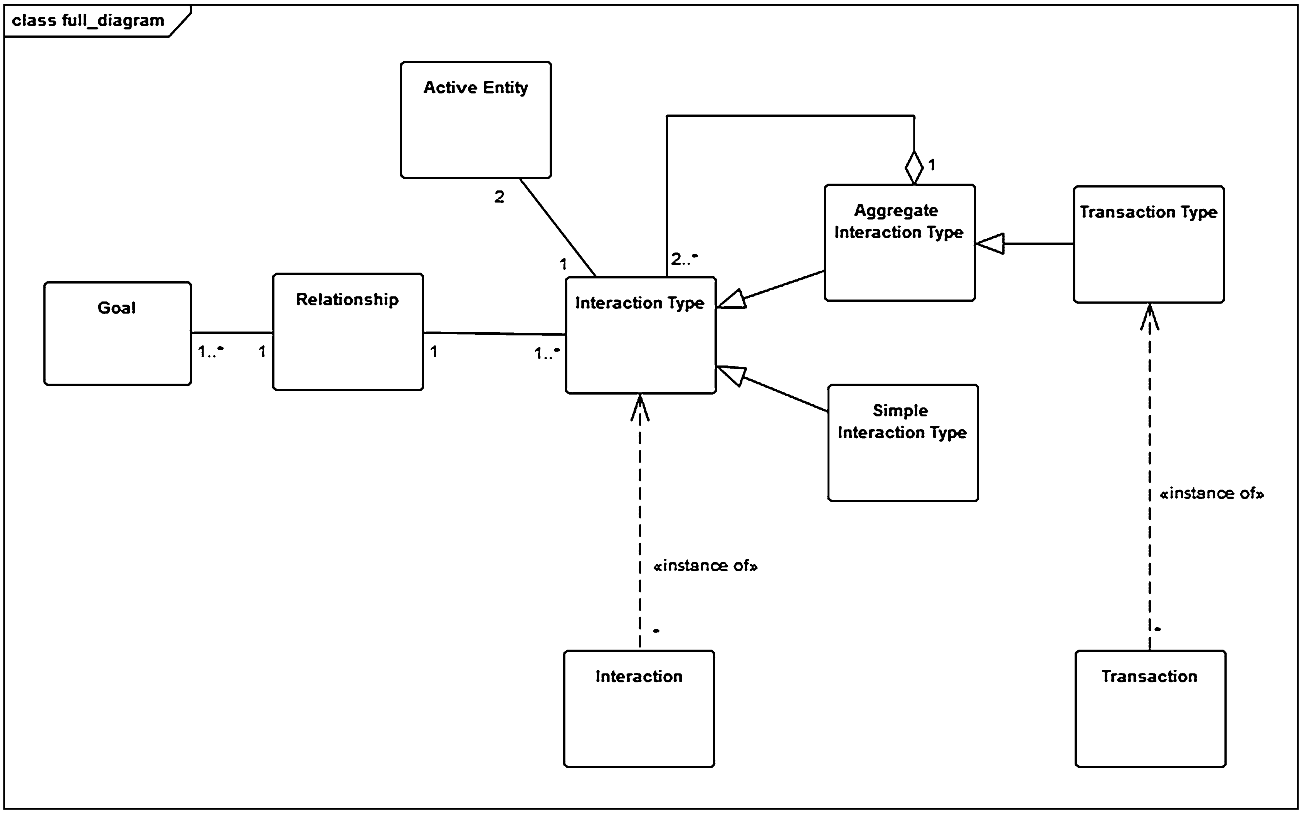


Fig. 5 The model for the representation of transactions

We say that the instance i_k derived by IT_k depends on the instance i_j derived by IT_j if i_k requires results of i_j, so that i_k cannot start its execution before the completion of i_j. Usually, the ordering key for the set of interactions is the starting time of each interaction in the transaction.

Example 2 Recalling example 1, Bank Transfer is a transaction type because it is an aggregate interaction type composed of five simple interaction types whose instances are executed in sequential order; c_1="C1 is a client of B1" represents the constraints, and the precedenceGraph in this example becomes an ordered list:

```
Bank Transfer(C1,C2) = { C1, C2, goal,
                          beginTransaction(C1,C2)→
                            Withdrawal(C1,B1)→ Fund Transfer(B1,B2)→ Deposit(B2, C2)→
                          terminateTransaction(C1,C2),
                         c1 }.
```

The simple interaction type Withdrawal is now defined as:

$$\text{Withdrawal} = \{\text{C1, B1}, \textit{goal}, \textit{withdrawalMessage}, \textit{withdrawalConstraints}\}$$

where *withdrawalMessage* contains the data necessary to define the interaction structure such as the account number, the withdrawal amount, the currency, and the value date while *withdrawalConstraints* c_2 = "balance(C1) ≥ amount(C1)" requires that the balance of C1 must be greater than the withdrawal amount.

The Fund Transfer and Deposit interaction types can be defined in a similar way; the *constraint* c_3 of Fund Transfer is c_3 = "B1 and B2 use interoperable software applications and the connection between B1 and B2 is operating" while the *constraint* c_4 = "C2 is a client of B2" is associated to Deposit.

Changing focus from structure to systemic behaviour, the interaction BankTransfer(Brown, White, 300) discussed in example1 is now a transaction.

The simple interactions activated during the relationship between Brown and White respect the structure of their corresponding interaction types. Each message in the transaction below is a set of key-value couples in which each element of the message structure is associated to the instance value. Furthermore, the integer used as a suffix to the name of each interaction uniquely identifies the interaction type; beginTransaction has the same suffix of the terminateTransaction to state that all the including interactions are part of the same transaction.

```
beginTransaction_095("Brown", White)

Withdrawal_123 = {  "Brown", "BankAcme1", goal="Withdrawal operation as part of BankTransfer No. 456557",
                                   [ accountNumber= "56245677", amount= 300,  currency= "€",
                                     recipientBank="BankAcme2",  valueDate="16/09/2016" ],
                             c2=true }

Funds Transfer_167 = { "BankAcme1", "BankAcme2",
                                   goal="Fund Transfer operation as part of BankTransfer No. 456557",
                                    [ recipientAccountNumber= "56245677", amount= 300, currency= "€",
                                     valueDate="16/09/2016"  ],
                                  c3=true }

Deposit_586 = { "BankAcme2", "White", goal="Deposit operation ending Bank Transfer No= 456557",
                          [accountNumber= "8740965", amount= 300, currency= "€", valueDate="19/09/2016"],
                          c4=true }

terminateTransaction_095("Brown", "White")
```

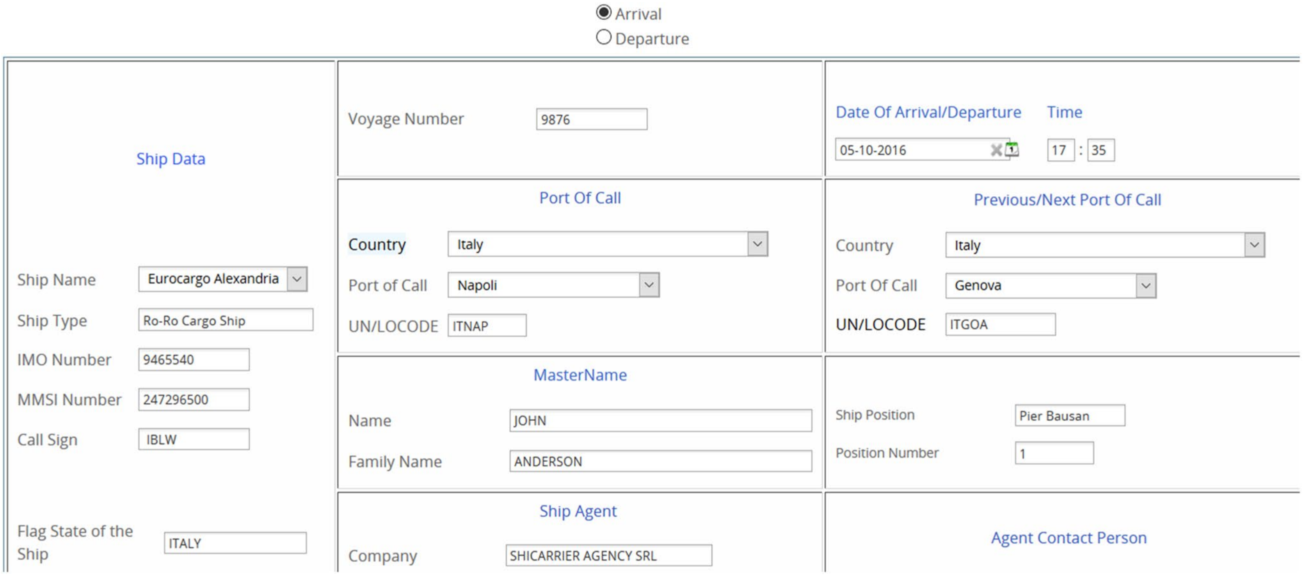
The *constraints* defined above act as guards and the transaction is allowed only when their value is true; note that due to the atomicity property of transaction, even the single interaction is executed only when all the interactions are executed; this implies that all the constraints must evaluate to true, so that the transaction is successful.

