Communication Analysis: A Requirements Engineering Method for Information Systems*

Sergio España[1], Arturo González[2], and Óscar Pastor[1]

[1] Centro de Investigación en Métodos de Producción de Software
Universidad Politécnica de Valencia
{sergio.espana,opastor}@pros.upv.es
[2] Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
agdelrio@dsic.upv.es

Abstract. Developing Information Systems (ISs) is a hard task for which Requirements Engineering (RE) offers a good starting point. ISs can be viewed as a support for organisational communication. Therefore, we argue in favour of communication-oriented RE methods. This paper presents Communication Analysis, a method for IS development and computerisation. The focus is put on requirements modelling techniques. Two novel techniques are described; namely, Communicative Event Diagram and Communication Structures. These are based on sound theory, they are accompanied by prescriptive guidelines (such as unity criteria) and they are illustrated by means of a practical example.

Keywords: Communication Analysis, requirements engineering, communication theory, enterprise information systems, requirements structure.

1 Introduction

Information Systems (ISs) development and computerisation is a wicked problem[1]. To a large extent, this is due to their socio-technical nature [27] and to the intervention of multiple stakeholders with often conflicting needs and world views. To overcome conflicts and to reach agreement, stakeholders' perceptions have to be placed in a knowledge base that is shared with IS developers. Requirements Engineering (RE) facilitates this process by offering techniques for the discovery and description of stakeholders' needs. However, there exist many different explanations of what a requirement is and what it is not (e.g. what vs. how [10], why [35]). The authors' stance in this matter is summarised as follows (for more reasoned arguments see [20]):

– A requirements engineering method should prescribe a requirements structure that fits the problem trying to be solved, it should offer contingent and prescriptive methodological guidance and it should be illustrated with representative examples.

* Research supported by the Spanish Ministry of Science and Innovation (MICINN) project SESAMO (TIN2007-62894), the MICINN FPU grant (AP2006-02323), and FEDER.

[1] Among other characteristics, wicked problems do not have a unique solution and their statement is not clear until they are solved (in part due to stakeholder discrepancies) [34].

P. van Eck, J. Gordijn, and R. Wieringa (Eds.): CAiSE 2009, LNCS 5565, pp. 530–545, 2009.

– Requirements specifications should offer an external view of the system under development. In case internal details are included, the requirements structure should clearly differentiate the problem space (external) from the solution space (internal).

Attending to academic literature and industrial practice, various conceptions of ISs are found. Some authors consider ISs as a mere representation of reality [40]. Under this perspective, ISs can be described following an ontological approach; that is, focusing on the objects perceived in the universe of discourse (e.g. OO-Method [32]). Other perspectives focus on organisational intentions (e.g. Maps [35]), value object exchanges (e.g. e3-value [23]), etc. The authors view an IS as a support for organisational communications [26, 27]. Therefore, a communicational approach to ISs analysis is necessary; that is, we claim that ISs requirements engineering should take into account users' communicational needs.

We propose Communication Analysis as a method for the development and computerisation of enterprise Information Systems. This method focuses on communicative interactions that occur between the IS and its environment. The method stems from IS foundations academic research [22, 31] and it evolves by means of the collaboration with industry. Communication Analysis is currently being used by important Spanish enterprises and governmental institutions. The communicational perspective of the method has been overviewed in a previous publication [20]. This paper presents with greater detail the modelling techniques that Communication Analysis proposes for requirements specification. The main contributions of this paper are the following:

- Communicative Event Diagram is presented – a business process modelling technique that adopts a communicational perspective and facilitates the development of an IS that will support those business processes.
- Communication Structures are presented – a modelling technique for the specification of messages communicated with (and within) the organisation.
- Both modelling techniques fit well into a requirements structure, and they are both soundly founded on concepts borrowed from diverse disciplines (e.g. Systems Theory, Communication Theory, Information Systems Theory). Methodological guidance is based on sound criteria and illustrative examples are offered.

The rest of the article is structured as follows. Section 2 presents an overview of the approach, highlighting the proposed requirements specification structure and the method workflow. Section 3 describes the case that is used to illustrate the proposal. Section 4 describes Communication Analysis modelling techniques, paying special attention to Communicative Event Diagram and Communication Structures, and illustrating them. Section 5 presents a review of related works. Section 6 presents conclusions and future works.

2 Overview of the Approach

From a systemic point of view, the kind of problem that we are confronting involves at least three systems. The Organisational System (OS) is a social system that is interested in observing, controlling and/or influencing a portion of the world [27]. We refer as Subject System (SS) to the portion of the world in which the OS is interested (a.k.a universe of discourse). An Information System (IS) is a socio-technical system, a set of agents of different nature that collaborate in order to support communication between the OS and its environment (and also within the OS) [27]. We refer as Computerised Information System (CIS) to the part of the IS that is automated.

Therefore, we argue that systemic principles need to be applied to IS requirements engineering. Quite often, the set of requirements are organised as plain enumerated lists. We claim that a requirements structure suited to ISs development is more appropriate than a list. Communication Analysis proposes a requirements structure that allows a stepwise refinements approach to ISs description (by following systemic principles). Also, the proposed method allows tackling with static and dynamic perceptions of reality (by giving support to discovering and describing that duality). Figure 1 shows the first dimension of the proposed requirements structure and the activities that are related to each requirements level. The structure and the method flow of activities have been overviewed in a previous publication [20].

