



Chapter V

Aspect-Oriented Software Components

Anders I. Mørch
InterMedia, University of Oslo, Norway

This chapter presents a novel approach for conceptualising and organising computer applications for the purpose of simplifying end-user tailorability and evolutionary application development. The chapter combines ideas from software systems, end-user programming, Gestalt psychology, and everyday tools. An application unit is the basic unit of tailorability. It is defined as a visual component extended with aspects. Aspects provide user-oriented entry points into an application unit at different levels of abstraction. Tools and techniques for locating, viewing and tailoring application unit aspects are presented, and the results of a user study are reported. A goal of writing this chapter has been to present a comprehensive view of some basic technology for user participation in evolutionary application development.

INTRODUCTION

End-user tailoring is about redesign and further development of existing computer applications without access to original developers (Henderson & Kyng,

1991). The basic unit of tailorability is an application unit (Mørch, 1995). This is a software component supplemented with aspects. An aspect is a way of seeing a component and adopted from Gestalt psychology (Köhler, 1947). The combination of software components and aspects is novel and explained in this chapter. The aspects provide alternative interfaces to an application and help to bridge the gap between using and programming applications. Application units share some of the characteristics of programmable components in commercial software applications such as VBA (Visual Basic for Applications; Lomax, 1998). When supplemented with aspects these components can support evolutionary development in ways that corresponds with the dynamics of evolutionary growth processes studied in other domains. This approach to adaptation and evolutionary application development is further described in chapter 2 of this book.

In a large-scale study of word processor usage Page et al. (1996) have found that a majority of users (92%) do tailor their word processors in various ways, such as setting parameters in preference forms to customize the systems. However, most users consider this difficult because they are not primarily interested in tailoring a system, but in using it to accomplish tasks, such as writing reports, drawing diagrams and coordinating work with others. A reason for the discrepancy between doing everyday tasks and tailoring computer applications is that tailoring tools are not well integrated with today's computer applications (Page et al., 1996) and work routines (Mackay, 1990). A challenge for future software developers is to build applications with support for seamless transition between using and programming and allowing users without particular computer background to develop their own applications, with the ultimate aim of empowering people to flexibly evolve advanced information systems.

MacLean et al. (1990) claim there is not a sharp distinction between using an application and programming it. They have identified several intermediate roles along a "tailoring slope." A tailor is characterized as a *tinkerer* or *handyman* who is bounded by *regular user*, on the one end of the slope, and *programmer*, on the other. Handyman and tinkerers have more tailoring skills than regular users, but they are not perceived as programmers.

Mackay (1990) reports similar findings in studies of tailoring behaviour in a research organisation. She observed that many users are not able to tailor their applications without help from more capable users, referred to as *translators*. A translator is a person who is able to take a user's problem and translate it into an implementable solution. The translator's role is often associated with the work of a system administrator or a "super user" in a user organisation.

This chapter presents an approach to bridge the gap between using an application and programming it by providing end users with easy-to-use tailoring tools that operate on application units. The chapter is organized as follows. Aspects

18 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/aspect-oriented-software-components/4216

Related Content

An Adaptive ICT-Enabled Model for Knowledge Identification and Management for Enterprise Development

Agnes Mindila, Anthony Rodrigues, Dorothy McCormick and Ronald Mwangi (2014). *International Journal of System Dynamics Applications* (pp. 71-89).

www.irma-international.org/article/an-adaptive-ict-enabled-model-for-knowledge-identification-and-management-for-enterprise-development/109066

Case Studies in Transformation towards Industrial Sustainability

Lloyd Fernando and Steve Evans (2015). *International Journal of Knowledge and Systems Science* (pp. 1-17).

www.irma-international.org/article/case-studies-in-transformation-towards-industrial-sustainability/130745

Resilient and Timely Event Dissemination in Publish/Subscribe Middleware

Christian Esposito and Domenico Cotroneo (2012). *Technological Innovations in Adaptive and Dependable Systems: Advancing Models and Concepts* (pp. 1-20).

www.irma-international.org/chapter/resilient-timely-event-dissemination-publish/63571

Dynamic Evolution of Knowledge Modules Contexts Oriented to Business Process

Xing Shi, Hongfei Zhan, Junhe Yu and Rui Wang (2019). *International Journal of Knowledge and Systems Science* (pp. 15-26).

www.irma-international.org/article/dynamic-evolution-of-knowledge-modules-contexts-oriented-to-business-process/236679

Self-Adaptable Discovery and Composition of Services Based on the Semantic CompAA Approach

J. Lacouture and P. Anioté (2013). *Innovations and Approaches for Resilient and Adaptive Systems* (pp. 150-170).

www.irma-international.org/chapter/self-adaptable-discovery-composition-services/68949