3 The case study

The application that we present in this section has been realized in the Smart Tunnel Project, financed by Italian Government, to realize a platform for intelligent logistic services of urban ports. The purpose is to reduce the bureaucracy on the storage time of goods and removing the inefficiencies of the bureaucratic interconnections improving the level of efficiency and sustainability of the transport of goods in urban areas. The application example shown below concerns the necessary fulfilments that a shipping company must do to achieve the required clearance to enter or leave national waters or clearance for a ship to berth (IMO 2011). The FAL Convention of International Maritime Organization defined a set of common requirements and standards for electronic transfer of information and standardized forms (FAL forms) aiming at the uniformity in procedures (IMO 2016a, b).

In order to apply the model proposed in the previous section, we focus our attention on the relationship between two active entities: the shipping company and the Port Authority (B2G); this relationship can be brought back to interaction types such as "Departure from" or "Arrival to" a port. In both cases, several requirements must be fulfilled before specific time points. For example, before the arrival to a port, a ship must send to the Port Authority the following documents (Pedersen 2012):

- 72h Pre-arrival : Paris MoU Report
- 24h Pre-arrival: Ship Pre-Arrival Security Information (ISPS)
  IMO Cargo Declaration (FAL2)
  IMO Crew List (FAL5)
  IMO Passenger List (FAL6)
  Imo Dangerous Goods Manifest (FAL7)
  Waste Declaration (WD)
- 2h Pre-Arrival: IMO General Declaration (FAL1)
  IMO Ship's stores Declaration (FAL3)
  IMO Crew Effect Declaration (FAL4)



● Arrival
○ Departure

Ship Data

Ship Name: Eurocargo Alexandria
Ship Type: Ro-Ro Cargo Ship
IMO Number: 9465540
MMSI Number: 247296500
Call Sign: IBLW
Flag State of the Ship: ITALY

Voyage Number: 9876

Date Of Arrival/Departure: 05-10-2016
Time: 17 : 35

Port Of Call
Country: Italy
Port of Call: Napoli
UN/LOCODE: ITNAP

Previous/Next Port Of Call
Country: Italy
Port Of Call: Genova
UN/LOCODE: ITGOA

MasterName
Name: JOHN
Family Name: ANDERSON

Ship Position: Pier Bausan
Position Number: 1

Ship Agent
Company: SHICARRIER AGENCY SRL

Agent Contact Person

Fig. 6 The FAL1 form implemented in SMART TUNNEL PCS

Figure 6 shows the FAL1 form that the shipping company must send to the Port Authority in order to get clearance.

Starting from these requirements, three interaction types are defined: 72hPre-Arrival, 24hPre-Arrival and 2hPre-Arrival. Another interaction type, called Clearance, defines the message structure that the Port Authority uses to grant or deny the permission to enter the port. Therefore, according to the model introduced in the previous section, ArrivalTo is an aggregate interaction type that allows for the activation of only one of its elements according to the journey duration:

At a lower level of interaction, each aggregate is a transaction type composed of a certain number of simple interaction types. The deriving interactions have to be executed before a certain time from the arrival. We report below the aggregate MoreThan72hJourney; the other two aggregate interaction types are similar. When a constraint is empty (there is no constraint acting as a guard), we assume that its evaluation is true.