The first dimension concerns several (systemic) requirements levels. *L1.System/subsystems* level refers to an overall description of the organisation and its environment (OS and SS, respectively) and also involves decomposing the problem in order to reduce its complexity. *L2.Process* level refers to business process description both from the dynamic viewpoint (by identifying flows of communicative interactions, a.k.a. communicative events) and the static viewpoint (by identifying business

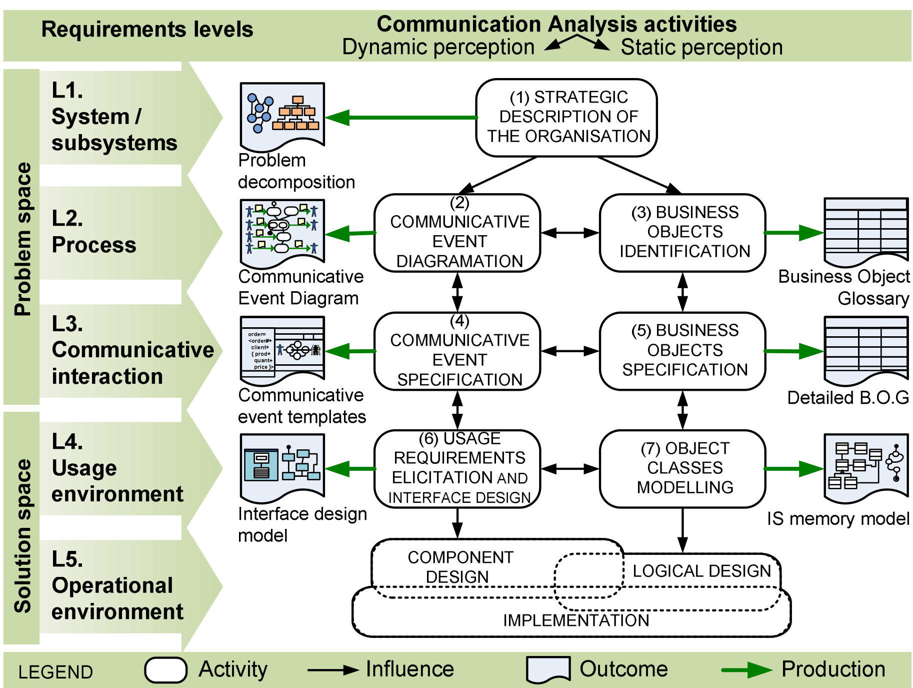


Fig. 1. Communication Analysis requirements levels and workflow

objects). *L3.Communicative interaction* level refers to the detailed description of each communicative event (e.g. the description of its associated message) and each business object. *L4.Usage environment* level refers to capturing requirements related to the usage of the CIS, the design of user interfaces, and the modelling of object classes that will support IS memory. *L5.Operational environment* level refers to the design and implementation of CIS software components and architecture.

Levels L1, L2 and L3 belong to the problem space, since they do not presuppose the computerisation of the IS and they aim to discover and describe the communicational needs of users. Levels L4 and L5 belong to the solution space, since they specify how the communicational needs are going to be supported. This paper focuses on the problem space and it describes in detail some of its modelling techniques.

3 Illustration Case Description

In order to exemplify the application of the method, we use an illustration case.

A photography agency manages illustrated reports (a.k.a. reports) and distributes them to publishing houses. Freelance photographers apply to work for the agency. The agency management board decides whether the photographer is accepted or not, and which quality level is assigned to them. Accepted photographers provide reports to the agency. Publishing houses buy reports from the agency catalogue and, sometimes, they request an exclusive report (a.k.a. exclusive) on a particular subject. Each exclusive is assigned to an interested photographer, as long as they do not have any other pending exclusive. Reports and exclusives are sent to publishing houses through a courier company, along with a delivery note. Then the messenger returns to the agency the delivery note, signed by the publishing house. Monthly, the agency issues publishing house invoices and photographers cheques.

4 Communication Analysis Modelling Primitives and Guidelines

Before describing Communication Analysis requirements levels, it is worth enumerating three functions of communication defined by Jakobson [25]. These functions allow us to structure requirements and to underpin the concepts underlying the method:

- Phatic: it aims to establish, maintain contact, and ensure operation of the (physical or psychological) communication channel between the addresser and the addressee.
- Referential: the purpose of this function is to convey context-related information.
- Connative: it aims to convey commands, to (attempt to) transform reality or people, to affect the course of events or behaviour of individuals.

4.1 L1. System/Subsystems Level

On the first requirements level, the analyst describes the OS from the strategic point of view. On the one hand, when the organisation is complex, it is advisable to decompose the problem into subsystems or organisational areas. On the other hand, the analyst elicits requirements related to strategic-level business indicators.

The photography agency case is of manageable size. Even though, three subsystems can be distinguished: Customer Service Department (it serves publishing houses), Production Department (it deals with photographers and manages reports), Accounting Department. With regard to strategic business indicators, the management board is interested in growth indicators that serve as a scorecard; e.g. increase in the number of photographers, increase in the number of exclusives, cash flow.

4.2 L2. Process Level

On this requirements level, Communication Analysis proposes describing business processes from a communicational perspective. The aim is to discover communicative interactions between the IS and its environment, and to describe them taking into account their dynamic and static aspects; that is, creating the Communicative Event Diagram and the Business Objects Glossary, respectively. In the following, a series of definitions clarify the concepts upon which the modelling techniques are built.

We refer as *communicative interaction* to an interaction between actors with the aim of exchanging information. FRISCO report [16] presents a generic model of ISs that considers an IS as a support for communicative interactions. In a previous publication, the authors extend this model in order to deepen the communicative point of view [31]. Depending on the main direction of communication, the following types of communicative interactions can be distinguished:

- *Ingoing communicative interactions* primarily feed the IS memory with new meaningful information. These interactions often appear in the shape of business forms.
- *Outgoing communicative interactions* primarily consult IS memory. These interactions often appear in the shape of business indicators, listings and printouts.

Industrial experience has shown us that ingoing communicative interactions entail more analytical complexity. Therefore, we advise the analyst to focus, first of all, on ingoing communicative interactions[2].

A *communicative event* is a set of actions related to information (acquisition, storage, processing, retrieval and/or distribution), which are carried out in a complete and uninterrupted way, on the occasion of an external stimulus [22].

Communication Analysis offers *unity criteria* to allow identifying communicative events, also facilitating the determination of their granularity. This way, a communicative event can be seen as an ingoing communicative interaction that fulfils the unity criteria. Each unity criterion is related to a communication function. Table 1 summarises unity criteria and their application, see [21] for detailed information.

Communication Analysis proposes to specify the flow of communicative events by means of the **Communicative Event Diagram** (CED). The primitives of this modelling technique are shown at the bottom of Figure 2 and explained next.

[2] Communication Analysis also takes into account outgoing communicative interactions. However, when the OS needs complex indicators for performance management, techniques such as the Balanced Scorecard are recommended.

Table 1. Unity criteria to identify and encapsulate communicative events

Criterion (Communication function) Definition
Trigger unity (Phatic function) Trigger responsibility is external. The event occurs as a response to an external interaction and, therefore, some actor triggers it. This (primary) actor is the one that provides the information that is conveyed in the event.
Communication unity (Referential function) Each and every event involves providing new meaningful information. Thus, an interaction needs to provide new facts in order to be considered an event. Input messages are representations of something that happens in the IS environment.
Reaction unity (Connative function) The event is a composition of synchronous activities; thus, these activities can communicate the information they need from each other. Events are asynchronous among each other; thus, events need a shared IS memory to communicate.
An example of their application. According to the unity criteria, two communicative events are identified with regard to photographer subscriptions: PHO 1 and PHO 3 (see Figure 2). Both events fulfil the three unity criteria: both have an external actor that triggers them (a photographer and the management board, respectively), both result in the provision of new meaningful information (the application and the resolution, respectively), and both are compositions of synchronous activities. Considering them to be only one communicative event would result in violating the trigger criteria (each has a different primary actor) and the reaction criteria (they *are* asynchronous: PHO 1 can occur at any moment during office hours, PHO 3 occurs Monday mornings).

Each *communicative event* is represented as a rounded rectangle and is given an identifier and a descriptive name. The *identifier* serves for traceability purposes and it is usually a code composed of a mnemonic (related to the system to which the event is ascribed) and a number (e.g. PHO 3). With regard to the *name*, we recommend to consistently use either an external nomination (primary actor + action + object + qualifier; e.g. "Photographer submits an application") or an internal nomination (support actor + action + object + qualifier; e.g. "Clerk receives a photographer application"). For instance, in the illustration case we have opted for an external nomination. For each event, involved actors are identified. Communication Analysis distinguishes several roles (see theoretical basis in [31]):

- The *primary actor* triggers the communicative event by establishing contact with the OS and provides the conveyed input information. Therefore, primary actors are modelled as senders of ingoing communicative interactions. For instance, the management board is the primary actor of event PHO 3.
- The *support actor* is in charge of physically interacting with the IS interface in order to encode and edit input messages. Support actors are specified at the bottom of the event rounded rectangle. Sometimes the primary actor and the support actor are different persons (e.g. photographer and clerk, respectively, in event PHO 1). Other times both roles are played by the same person (e.g. the salesman in event PHO 2).

– *Receiver actors* are those who need to be informed of the occurrence on an event. In order to truly understand the meaning of messages in organisations, it is necessary to analyse these actors. They are modelled as receivers of outgoing communicative interactions (e.g. in PHO 3 the photographer is informed of the resolution).
– *Reaction processors* are those in charge of performing the IS reaction to the message. This role is not depicted in the CED.

The messages associated to communicative events are conveyed via *ingoing communicative interactions* and *outgoing communicative interactions*. In the CED,

