Decision Support Systems 51 (2011) 350–360

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Using service responsibility tables to supplement UML in analyzing e-service systems

Xin Tan [a,1], Steven Alter [b,2], Keng Siau [c,*]

[a] Fairleigh Dickinson University, 1000 River Road, H-DH2-06, Teaneck, NJ 07666, United States
[b] University of San Francisco, 2130 Fulton Street, MH-314, San Francisco, CA 94117-1045, United States
[c] University of Nebraska-Lincoln, 209 CBA, Department of Management, Lincoln, NE 68588-0491, United States

ARTICLE INFO

Article history:
Received 10 May 2009
Received in revised form 16 January 2011
Accepted 16 January 2011
Available online 21 January 2011

Keywords:
Systems analysis and design
Work system method
Service responsibility table
Unified modeling language

ABSTRACT

This paper proposes using Service Responsibility Tables (SRTs) as a tool in analyzing e-service systems. First it discusses difficulties and deficiencies of using formal modeling languages such as UML in analyzing e-service systems. It proposes using SRTs as an informal language and lightweight analytical tool to be used by business professionals in analyzing e-service systems. SRTs are based on a service value chain framework but do not rely on abstract concepts and constructs, and therefore can be used by business professionals to supplement UML. We suggest a set of heuristics for transforming SRTs into two key UML diagrams, thereby illustrating how SRTs can be used as a bridge from relatively informal modeling by business professionals to formal modeling techniques for systems design and implementation by IT professionals.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

The advent of the Internet and other modern information and communication technologies (ICTs) has led businesses worldwide to embrace e-commerce and e-service. E-commerce has been widely accepted as a viable business model for buying and selling products/services over the Web, leading to a new paradigm known as e-service, or electronic service, defined as the provision of service over electronic networks [42]. Common examples of such e-service include online package tracking, email notification of order status, and more recently mobile banking. The advantages of providing customers with e-service include reducing operating expenses, allowing for personalization, and improving customer satisfaction [8,40]. Through e-service, enhanced service experience and higher levels of customer satisfaction tend to increase revenues and profitability [42]. For customers, the benefits of e-service include addressing new business needs, saving money and time in traveling, and avoiding awkward interpersonal encounters [32].

Whereas e-service systems may offer many benefits to businesses and their customers, developing e-service systems is challenging [56]. E-service is predominately self-service [40]. In other words, customer use of e-service systems implies coproduction of service, frequently requiring customers to engage in new behaviors [33]. There is a general agreement on the importance of user involvement during systems analysis and design [10,55].

The self-service nature of e-service poses several difficulties in effectively involving users' participation in systems analysis and design stages. First, ordinary customers, i.e. the intended users of e-service systems, often have difficulty identifying and describing the capabilities and features they want [33]. Even if the system totally reflects what they requested, it often omits important capabilities that the users failed to request [34].

Second, formal requirements modeling methods, such as Unified Modeling Language (UML), are frequently used to create requirements representations that need users' review and approval [22]. Such formal notations are difficult to understand for people with little or no technology background [25]. Users have difficulty verifying the accuracy and completeness of requirements models that are presented in formalisms that are unfamiliar to them at best, and sometimes seem impenetrable to typical business professionals.

This paper proposes Service Responsibility Tables (SRTs) as tools that business professionals and service customers can use in the analysis stage of e-service system development. The idea of SRTs came from service-related extensions of work system research [2,4], which has focused for over a decade on developing lightweight systems analysis tools for business professionals. The form of SRTs is based on the structure of the service value chain framework [4,6,7]. Tan, Alter, and Siau [57] propose that the use of SRTs might alleviate difficulties in analyzing e-service systems by helping users and analysts summarize and discuss important elements and processes involved in service provision. In comparison with formal modeling methods, SRTs are easier to use by business professionals who typically have little or no knowledge of the heavyweight analytical tools. SRTs can

* Corresponding author. Tel.: +1 402 472 3078; fax: +1 402 472 5855.
E-mail addresses: xtan@fdu.edu (X. Tan), alter@usfca.edu (S. Alter), ksiau@unl.edu (K. Siau).
[1] Tel.: +1 201 692 7283; fax: +1 201 692 7219.
[2] Tel.: +1 415 422 6383; fax: +1 415 422 2502.

0167-9236/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2011.01.001

help them devote their cognitive effort to eliciting and identifying requirements in e-service systems, instead of making sense of the syntax and grammar of formal modeling languages.

We are not proposing that SRTs should replace UML, which is the *de facto* modeling standard of the software development community [21]. Instead, we propose to use SRTs as a lightweight analytical tool for business professionals. The resulting SRTs can be transformed to rigorous UML diagrams by analysts who are able to fill in missing details needed to produce syntactically and substantively correct UML diagrams. We suggest using a set of heuristics to facilitate the transformation from SRTs to UML diagrams.