ArrivalTo = {Ship, PortAuthority, *goal*, *setOfInteractionTypes*, *constraints*} with:

goal= clearance
setOfInteractiontypes={MoreThan72hJourney,MoreThan24hJourney, MoreThan2hJourney}

constraints= select(MoreThan72hJourney,MoreThan24hJourney,MoreThan2hJourney)

```
MoreThan72hJourney={Ship, PortAuthority, clearance, precedenceGraph, emptyConstraint}

precedenceGraph= {beginTransaction→72hPre-Arrival→24hPre-Arrival→
                                                        2hPre-Arrival→Clearance→terminateTransaction}

where

72hPre-Arrival = { Ship, PortAuthority, “clearance”,
                   ParisMouReport,
                   c1=“send before 72h from the arrival” }

24hPre-Arrival = { Ship, PortAuthority, “clearance”,
                  sequence(ISPS,FAL2,FAL5,FAL6,FAL7,WD),
                  c2=“send before 24h from the arrival” }


2hPre-Arrival = { Ship, PortAuthority, “clearance”,
                  sequence(FAL1,FAL3,FAL4),
                  c3= “send before 2h from the arrival” }

Clearance = { PortAuthority, Ship, “clearance",
                Clearance,
                c4=“Port Authority has received all required documents” }
```

The messages exchanged between a Ship and the PortAuthority are the documents required by the PortAuthority in order to release the clearance. A single document is sent during a “Clearance interaction” while a set of documents can be exchanged as shown by the 24hPre-Arrival interaction type. In this case, the message structure chosen is a sequence of documents (ISPS,FAL2,FAL5,FAL6,FAL7,WD).

In this case study the message structure has been implemented by means of XSD files, each one defining the structure of a document to be exchanged in a XML format. The graphic representation in Fig. 7 shows the FAL1.xsd structure used to render the FAL1 form of Fig. 6.

Each time that a ship of the company arrives in a different port, at least one of the previous transaction types is used. For example, suppose that the ship called “Eurocargo Alexandria” is executing a short voyage from Genoa toward Naples and that the estimated duration is 10 h. In this case, before 2 h of arrival, the ship is obliged to send the required document to the Naples Port Authority. In this case, an instance of transaction type MoreThan2hJourney is activated.

$$\begin{aligned}&\text{Journey\_AB12345\_2016\_10\_05}\\&\quad=\{\text{2h Pre-ArrivalToNaples},\ \text{Clearance54678}\}\end{aligned}$$

The interactions 2 h Pre-ArrivalToNaples and Clearance54678 are also created as instances of 2 h Pre-Arrival and Clearance interaction types.

```
2h Pre-ArrivalToNaples= { “Eurocargo Alexandria”, “NaplesPortAuthority”,
                        [FAL1_2016_10_05.xml, FAL3_2016_10_05.xml, FAL4_2016_10_05.xml],
                  startTransaction }

Clearance_I1 = { “NaplesPortAuthority”, ““Eurocargo Alexandria”, Clearance_54678.xml, commitTransaction }
```