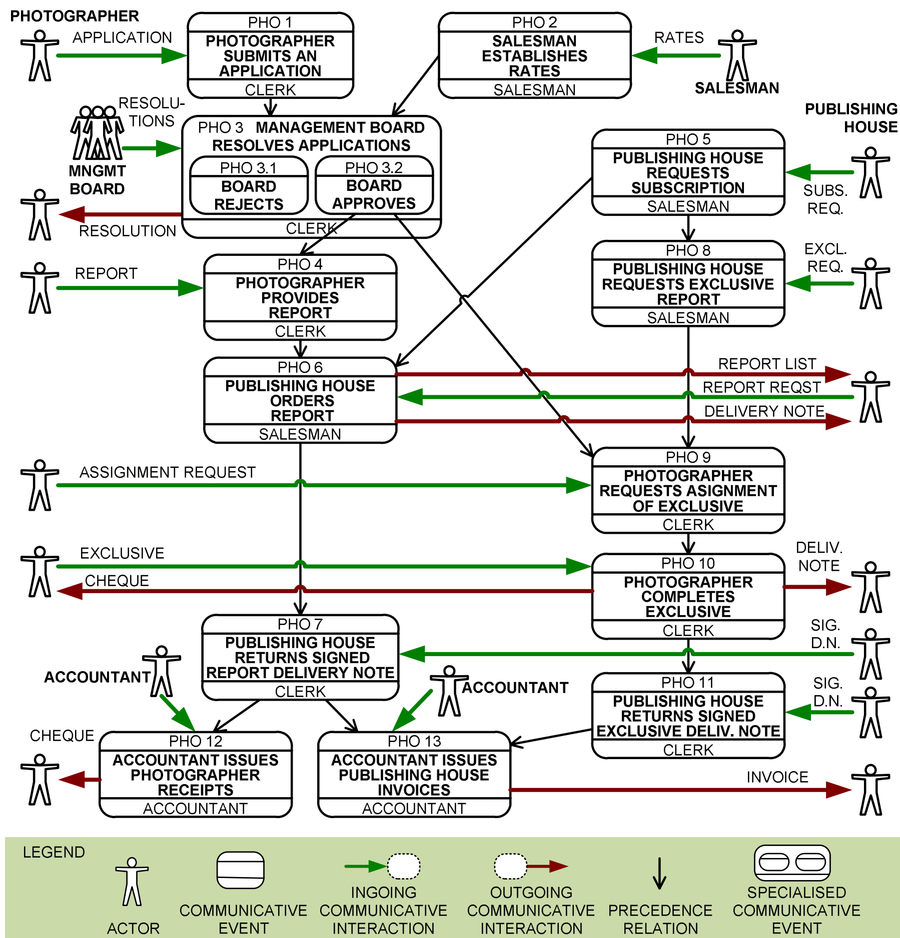


Fig. 2. Communicative Event Diagram of the photography agency[3]

[3] Note that the labels of Photographer actors (to the right) and Publishing house actors (to the left) are omitted for reasons of space. Also, some communicative interaction labels have been abbreviated (e.g. SIG.D.N. stands for SIGNED DELIVERY NOTE).

messages are given a name (by labelling communicative interactions)[4]. Communicative interactions are modelled as arrows placed in the horizontal axis. The vertical axis is reserved for *precedence relations* among communicative events, which are also modelled as arrows[5]. E.g. PHO 3 requires the previous occurrence of PHO 1 and PHO 2.

Communicative events are specialised whenever each specialised variant leads to a different temporal path (i.e. distinct precedence relations). It must be avoided specialising an event as a result of different communication channels, since the message remains the same (e.g. a publishing house can order a report in person or by telephone).

This requirements level also provides a static perspective of business processes, by means of business objects. We refer as *business objects* to the conceptions of those entities of the Subject System in which the OS is interested. Frequently, stakeholders describe business objects as complex aggregates of properties. Business objects are identified and described in a **Business Object Glossary**. Also, users are asked to hand out business forms to the analysts, who catalogue them for later form analysis. For instance, the photography agency manages the following business objects: photographer records, publishing house records and reports[6]. Static analysis also implies eliciting business indicators that are associated to subsystems or processes. For instance, the photography agency is interested in business indicators related to its three subsystems:

- Customer Service Department requires payments and takings indicators that allow them monitoring debts (e.g. publishing house indebtedness).
- Production Department requires productivity and profitability indicators (e.g. delivery performance to customer, photographer productivity).
- Customer Service Department requires client-related indicators that allow them to monitor customer loyalty (e.g. consumption rates).

4.3 L3. Communicative Interaction Level

Communicative events that appear in the CED need to be described in detail. Requirements associated to an event are structured by means of an **Event Specification Template**. The template is composed by a header and three categories of requirements: contact, communicational content and reaction requirements. These categories are related to phatic, referential and connative communication functions, respectively.

The *header* contains general information about the communicative event; that is, the event identifier, its name, a narrative description and, optionally, an explanatory diagram. The event identifier and name come from the CED; event identification needs to be kept consistent throughout the entire analysis and design specification in order to enhance requirements traceability. Since requirements specifications is meant, first of all, to facilitate problem understanding, a narrative description of the event is strongly advised. Also, whenever the event is complex, an explanatory diagram illustrating its associated flow of tasks shall be included.

---

[4] Message structure is specified in detail in a later activity (see Section 4.3).
[5] Complex business processes may require other operators. Start and end symbols can also be used. Besides, loops appear in many business processes.
[6] The description of business objects is omitted for the sake of brevity.

Table 2. Primitives and grammar of Communication Structures modelling technique

<table>
<tr><th>CSs primitives</th><th>EBNF grammar for Communication Structures[7]</th></tr>
<tr><td>Aggregation
A = < a + b + c >
A is composed of fields a and b and c.</td><td rowspan="4"><code>communication structure
= structure name, '=', initial substructure;
initial substructure
= aggregation substructure
| iteration substructure;
aggregation substructure
= '<', substructure list, '>';
iteration substructure
= '{', substructure list, '}';
specialisation structure
= '[', substructure list,
{ '|', substructure list },']';
substructure list
= substructure, { '+', substructure };
complex substructure
= aggregation substructure
| iteration substructure
| specialisation structure;
substructure
= substructure name, '=', complex substructure
| identifier field
| field;</code></td></tr>
<tr><td>Alternative
A = [a | b | c]
A is either composed of field a or b or c, (only one of them).</td></tr>
<tr><td>Iteration
A = { B }
A is composed of several substructures of type B.</td></tr>
<tr><td>Identification
a(id)
Field a identifies an object that is already known by the IS.</td></tr>
</table>

Contact requirements are related to the conditions that are necessary in order to establish communication. For instance[8], the primary actor, possible communication channels (e.g. fax, email, in person), availability and temporal restrictions (e.g. office hours for order reception), authentication requirements (e.g. in Spain, bureaucratic proceedings often require showing an identity card).

