Journal of Visual Languages and Computing 000 (2017) 1–15

Contents lists available at ScienceDirect

Journal of Visual Languages and Computing

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

Weaving semiotic engineering in meta-design: A case study analysis

Daniela Fogli

Department of Information Engineering, University of Brescia, Via Branze 38, 25123 Brescia, Italy

ARTICLE INFO

Article history:
Received 9 February 2016
Revised 29 January 2017
Accepted 5 April 2017
Available online xxx

Keywords:
End-user development
Semiotic engineering
Meta-design
Domain expert
Case study

ABSTRACT

The work presented in this paper aims at investigating how semiotic engineering and meta-design could be combined to support the development of socio-technical environments that enable End-User Development (EUD). In particular, I investigate the relationships existing between domain experts, playing the role of end-user developers, and meta-designers, mediated by EUD environments, and between end users and end-user developers mediated by EUD products. To this end, three case studies are considered, which belong to the recent research experience in EUD of the author. The case studies are concerned with three different application domains, namely physical prototyping of social products, accumulation and sharing of territory knowledge for first aid intervention, development of e-government services for the citizens of a municipality. The analysis is carried out both on the design process, by adopting a meta-design perspective, and on the product developed in each project, according to a semiotic engineering perspective. The analysis allows to shed light on the human and technical actors involved in EUD and on their communication processes, in order to understand which kinds of interaction visual languages and social conditions should be defined to encourage a continuous user-system co-evolution. As a result, the paper finally proposes some operative indications for the design of systems enabling EUD, which capitalize on semiotic engineering and meta-design ideas.

1. Introduction

The ongoing evolution of information and communication technology offers users new tools and creates numerous expectations for software systems' potential. Computer users are evolving from passive consumers of data and computer tools into more active producers of information and software artifacts [32]. This evolution creates opportunities for new research but also sets out new challenges for designing and implementing interactive systems that support people's activities in various fields. Numerous hurdles and barriers rear up to make system design difficult [18,55].

The communication gap between designers and users has long been recognized as a major source of trouble [18,46,55,56], and today it became more critical than ever due to the widespread use of Web 2.0 applications, multi-modal interfaces, and end-user computing by people neither expert in software technologies nor interested in them. Users and designers generally possess distinct types of knowledge, and follow different approaches, processes, and reasoning strategies in modeling, performing, and documenting the tasks to be carried out in a given application domain. As a result, users tend to form a mental model of the interactive system that usually differs from the designers' mental model [50], thus compromising an effective user-system interaction.

To address this problem, user-centered and participatory design approaches have been proposed in the Human-Computer Interaction (HCI) literature. The former gather system requirements through consultation with users as experts of the application domain and by analyzing users' existing documentation or work practice; they also involve users in usability evaluation of the designed system [1,57,61,62]. Participatory design approaches call for the participation in the design process of representative users – *domain experts* in [21] – not only as experimental subjects but also as active members of the design team [6,8,9,66]. In participatory design, domain experts are directly involved in the creation of interaction scenarios [64] and/or static or interactive prototypes [8], and, as a consequence, in the selection of the visual representation for objects and operations characterizing the tasks to be supported.

On the other hand, de Souza explores the relationship between designers and users from a semiotic perspective, by modeling such a relationship as a computer-mediated communication [25].

In other terms, semiotic engineering views interactive systems as *meta-communication artifacts*, that is, artifacts that communicate a message about communication itself [25].

This communication does not only encompass the words, graphics, and icons exchanged in each step of the interaction, but also the behaviors that take place and the artifacts created as the result of the user-system 'cooperation'. Therefore, the semiotic engineering approach proposed in [25] goes beyond used-centered

E-mail address: daniela.fogli@unibs.it

http://dx.doi.org/10.1016/j.jvlc.2017.04.002
1045-926X/© 2017 Elsevier Ltd. All rights reserved.

and participatory design, by emphasizing the need for sustaining the so-called progressive semantization [69] that makes the user co-evolve with the system during usage [14,44]. Indeed, a further important hurdle that neither user-centered design nor participatory design is able to address effectively, is the arising, at the use time, of new requirements that ask for new or different system functionalities to support users' activities. This is usually regarded as a twofold evolution, namely an evolution of the users who ask for the evolution of the system to support new tasks, and the arising of new user's needs as the effect of system evolution. This overall phenomenon has been called *co-evolution of system and users* and denotes all those situations where a system and its users co-evolve in a continuously self-adapting mutual fit [23]. To take the co-evolution phenomenon into consideration, a recent research trend in HCI is creating interactive systems that support End-User Development (EUD) [54].