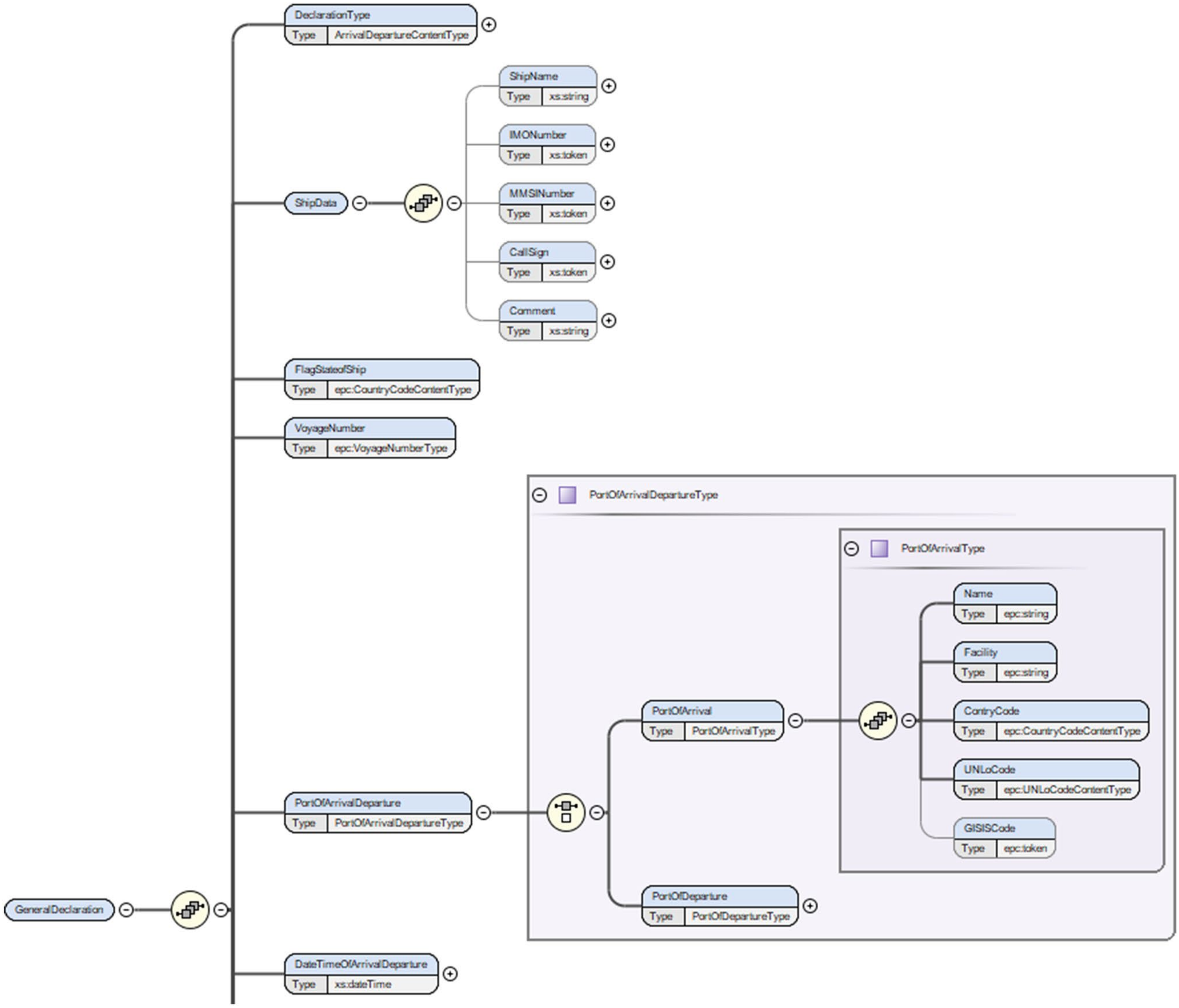


Fig. 7 The XSD structure of messages for FAL1 (Arrival/Departure Clearance)

An excerpt of FAL1_2016_10_05.xml is reported below. It represents the message exchanged between the ship and the port authority in order to allow the berth.

```
<ns0:GeneralDeclaration
      xmlns:epc="epc.xsd"
      xmlns:ns0="FAL1.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
<ns0:DeclarationType>Arrival</ns0:DeclarationType>
    <ns0:ShipData>
        <ShipName>Eurocargo Alexandria</ShipName>
        <IMONumber>9437907</IMONumber>
        <MMSINumber>247296500</MMSINUmber>
         <CallSign>IBLW</CallSign>
      </ns0:ShipData>
      <ns0:VoyageNumber>9876</ns0:VoyageNumber>
      <ns0:PortOfArrivalDeparture>
              <ns0:PortOfArrival>
                  <Name>Naples</Name>
                  <ContryCode>IT</ContryCode>
                  <UNLoCode>NAP</UNLoCode>
              </ns0:PortOfArrival>
          </ns0:PortOfArrivalDeparture>
          <ns0:DateTimeOfArrivalDeparture>2016-10-05T17:35:00.0
          </ns0:DateTimeOfArrivalDeparture>
          <ns0:FlagStateofShip>IT</ns0:FlagStateofShip>
          <ns0:NameOfMaster>
              <GivenName>John</GivenName>
              <FamilyName>Anderson</FamilyName>
          </ns0:NameOfMaster>
      .
      .
      .
```