Communicational content requirements specify the message conveyed in an event and related restrictions (e.g. reliability: certifying that a diploma provided by a student is not fraudulent). With regard to the message, both metalinguistic aspects (e.g. message field structure, optionality of fields) and linguistic aspects (e.g. field domains, example values) need to be specified. Communication Analysis proposes a message modelling technique. **Communication Structure** (CS) is a modelling technique that is based in structured text and allows specifying the message associated to a communicative event. The structure of message fields lies vertically and many other details of the fields can be arranged horizontally; e.g. the information acquisition operation, the field domain, the link with the business object, an example value provided by users. A communicative event can not be fully understood until its CS is defined in detail. Specifying with precision an event CS forces and helps analysts and users to appropriately mark the event boundary and meaning. Table 2 shows the Communication Structures grammar. On left-hand side column, the primitives are informally

[7] This table summarises the main syntactical rules of the grammar. The elements `structure name`, `substructure name`, `identifier field` and `field` can be considered character strings.

[8] For reasons of space, not all kinds of requirements in each category are included.

explained; on the right-hand side column, an EBNF grammar [24] that allows expressing CSs is presented.

Reaction requirements describe how the IS reacts to the communicative event occurrence. Typically, the IS stores new knowledge, extracts all the necessary conclusions that can be inferred from new knowledge, and makes new knowledge and conclusions available to the corresponding actors. Therefore, this category of requirements includes business objects being registered and outgoing communicative interactions being generated by the event, among other requirements.

A simplified Event Specification Template of event PHO 3 is shown next.

PHO 3 Management board resolves applications

Goals: The IS aims to obtain a response to outstanding photographer applications.
Description: Monday mornings, the management board holds a meeting. A member of each department is present. A Production Department clerk has prepared a list of outstanding (pending) photographer applications and a résumé of each applicant. Management board proceeds to evaluate and resolve each application. Depending on the documentation, a photographer is either accepted or rejected. Accepted photographers are classified into a quality level (this level will determine their rates). After the meeting, the list of resolved applications is returned to Production Department.
Explanatory diagram: (Not included)

Contact requirements

Primary actor: Management board. **Communication channel**: In person.
Temporal restrictions: This communicative event occurs Monday mornings.
Frequency: Of the 10-20 monthly applications, around 5 are accepted.

Communicational content requirements

Support actor: Production Department clerk
Communication Structure: (See some comments below)

FIELD	OP	DOMAIN	BUSINESS OBJ.	EXAMPLE VALUE
RESOLUTIONS =			PHOTOGRAPHER	
{ Application ()=			(ID card #)=	
< ID card # +	i	text	<	19.345.631-Q
Name +	d	text		Sergio Pastor González
Address +	d	text		Camino de Vera s/n
Postcode +	d	text		46022
City +	d	text		Valencia
Phone # +	d	text		9638700000 ext 83534
Equipment +	d	text		Canon A1 w. telemacro
Experience +	d	text		Worked for Mangum Ph
Portfolio +	d	document		N/A (sample of work)
Resol. date +	i	date	resol date +	November 21, 2008
Decision +	i	[acc|rej]	decision +	acc
[Accepted =		Decision=acc		
< Level >]	i	Rate<level>	level	1 (highest quality level)
>			>	
}				

LEGEND

CSs Primitives

<+> aggregation
{ } iteration
[|] alternative
( ) selection

Information aquisition operations

d derivation
i input

Management board resolves each application (see the iteration of applications). Note that, for each application (identified by the ID card #), the only fields that constitute new information are the decision on whether to accept or reject the photographer (Decision field) and, in case of acceptance (message alternative and associated condition Decision=acc), the assigned quality level (Level field). The

<table>
<tr><td colspan="2">rest of the fields are derived from the IS memory (these data is introduced by a previous event; namely, PHO 1 Photographer submits an application). The business object column links dynamic perspective (communicative interaction description) with the static perspective (Business Object Glossary). Note that only new facts are stored in the IS memory. Example values enhance user-analyst communication.</td></tr>
<tr><td colspan="2">Reaction requirements</td></tr>
<tr><td>Business object: If the application is accepted, the photographer becomes part of the agency. The clerk creates a photographer record that includes photographer's personal and contact details (See scanned form at the right-hand side).

