

Interaction and Context in Service–Oriented E–Collaboration Environments

Christoph Dorn

Vienna University of Technology, Austria

Schahram Dustdar

Vienna University of Technology, Austria

Giovanni Giuliani

HP Italiana SRL, Italy

Robert Gombotz

Vienna University of Technology, Austria

Ke Ning

National University of Ireland, Ireland

Sébastien Peray

European Microsoft Innovation Center, Germany

Stephan Reiff-Marganiec

University of Leicester, UK

Daniel Schall

Vienna University of Technology, Austria

Marcel Tilly

European Microsoft Innovation Center, Germany

INTRODUCTION

As it has been observed in the recent decade, collaborating teams become ever more unstable, less tightly coupled and more distributed and mobile. Workers participate in multiple teams that pursue different goals that need not be related in any way. This radical way in which the workplace is changing for the individual and the team requires highly adaptable groupware and intelligent support for the individual in order to minimize the time lost for management and coordination when switching between different teams, different workspaces, and different contexts. Thus, a service-oriented approach seems promising to provide individual, context-aware building blocks for adaptable groupware.

To achieve this goal, we start by analyzing patterns of human interaction. Together with a context

meta-model, such patterns enable effective selection, adaptation, and invocation of services.

BACKGROUND

Many context frameworks target specific groups such as mobile users (Tang et al., 2001; Bardram & Hansen, 2004) or small mobile groups (Pokraev et al., 2005) acting independently of others. More generic frameworks try to cover a wider area but hence lack explicit support for things like group interaction. Exemplary tools focusing mostly on context and hardly on the interaction between people on a broader scope are CASS (Fahy & Clarke, 2004), CoBra (Chen, Finin, & Joshi, 2003), CORTEX (Biegel & Cahill, 2004), Gaia (Roman et al., 2002), Hydrogen (Hofer, Schwinger, Pichler, Leonhartsberger, & Altmann, 2002), and

SOCAM (Gu, Pung, & Zhang, 2004). An overview of these and additional frameworks can be found in Baldauf et al. (2006).

On the other hand, groupware systems such as the file-orientated BSCW (Bentley, Horstmann, Sikkil, & Trevor, 1995), virtual office-like Groove (www.groove.net), or the process-aware ad hoc collaboration tool Caramba (Dustdar, 2004) focus on collaborative functions while taking context only to some extent into account. Moreover, highly synchronous collaboration tools such as coediting neglect long-term team interactions.

Most teamwork tools are tightly integrated applications, whereas a service-oriented collaboration architecture reflects the notion of scoped functions wrapped as services that are provided, described, published, found, invoked and aggregated. Hence, a SOA oriented approach to context awareness (Gu et al., 2004) and collaboration (Jørstad, Dustdar, & van Do, 2005) seems to be very promising, but the notion of context as proposed by Dey and Abowd (1999), needs to be extended beyond involved services (Dorn & Dustdar, 2006) to explicitly include teams as a first order entity.

The exhaustive review of current literature by Powell, Picolli, and Ives (2004) reveals that research efforts have merely focused on distributed teams as a whole without analyzing the internal interaction. Hence, three interaction patterns provide the basis to improve our understanding of e-collaboration requirements.

INTERACTION PATTERNS

The term *interaction pattern* refers to a common, reoccurring interaction scenario between actors. The term *relation* refers to a tie or link between two actors within a pattern. We take three initial interaction pat-

terns that are well known in the domain of software engineering (SE) and apply them to the domain of human collaboration.

PROXY PATTERN

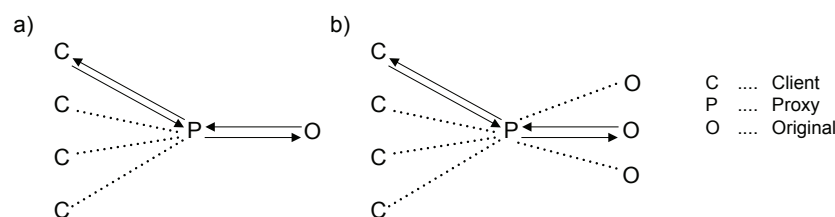
Originally, the Proxy pattern was introduced by Gamma, Helm, Johnson, and Vlissides (1994, p. 207) as a structural pattern in software design. The intention for using a proxy is “to provide a surrogate or placeholder for another object to control access to it.” Besides forwarding the clients’ requests and sending back the response, a proxy can do pre- or post-processing depending on its type. A real-life example of a proxy in human collaboration is a secretary. He or she receives e-mails, phone calls, messages, etc., which are actually intended for a different entity, the boss. The secretary pre-processes these client requests by, for example, filtering out unwanted requests—protection proxy—or even answering simple requests without having to involve the boss—cache proxy (Dustdar & Hoffmann, 2006).

A proxy pattern usually describes a 1:1 relationship between proxy and original as depicted in Figure 1a. However, there are two exceptions, remote proxies and firewall proxies, where a proxy is responsible for multiple originals (Figure 1b).

BROKER PATTERN

The Broker architectural pattern can be used to structure distributed software systems with decoupled components that interact by remote invocations. “A broker component is responsible for coordinating communication, such as forwarding requests, as well as

Figure 1. Proxy pattern



7 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/interaction-context-service-oriented-collaboration/12455

Related Content

Narrative Reasoning

(2012). *Approaches for Community Decision Making and Collective Reasoning: Knowledge Technology Support* (pp. 144-178).

www.irma-international.org/chapter/narrative-reasoning/67325

Shared Mental Model Development During Technology-Mediated Collaboration

Hayward P. Andres (2011). *International Journal of e-Collaboration* (pp. 14-30).

www.irma-international.org/article/shared-mental-model-development-during/55425

Improving Virtual Design Team Performance Through Use of a Collaborative Sketching Application

Brett Stone, John Salmon, Ammon Hepworth, Steven Gorrelland Michael Richey (2017). *International Journal of e-Collaboration* (pp. 1-22).

www.irma-international.org/article/improving-virtual-design-team-performance-through-use-of-a-collaborative-sketching-application/215449

e-Collaboration in Distributed Requirements Determination

Roberto Evaristo, Mary Beth Watson-Manheimand Jorge Audy (2005). *International Journal of e-Collaboration* (pp. 40-56).

www.irma-international.org/article/collaboration-distributed-requirements-determination/1931

An Awareness Framework for Divergent Knowledge Communities

Farhad Daneshgar, Gerome Canaland Alicia Diaz (2008). *Encyclopedia of E-Collaboration* (pp. 42-47).

www.irma-international.org/chapter/awareness-framework-divergent-knowledge-communities/12402