4 Methodological implications

According to VSA, the systemic approach does not coincide with the holistic approach and is not opposed to the analytical-reductionistic approach (Golinelli and Barile 2008). Rather, it represents a path that, within a continuum that has in reductionism and holism its extremes, is able to reconcile both. Consequently, each result, that can contribute to a deeper understanding of the concept of relation/relationship, will have implications on the study of artificial and social systems by means of the systemic approach. Our research adds a further insight to the dichotomy perspective analyzed in (Barile and Saviano 2011) with respect to the following aspects:

1. Starting from the study of relationship-interaction, and hence from the structure-system point of view, it becomes possible a finer grain analysis, because the relationship is further decomposed in its "structural" constituents, that is the interaction types;
2. The identification and formalization of internal/external interaction types that are part of the structure of an active entity (i.e., an enterprise) allow different stakeholders to have a more precise comprehension about:

- the variety of interactions that happens within the system and between the system and its supra-systems;
- the goals that are relied to relationships.

The second point can be useful, for example, in the relevance analysis suggested by Polese and Di Nauta (2013). Since the interaction type proposal is aligned to VSA, it has an impact also with respect to other FCs. If we consider the FC "System hierarchy":

> According to the recursivity principle, every system "belongs" to a L level, and can recognize supra and sub-systems. Supra-systems will be identified at a

L + 1 level. Sub-systems will be defined at a L − 1 level,

the interaction type metamodel addresses this aspect as regards relations/relationships. As shown in Sect. 3.1 and 3.2, an interaction type is possibly decomposed into one or more lower level elements that are interaction types too; this approach can ease the system analysis. The recursive model shown in Sect. 3.2 is also able to represent the hierarchical structure of goals, showing how the achievement of low-level goals allows for the achievement of the top-level one.

The interaction type metamodel has also an interesting interpretation, when we look at the consonance property between two systems: the more interaction types comprise the relationship between active entities the greater is the consonance between active entities. Shifting view from static properties to dynamic behavior, many interaction types also mean that there exist several interaction possibilities between parties; when several harmonious interactions of various kind occur, then the resonance manifests itself at a higher degree.

A final consideration concerns the application of our metamodel to software design. The focus on relationship management is first considered at business level; here the designer focuses his attention on involved entities, business objects, exchanged messages and related goals at an appropriate abstraction level. If necessary, this analysis can be repeated applying the recursive metamodel of interaction type adding the necessary details. Finally, taking the developed models as starting point, the software realization is pursued by means of appropriate implementation languages. The case study of Sect. 4 shows a concrete application of this top-down development method.

5 Conclusions

Relationship management is a research field with several implications deriving from the different application domains and the variety of definitions, models and theories that have been proposed to approach the problem. Mature research results have clarified that the relationship concept is multidimensional. Damkuviene and Virvilaité (2007) point out the importance to study a relationship from one side as a repeated interaction with an organization and, on the other, considering the emotional bonds with an organization. Holmlund and Törnroos (1997) propose a classification of relational concepts according to three dimensions: structural (links, ties, connections, institutional bonds), economic (investments, economic bonds) and social (commitment, trust, atmosphere, attraction, social bonds). In this scenario, our contribution follows the research flow fed by the *VSA* community, in particular enriching the debate about the structure-system approach to the study of phenomena. The main contribution of this paper consists of the introduction of a novel concept called interaction type that can provide further insight when studying relationships and interactions. Through interaction types we can pursue a fine grain analysis on the structural properties of a relationship and we can define (or recognize) at different abstraction layer the goals of the relationship and the structure of messages that will be involved during interactions. Our approach is aligned with the VSA structure-system paradigm but suggests the possibility of a finer grain analysis of relationships and interactions through the decomposition of a relationship in its interaction types. The top-down method for the development of software systems shown in Sect. 4 is a natural consequence of the recursive nature of interaction types.