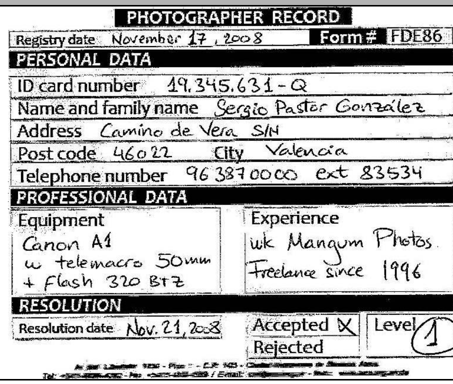
Outgoing communicative interaction: After this communicative event, a letter informing of the resolution is sent to the photographer[9].</td><td>
PHOTOGRAPHER RECORD
Registry date November 17, 2008 Form # FDE86
PERSONAL DATA
ID card number 19.345.631-Q
Name and family name Sergio Pastor González
Address Camino de Vera S/N
Post code 46022 City Valencia
Telephone number 96 387 0000 ext 83534
PROFESSIONAL DATA
Equipment: Canon A1 w telemacro 50mm + flash 320 BTZ
Experience: wk Mangum Photos. Freelance since 1996
RESOLUTION
Resolution date Nov. 21, 2008 Accepted X Rejected Level 1</td></tr>
</table>

Fig. 3. Event Specification Template of communicative event PHO 3

5 Related Works

There exist distinct orientations with regard to requirements elicitation for IS development. *Goal-oriented approaches* intend to identify stakeholders' necessities, modelling them as goals, where a "a goal is an objective the system under consideration should achieve" [39]. E.g. Map [35], i* [43] and KAOS [9]. Among *agent-oriented approaches*, which design the system as a set of autonomous and automatable agents, Tropos includes a goal-oriented requirements stage [6]. *Usage-based approaches* describe the interaction between the user and the software system under development. E.g. Use Cases [30] and Info Cases (an extension of the former) [18]. *Value-oriented approaches* identify and model value object exchanges [41]. E.g. e3-value [23]. Aspect-oriented approaches apply the separation of concerns principle [14] to RE. E.g. Early Aspects [33] and Theme/DOC [3]. Some organisational modelling approaches propose modelling and integrating multiple views of the system [5, 36, 11]. There also exist *communicational approaches*. In this field, a widely extended orientation is the Language Action Paradigm (LAP) [17, 42], which is mainly based on the work of Austin [1] and Searle's speech act classification [37]. Communicative Action Paradigm (CAP) is an evolution of LAP that extends the paradigm to non-verbal communication [13]. Several approaches stem from LAP, such as Action Workflow [28], SAMPO [15], Business Action Theory [19], DEMO [12] and Cronholm and Goldkunhl's Communication Analysis [8]. SANP [7] adopts a similar approach, but it is based on Ballmer and Brennenstuhl's speech act classification [2]. *Semiotic approaches* to organisational modelling have also been proposed [38].

[9] This communicative interaction is a printout and it is not described in detail for the sake of brevity.

This paper presents a communicational approach. Communication Analysis does not consider goal modelling, but the industrial projects in which the authors have been involved (which have contributed to consolidate the method) did not require it. Likewise, value network modelling has not been considered, but the method is usually put into practice in existing organisations with well established businesses, not with the intention to support an “innovative e-commerce idea” [23]. In any case, IS development is better tackled with a contingent approach, so we are open to integrating these or other perspectives with Communication Analysis (see Section 6).

Some features give advantage to our proposal over other methods. The actor roles argued in Section 4.2 distinguish Communication Analysis from other approaches. Many business process modelling techniques use support actors and/or reaction processors as criteria for organising processes in swimlanes but primary actors are disregarded (e.g. [11]). However, primary actors are central to our approach[10]. Furthermore, most RE approaches do not specify communicational content of interactions (or message specification is mixed with system usage description, as in Use Cases). Info Cases is an exception, since this technique proposes a structured text specification of messages. However, Communication Structures have greater expressiveness (e.g. alternative, iteration, information acquisition operation)[11].

With regard to communicational approaches, we share with them the communicational perspective and many foundations borrowed from Communication Theory. However, Communication Analysis does not necessarily preconceive a specific speech acts classification nor assumes conversational patterns, as LAP-based approaches do. An analyst following our approach does not impose patterns on the organisation, but confines to discovering the communication needs of the organisational stakeholders, shedding light on their work practice and identifying possible improvements. Our proposal coincides with the one by Cronholm and Goldkuhl in the communicational perspective. Also, both proposals consider organisational documents (e.g. business forms) invaluable sources of information. However, Cronholm and Goldkuhl choose existing documents as a starting point in RE process and their modelling notation (the Document Activity Diagram) is document-centred; that is, communicative interactions are subordinate to documents. We choose communicative interactions as a starting point and the modelling notation that we propose in this paper (the Communicative Event Diagram) is communicative interaction-centred. We argue that communicative events represent pure work practice and that it is possible to discover and describe them independently of their associated documents. Documents are a specific technological support[12] (solution space) for a communicative event (problem space); that is, documents are the result of a previous IS implementation.

With regard to the conceptual framework for understanding business processes by Melão and Pidd [29], our proposal combines two of the four perspectives; namely the constructivist and mechanistic perspectives.

---

[10] Organisations can replace many support agents with computer interfaces (e.g. clerks vs. web-based forms) and IS reaction processors are typically automated. Primary actors, however, are irreplaceable because they are the ultimate source of information.

[11] Making an in-depth comparison of our proposal with regard to other methods (e.g. feature comparison, performance evaluations) can not be dealt with in one single paper; we are currently working on empirical validation and results will be available as part of future works.

[12] We do not necessarily refer to computer technology; paper is an ancient form of technology.