The application of a semiotic engineering approach to the design of interactive systems opens up new ways for dealing with the communication gap between users and designers, as well as for supporting end-user development [27], in order make the co-evolution of users and systems possible. The goal of EUD techniques is to make users able to develop their own software artifacts not only at the design time, but also during their usage. To achieve this goal, research scholars have proposed to adopt *meta-design,* that is, a conceptual framework aimed at creating the socio-technical conditions that allow the owners of problems (end users) to act as designers at use time [34,35]. In this sense, meta-design is rightly considered as a research area related to EUD [67]. Furthermore, meta-design proved to be a suitable approach to address all those situations where some domain experts play the role of mediators between software designers and pure end users, by creating themselves the software artifacts tailored to end users' culture, skills and background. This view on meta-design has been adopted, implicitly or explicitly, in a variety of domains: from medicine [20] to cultural heritage [4,19], from assistive technologies [17] to e-government [42], from neuro-rehabilitation [68] to robotics [5]. However, less investigation has been carried out so far on the relationship between the semiotic engineering approach to EUD and the meta-design approach to EUD. An initial proposal in this direction appears in [12], which combines the seeding and reseeding concepts of the Seeding, Evolutionary Growth and Reseeding (SER) process model for meta-design [36] with the categories of signs (static, dynamic and metalinguistic signs) proposed in semiotic engineering [25].

In order to better explore such relationship and gather evidences of how semiotic engineering may become effective in the meta-design of interactive systems enabling EUD, this paper analyses three different projects, where end users, as domain experts, participated either in *artifact creation* or *system tailoring*, without the need to learn any programming language or software engineering technique, but by simply capitalizing on notation, knowledge and habits of their own work practice. In other terms, attention is put on the users in the role of *end-user developers* [44], and EUD tools are regarded in this paper as powerful mediators between end-user developers and meta-designers, somewhat extending the semiotic framing of end-user development proposed in [27].

The three case studies that will be considered in this paper are derived from the research experience of the author in the last five years. They are concerned with three different application domains – physical prototyping, first aid and e-government – and regard as a consequence three different user communities, with their own organization, work practice, skills, and idiosyncrasies. The first and most recent case study is about supporting industrial design students in the creation of physical prototypes able to interact with social networks [45]. The second case is concerned with the design of a collaborative web mapping application to be used as a training tool and knowledge-sharing repository by the volunteers of an Italian non-profit association for public assistance and first aid [39]. In the e-government project, civil servants have been involved in the creation of web pages for supplying citizens with specific municipality services [40,42].

With the help of the above case studies and following Schön's idea of "reflection-on-action" [65], a qualitative research study is carried out to offer a reflection on EUD practice and socio-technical phenomena surrounding it from a more general and unified point of view with respect to the project descriptions provided in the author's previous works. The first and third case studies are concerned with the creation of new software artifacts on behalf of end-user developers, but are not based on a full-fledged programming language. The second case, instead, has mainly to do with system tailoring, intended as both customization and extension [58]. Therefore, the analysis of the three cases from a semiotic perspective led to identify different results, which are somewhat complementary to those discussed in [27], where EUD for customization and extension was the main focus. In particular, the relationships between semiotic engineering and meta-design are investigated through abductive reasoning [63], that is, through logical inference from observations to theory, in order to understand if these approaches to EUD may cross-fertilize. Far from providing a comprehensive theory, with these results the paper would like to contribute to the cross-disciplinary research on interactive computer systems advocated in [28].

The paper is organized as follows: Section 2 discusses background and related work about design approaches in HCI and their relationship with semiotic engineering; it also discusses literature work on meta-design and adds some more information about the motivations underlying the present work. Section 3 presents the research methodology adopted in this paper inspired to the systematic combining approach of Dubois and Gadde [30] grounded on abductive logic [63], and introduces the three case studies framing them in the taxonomic classification proposed in [15]. Sections 4–6 provide details about the considered case studies, as well as a discussion from meta-design and semiotic engineering perspectives. Section 7 summarizes the main findings gathered through the analysis of the case studies, thus providing a synthesis about a possible combination of meta-design and semiotic engineering, which could be useful for future EUD projects. Section 8 highlights the limitations and open issues of the research. Section 9 briefly summarizes the work and concludes the paper.

2. Background and related work

User-centered design and participatory design approaches have been proposed over the years to foster the development of interactive systems that are 'easy-to-use'. In particular, participatory design aims at involving domain experts not only in the task analysis, requirements specification and evaluation of prototypes, like user-centered design does, but also in the design of the conceptual model of the system; this provides domain experts with the possibility of having their voice heard whenever decisions about language, signs, animations, screen layout, and system transitions are taken by the design team.

However, the emerging need of systems able to co-evolve with their users during user-system interaction asks for systems that are also 'easy-to-develop-and-tailor' [23], namely that provide users with the tools for becoming developers at use time, and not only designers at design time.

These possibilities have been originally offered in commercial software by end-user programming (EUP) techniques, such as macro recording in word processors, creation of formulas in spreadsheets [60] or filter definition in e-mail clients. Nowa-