The three views presented in the paper can be used according to the characteristics of the considered problem. For problems where the relationships are described at one interaction level, the simple model of interaction type can be used, whereas the recursive and hierarchical model better describes situations where relationships and interactions must be taken into account at different levels. Finally, the transactional model can be used when the transactional properties of the system under specification are important.

The abstraction level of *VSA* implies both a simplicity of its concepts and a general validity of its framework (Golinelli et al. 2012) and it is at the right abstraction level to be accepted by research communities working in different fields. In the spirit of VSA, the concept of interaction type is provided at an abstraction level that should not pose constraints on its use. The interaction type is a metamodel that can be adapted to the needs of a given research field or extended to the case of a complex network of relationships. We are aware that in some cases the adaptation could be not so easy to obtain as in the case of the representation of the emotional bonding in a relationship. We believe that further research activities will take advantage of the formalization proposed in this paper, especially when the transformation from the metamodel to model stage must be taken into account so as to apply the model in a given application domain.

The lesson learned from the case study is that the class of software systems aimed at relationships management in settings such as B2B, B2C and B2A takes advantage of the application of our metamodel. However, further research is necessary to identify a complete software development methodology focused on relationships and interactions through the interaction type metamodel; this methodology could be used as a valid alternative to the traditional approaches based on functions or data, at least for software applications where relationships management prevails. This idea will be subject to future study.

References

Aiello R, Bisogno M, Nota G (2016) Accountability, transparency and open-data philosophy in the public sector: a knowledge network model. EGPA Spring workshop, At Modena

Allen S, Deragon J, Orem M, Smith C (2009) The Emergence of the relationship economy: the new order of things to come. Silicon Valley, California, USA, Link to your world

Barile S (2000) Contributi sul pensiero sistemico in Economia d'Impresa. ARNIA, Salerno, p 18

Barile S, Polese F (2010) Smart service systems and viable service systems: applying systems theory to service science. Serv Sci 2(1–2):21–40

Barile S (ed) (2013) Contributions to theoretical and practical advances in management. A viable systems approach (VSA), Vol. II. ARACNE, Roma

Barile S, Saviano M (2011) Foundations of systems thinking: the structure-system paradigm. In: Various authors, contributions to theoretical and practical advances in management. A viable systems approach (VSA). ASVSA, Associazione per la Ricerca sui Sistemi Vitali, International Printing Srl, Avellino, pp 1–26

Barile S, Pels J, Polese F, Saviano M (2012) An introduction to the viable systems approach and its contribution to marketing. J Bus Mark Manag 5(2):54–78

Bruegge B, Dutoit A (2010) Object-oriented software engineering using UML, patterns, and java, 3rd edn. Pearson, USA

Buttle F, Maklan S (2015) Customer relationship management: concepts and technologies. Routledge, India

Chen IJ, Popovich K (2003) Understanding customer relationship management (CRM): people, process and technology. Bus Process Manag J 9(5):672–688

Damkuviené M, Virvilaité R (2007) The concept of relationship in marketing theory: definitions and theoretical approach. Econ Manag 12:318–325

Dwyer F, Schurr P, Oh S (1987) Developing buyer-seller relationships. J Mark 51(2):11–27

Foo M-H, Douglas G, Jack MA (2008) Incentive schemes in the financial services sector: moderating effects of relationship norms on customer brand relationship. Int J Bank Mark 26(2):99–118

Ford D, Gaddeb L-E, Håkansson H, Shehota I, Waluszewski A (2010) Analysing business interaction. IMP J 4(1):82–103

Gadde LG, Mattsson LG (1987) Stability and change in network relationships. Int J Res Mark 4(1):29–41

Gamma E, Helm R, Johnson R, Vlissides J (1994) Design patterns: elements of reusable object-oriented software. Addison Wesley, Reading, MA