- In order to discover business processes, a constructivist stance is adopted: business processes are considered a social construct that is agreed among stakeholders, and the requirements engineer acts as facilitator in this agreement.
- In order to describe business processes, a mechanistic stance is adopted: the use of models that are based on actors, events and messages allows creating a requirements specification that serves as a starting point for later software design.

Discovering requirements allows their description and, conversely, IS description provides feedback to discovery by allowing new interactions with stakeholders (e.g. to formulate new questions). Both perspectives are intertwined. The combination of hard (mechanistic) and soft (constructivist) approaches is not new to the ISs scene [4], but Communication Analysis contributes a requirements structure and communication-oriented modelling techniques that do not appear in previous proposals.

One last remark. The proposed business process modelling technique consists of a set of interrelated concepts, criteria plus other methodological guidance, and a notation. We believe that, in general, it is concepts and criteria that matter the most (not notations). We also acknowledge that some practitioners would be more comfortable using other notations for business process modelling (e.g. BPMN, Activity Diagrams, Use Cases). There is no problem with that, as long as the notation is adapted to support the communicational perspective. In fact, this has been done before.

6 Conclusions and Future Work

To sum up, Communication Analysis is an Information Systems (ISs) development method that proposes a flow of activities and a requirements structure. It is founded on Systems Theory and Communication Theory, among other scientific fields. An overview of Communication Analysis can be found elsewhere [20]. This paper focuses on the requirements elicitation stage and describes in detail several communication-based modelling techniques. The Communicative Event Diagram specifies business processes from a communicational point of view. In order to guide the analyst in identifying and determining the proper granularity of communicative events, unity criteria are proposed. Each communicative event is later specified by means of a template. Messages associated to communicative events are specified by means of Communication Structures, a notation based on structured text. The approach is exemplified using an illustration case (a photography agency).

Communication Analysis is currently being applied to big projects in industrial environments; e.g. the integration of Anecoop S.Coop (a Spanish major distributor of fruit and vegetables) with its associated cooperatives (>100). We plan to describe our industrial experience by means case study reports. Laboratory experiments have been carried out to test the benefits of the unity criteria; resulting models correction and data analysis is now being undertaken. Future work also involves developing a CASE tool that supports the method, researching how other perspectives (e.g. goal or value orientation) may extend our approach under certain project circumstances, and the integration of Communication Analysis and the OO-Method, an MDA-based method with software generation capabilities.

References

1. Austin, J.L.: How to do things with words. Oxford University Press, Oxford (1962)
2. Ballmer, T.T., Brennenstuhl, W.: Speech act classification: A study of the lexical analysis of English speech activity verbs. Springer, Berlin (1981)
3. Baniassad, E., Clarke, S.: Theme: an approach for aspect-oriented analysis and design. In: 26th International Conference on Software Engineering (ICSE 2004), pp. 158–167. IEEE Computer Society Press, Los Alamitos (2004)
4. Brown, J., Cooper, C., Pidd, M.: A taxing problem: the complementary use of hard and soft OR in the public sector. Eur. J. Oper. Res. 172(2), 666–679 (2006)
5. Bubenko, J.A., Brash, D., Stirna, J.: EKD User Guide. Dept. of Computer and Systems Science tech. report, Stockholm University (1998)
6. Castro, J., Kolp, M., Mylopoulos, J.: Towards requirements-driven information systems engineering: the Tropos project. Information Systems 27, 365–389 (2002)
7. Chang, M.K., Woo, C.C.: A speech-act-based negotiation protocol: design, implementation, and test use. ACM Trans. Inf. Syst. 12(4), 360–382 (1994)
8. Cronholm, S., Goldkunhl, G.: Communication Analysis as perspective and method for requirements engineering. In: Mate, J.L., Silva, A. (eds.) Requirements engineering for socio-technical systems, pp. 340–358. Idea Group Inc. (2004)
9. Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-directed requirements acquisition. Sci. Comput. Program. 20(1-2), 3–50 (1993)
10. Davis, A.M.: Software Requirements: Analysis and Specification. Prentice-Hall, Englewood Cliffs (1990)
11. de la Vara, J.L., Sánchez, J., Pastor, O.: Business process modelling and purpose analysis for requirements analysis of information systems. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074. Springer, Heidelberg (2008)
12. Dietz, J.L.G.: Understanding and modelling business processes with DEMO. In: Akoka, J., Bouzeghoub, M., Comyn-Wattiau, I., Métais, E. (eds.) ER 1999. LNCS, vol. 1728, pp. 188–202. Springer, Heidelberg (1999)
13. Dietz, J.L.G., Goldkuhl, G., Lind, M., van Reijswoud, V.E.: The Communicative Action Paradigm for business modelling - a research agenda. In: 3rd International Workshop on the Language Action Perspective on Communication Modelling (LAP 1998). Jönköping International Business School (1998)
14. Dijkstra, E.W.: A discipline of programming. Prentice-Hall, Englewood Cliffs (1976)
15. Esa, A., Lehtinen, E., Lyytinen, K.: A speech-act-based office modeling approach. ACM Trans. Inf. Syst. 6(2), 126–152 (1988)
16. Falkenberg, E., Hesse, W., Lindgreeen, P., Nilsson, B., Oei, J.L.H., Rolland, C., Stamper, R., Van Assche, F., Verrijn-Stuart, A., Voss, K.: FRISCO. A Framework of Information Systems Concepts. IFIP WG 8.1 Task Group Report (1998)
17. Flores, F., Ludlow, J.: Doing and speaking in the office. In: Fick, G., Sprague, R.H. (eds.) Decision Support Systems: issues and challenges, NY, USA, pp. 95–118. Pergamon Press, Oxford (1980)
18. Fortuna, M., Werner, C., Borges, M.: Info Cases: integrating use cases and domain models. In: 16th International Requirements Engineering Conference (RE 2008), Barcelona, Spain, pp. 81–84. IEEE, Los Alamitos (2008)
19. Goldkuhl, G.: Generic business frameworks and action modelling. In: International workshop on the Language Action Perspective on Communication Modelling (LAP 1996). Tilburg, The Netherlands (1996)

