IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Hershey PA 17033-1117, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITB7519

Chapter 17

Specification of Components Based on the WebComposition Component Model

Martin Gaedke
University of Karlsruhe, Germany,

Klaus Turowski
University of the Federal Armed Forces, Munich, Germany

Developing application systems that use the World Wide Web (WWW, Web) as an application platform suffers from the absence of disciplined approaches to develop such Web-applications. Besides, the Web's implementation model makes it difficult to apply well-known process models to the development and evolution of Web-applications. On the other hand, component-based software development appears as a promising approach that meets essential requirements of developing and evolving highly dynamic Web-applications. With respect to Web-applications, its main objective is to build Web-applications from (standardized) components. Founded on these insights and based on a dedicated component model, we propose an approach to a disciplined specification of components.

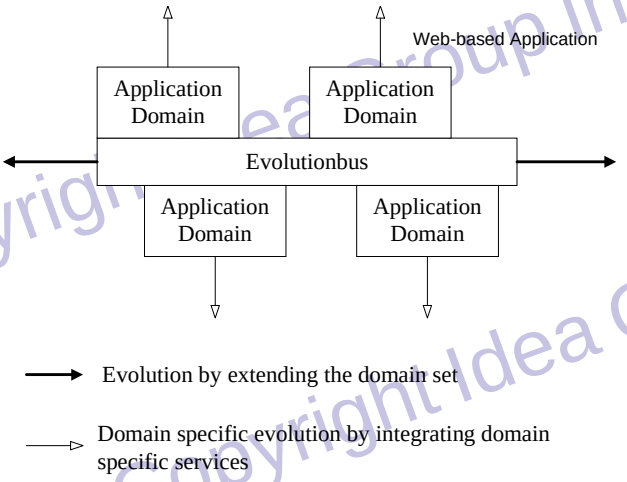
WEBCOMPOSITION COMPONENT MODEL

The *WebComposition* component model (Gellersen & Gaedke, 1999) describes the way of composing Web-applications from components. It bridges the gap between design and implementation by capturing whole design artifacts in components of arbitrary granularity. The resolution of a component is not preset but can vary depending on the level of detail required by the design concept in

Previously Published in *Managing Information Technology in a Global Economy,* edited by Mehdi Khosrow-Pour, Copyright © 2001, Idea Group Publishing.
This chapter appears in the book, Data Warehousing and Web Engineering by Shirley Becker.

question. A component may represent, e.g., an atomic feature such as the font size attribute, a complex navigation structure, implementations of hypermedia design-patterns, or simply compositions of other components. In this way, WebComposition supports the bridging of the gap between the design and the implementation model by offering a high-resolution implementation model relying on code-abstractions. We construct complete target language resources by compiling compositions of these components. In the following sub-sections, we describe the WebComposition approach, which is based on the WebComposition component model. The complete WebComposition approach defines a disciplined procedure of composing Web-applications with components (Gaedke, 2000). It is a synthesis of a component-oriented process model with a dedicated Web-application framework, reuse management, and a dedicated component-technology.

Figure 1: Dimensions of a Web-application's evolution space



WebComposition Process Model

The requirements for a software system change as time goes by. It is obvious that many kinds of influences are responsible for this, e.g., new regulations, changes in corporate identity or an extension of functionality. Such maintenance tasks are difficult to handle if we did not design the application with the possibility of later changes and extensions in mind.

The *WebComposition Process Model* focuses on the evolution of Web-applications by reusing components. It consists of three main-phases. The phases are derived from the common phases of (object-oriented) process models as well as solutions addressing the need of software reuse, and taking the principles of the