Golinelli GM (2000) L'approccio sistemico al governo dell'impresa. I—L'impresa sistema vitale. CEDAM, Padova

Golinelli GM (2010) Viable systems approach (VSA). CEDAM, Padova

Golinelli GM, Barile S (2008) Nascita ed evoluzione del pensiero sistemico in L'impresa come sistema—Contributi sull'Approccio Sistemico Vitale. Giappichelli, Torino, pp 27–43

Golinelli GM, Gatti M (2001) The firm as a viable system. SYMPHONYA Emerging Issue in Management, p 2

Golinelli GM, Barile S, Saviano M, Polese F (2012) Perspective shifts in marketing: toward a paradigm change? Serv Sci 4(2):121–134

Grönroos C (1994) From marketing mix to relationship marketing: towards a paradigm shift in marketing. Manag Decis 32(2):4–20

Grönroos C (1999) Relationship marketing: challenges for the organization. J Bus Res 46:327–335

Gummesson E (2012) Total relationship marketing. Elsevier, New York, USA

Håkansson H (1982) International marketing and purchasing of industrial goods. Wiley

Håkansson H, Shenota I (1995) Developing relationships in business networks. Routledge, London

Holmlund M (2004) Analyzing business relationships and distinguishing different interaction levels. Ind Mark Manag 33(4):279–287

Holmlund M, Törnroos J (1997) What are relationships in business networks? Manag Decis 5(4):304–309

IMO (2011) Guidelines for setting up a single window system in maritime transport—fal.5/circ.36. http://www.imo.org/en/OurWork/Facilitation/docs/FArelatednonmandatoryinstruments/FAL.5-Circ.36.pdf. Accessed Dec 2017

IMO (2016a) Fal forms and certificates. http://www.imo.org/en/OurWork/Facilitation/FormsCertificates/Pages/Default.aspx. Accessed Dec 2017

IMO (2016b) Guidelines for the use of electronic certificates—fal circ.39, rev.2. http://www.imo.org/en/ourwork/facilitation/electronicbusiness/documents/fal.5-circ.39-rev.2-guidelinesfortheuseofelectroniccertificates(secretariat).pdf. Accessed Dec 2017

Kilduff M, Tsai W (2006) Social networks and organizations. SAGE, London

Kleinaltenkamp M, Plinke W, Söllner A (2015) Theoretical perspectives of business relationships: explanation and configuration. Bus relatsh manag and marketing. Springer, Berlin, pp 27–54

Lambert D (2014) Supply chain management: processes, partnerships, performance, 4th ed. Fung Global Institute and World Trade Organization, USA

Palmatier RW (2008) Relationship marketing. Marketing Science Institute, Cambridge, Massachusetts

Pedersen JT (2012) One common framework for information and communication systems in transport and logistics: facilitating interoperability. In: Golinska P, Hajdul M (eds) Sustainable transport: new trends and business practices. Springer, Berlin Heidelberg, 165–196

Polese F, Di Nauta P (2013) A viable system approach to relationship management in S-D logic and service science. Bus Adm Rev 73(2):113–129

Roberts NC (2015) The age of direct citizen participation. Routledge, New York

Schellong A (2009) Citizen relationship management: a study of CRM in government. European University Studies. Frankfurt am Main, Berlin, Bern, Bruxelles, New York, Oxford, Wien, 2008. XX, p 242

Scott J (2013) Social network analysis. SAGE, Los Angeles, London, New Delhi, Singapore, Washington DC

Stromback J, Kiousis S (2011) Political public relations: principles and applications. Routledge Communication Series, New York

Waters RD, Bortree DS (2012) Advancing relationship management theory: mapping the continuum of relationship types. Publ Relat Rev 38(1):123–127

Zablah AR, Bellenger DN, Johnston WJ (2004) An evaluation of divergent perspectives on customer relationship management: towards a common understanding of an emerging phenomenon. Ind Mark Manag 33(6):475–489

Zolkiewski J (2004) Relationships are not ubiquitous in marketing. Eur J Mark 38(1/2):24–29