20. González, A., España, S., Pastor, O.: Towards a communicational perspective for enterprise Information Systems modelling. In: IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM 2008), Stockholm, Sweden. LNBIP, vol. 15, pp. 63–77. Springer, Heidelberg (2008)
21. González, A., España, S., Pastor, O.: Unity criteria for Business Process Modelling: a theoretical argumentation for a Software Engineering recurrent problem. In: 3rd Intl. Conf. on Research Challenges in Information Science (RCIS 2009), Fes, Morocco. IEEE, Los Alamitos (2009)
22. González, A.: Algunas consideraciones sobre el uso de la abstracción en el análisis de los sistemas de información de gestión. Ph.D thesis (in Spanish). Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia (2004)
23. Gordijn, J., Wieringa, R.J.: A value-oriented approach to e-business process design. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681, pp. 390–403. Springer, Heidelberg (2003)
24. ISO/IEC 14977: Information technology - Syntactic metalanguage - Extended BNF (1996)
25. Jakobson, R.: The speech event and the functions of language. In: Monville-Burston, M., Waugh, L.R. (eds.) On language, pp. 69–79. Harvard University Press, Cambridge (1990)
26. Langefors, B.: Theoretical analysis of Information Systems, 4th edn. Studentlitteratur, Lund (1977)
27. Lockemann, P.C., Mayr, H.C.: Information System Design: Techniques and Software Support. In: Kugler, H.-J. (ed.) IFIP 1986, North-Holland, Amsterdam (1986)
28. Medina-Mora, R., Winograd, T., Flores, R., Flores, F.: The action workflow approach to workflow management technology. In: ACM conference on Computer-Supported Cooperative Work (CSCW 1992), Toronto, Ontario, Canada, pp. 281–288. ACM, New York (1992)
29. Melão, N., Pidd, M.: A conceptual framework for understanding business processes and business process modelling. Inform. Syst. J. 10(2), 105–129 (2000)
30. OMG: Unified Modeling Language: Superstructure version 2.0, `http://www.omg.org/docs/formal/05-07-04.pdf` (accessed 11, 2008) (2005)
31. Pastor, O., González, A., España, S.: 31. Pastor, O., González, A., España, S.: Conceptual alignment of software production methods. In: Krogstie, J., Opdahl, A.L., Brinkkemper, S. (eds.) Conceptual modelling in Information Systems engineering, pp. 209–228. Springer, Berlin (2007)
32. Pastor, O., Molina, J.C.: Model-Driven Architecture in practice: A Software Production Environment Based on Conceptual Modeling. Springer, New York (2007)
33. Rashid, A., Sawyer, P., Moreira, A., Araújo, J.: Early Aspects: a model for aspect-oriented requirements engineering. In: 10th Anniversary IEEE Joint International Conference on Requirements Engineering, pp. 199–202. IEEE Computer Society, Los Alamitos (2002)
34. Rittel, H., Webber, M.: Dilemmas in a general theory of planning. Policy Sciences 4, 155–169 (1973)
35. Rolland, C.: Capturing system intentionality with Maps. In: Krogstie, J., Opdahl, A.L., Brinkkemper, S. (eds.) Conceptual modelling in Information Systems engineering, pp. 141–158. Springer, Heidelberg (2007)
36. Scheer, A.-W.: ARIS - Business Process Modeling, 3rd edn. Springer, New York (2000)
37. Searle, J.R., Vanderveken, D.: Foundations of illocutionary logic. Cambridge University Press, Cambridge (1985)
38. Stamper, R.K.: Organizational semiotics. In: Stowell, F., Mingers, J. (eds.) Information Systems: an emerging discipline, London, pp. 267–283. McGraw Hill, New York (1997)

39. van Lamsweerde, A.: Goal-oriented Requirements Engineering: a guided tour. In: 5th IEEE International Symposium on Requirements Engineering (RE 2001), Toronto, Canada, pp. 249–262. IEEE Computer Society Press, Los Alamitos (2001)
40. Wand, Y., Weber, R.: On the deep structure of information systems. Inf. Syst. J. 5, 203–223 (1995)
41. Weigand, H., Johannesson, P., Andersson, B., Bergholtz, M., Edirisuriya, A., Ilayperuma, T.: On the notion of value object. In: Dubois, E., Pohl, K. (eds.) CAiSE 2006. LNCS, vol. 4001, pp. 321–335. Springer, Heidelberg (2006)
42. Winograd, T., Flores, F.: Understanding computers and cognition: A new foundation for design. Addison-Wesley, Reading (1987)
43. Yu, E., Mylopoulos, J.: From E-R to "A-R" - Modelling strategic actor relationships for business process reengineering. In: Loucopoulos, P. (ed.) ER 1994. LNCS, vol. 881, pp. 548–565. Springer, Heidelberg (1994)