The rest of the paper is organized as follows. The first section introduces common cognitive problems that can affect requirements modeling, and explains why some of these problems are heightened in e-service systems analysis. The objective is to clarify the difficulties of applying formal modeling methods in the analysis stage. The second section provides background on a flexible class of analysis tools called SRTs, and explains how they can be used as lightweight analysis tools for business professionals and analysts. The third section summarizes initial heuristics for transforming SRTs into UML diagrams for documenting requirements. The usefulness of the heuristics is illustrated through a case. The conclusion section discusses the implications of the present study for research and practice, and highlights the directions for future research.

2. Related literature

Eliciting and documenting users' requirements for the system being built are crucial activities in system development projects. Despite the general agreement on the importance of user involvement during systems analysis and design, the level and quality of user involvement are often inadequate [30,31]. This section reviews related literature to summarize common problems in eliciting and documenting users' requirements. It also surveys the literature related to e-service to reveal the additional problems in using formal modeling methods in e-service systems analysis. It concludes with the features of a novice-friendly analytical tool to supplement UML for e-service systems analysis.

2.1. Common problems in requirements determination

One of the key objectives of systems analysis and design is to determine users' information requirements. Being able to understand users' information requirements is vital in any software project and a key factor in any successful software implementation [11–13,17]. Information requirements determination is a set of activities performed by systems analysts to assess the functionality required in a proposed system. According to Pohl [38], requirements determination involves four tasks that are often performed iteratively in practice:

i. Requirements specification: to understand the organizational situation that the system under consideration aims to improve and describe the needs and constraints of the system under development.
ii. Requirements negotiation: to establish an agreement on the requirements of the system among the various stakeholders involved in the process.
iii. Requirements representation: to develop a mapping of real-world needs onto a requirements model.
iv. Requirements validation: to ensure that the derived specification corresponds to the original stakeholder needs and conforms to the internal and/or external constraints set by the enterprise and its environment.

Requirements determination for information systems is a highly communicative, iterative, and creative activity. Holtzblatt and Beyer [28] claim that the success of information requirements determination depends on how well people communicate and understand each other. Davis [16] summarizes four sources of difficulties in information requirements determination: (1) the constraints on humans as information processors and problem solvers, (2) the variety and complexity of information requirements, (3) the complex patterns of interaction among users and analysts in defining requirements, and (4) the unwillingness of users to provide requirements. Valusek and Fryback [59] label these difficulties as communication obstacles "within" individual users, "among" users, and "between" users and analysts. More recently, Siau and Tan [48,49] elaborate further on the cognitive underpinnings of such communication problems in the context of conceptual modeling and requirements determination.

Frames of reference held by organization members are implicit guidelines that serve to organize and shape their interpretation of events and organizational phenomena, as well as to give meaning to it [62]. Inconsistent frames of reference among stakeholders may lead to communication problems in systems analysis and design. Cognitive inefficacy during systems analysis also may lead to inaccurate or incomplete requirements. In relation to systems development and use, Orlikowski and Gash [35] label such frames of reference as technological frames that are vital in understanding of the variety and complexity of user needs. For instance, analysts must manage the subjective nature of language to disseminate technology to users in a non-technical way. On the other hand, analysts often organize, classify, describe, and explain complex processes and concepts in a highly technical manner.

2.2. Nature of e-service systems

Businesses and non-profit organizations have started embracing e-service to deliver service to their customers through electronic networks, although there is a lack of agreement on the definition of e-service [40]. In this research, we define e-service as:

> *Acts or performances that are delivered through electronic devices and networks to help people complete tasks, solve problems, or conduct transactions.*

A review of e-service literature [40] identifies two inherent characteristics of e-service: information service and self-service. Information service refers to the assumption that information is the primary value exchanged between the two parties in an e-service relationship. Further, an e-service experience is a self-service experience [40], or customer coproduction of service [33]. Some researchers claim that "selling tangible goods online is itself an e-service that substitutes for physical retailing" [27].

These characteristics lead to challenges in developing and implementing e-service systems. First, the self-service nature of e-service has a high demand for customer engagement and learning [33,40]. Second, because e-service is typically conceptualized as information service, the dimensions of information quality, as well as non-technological features such as trust, reliability, and risk, should be considered as part of comprehensive requirements for e-service systems.

2.3. Why UML by itself is not sufficient for e-service systems analysis

With the dominance of object-oriented paradigm in information systems development, object-oriented modeling is expected to play an increasingly important role in requirements engineering. Consistent with this trend, UML has emerged as a *de facto* standard for object-oriented modeling language [21,22,29]. UML is a graphical modeling language for modeling system requirements, describing

دریافت فوری ← متن کامل مقاله

ISIArticles
مرجع مقالات تخصصی ایران

✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
✓ امکان دانلود نسخه ترجمه شده مقالات
✓ پذیرش سفارش ترجمه تخصصی
✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
✓ دانلود فوری مقاله پس از پرداخت آنلاین
✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات