

Chapter 12

Component-Based Modeling for Information Systems Reengineering

Malleswara Talla

Concordia University, Canada

Raul Valverde

Concordia University, Canada

ABSTRACT

An Information System can be envisioned as a set of interdependent components that provide the intended services. The component based modeling serves as a tool for collecting requirements of an Information System in user perspective and business perspective at various stages of software development. The chapter presents a methodology for component based modeling and development of an Information System, starting from the requirements definition phase, arriving at candidate components and creation of final components and their interfaces. The methodology aims at clarifying the intricate details and usage of an Information System via business type models and use-case models. The chapter presents the interaction diagrams in order to describe interactions among objects in systems perspective, and context diagrams for reflecting upon the business domain. Finally, the chapter proposes component replacement as a methodology for system reengineering, and model-view-control framework for component refinement and evolution in order to achieve a reengineered information system that reflects upon current requirements in business domain. The reengineering techniques proposed in this chapter can be applied to legacy systems to turn them into a component-oriented reengineered system.

INTRODUCTION

Recent trend in systems engineering is component-based using the technologies such as Component Object Model (COM), Common Object Request Broker Architecture (CORBA), Easy Java Simula-

tions (EJS), etc. A *component-based model* is an interconnected set of software components that collectively represent an information system. A component is a modular piece of software that provides a service. Components communicate via interfaces. Component-based software architecture allows system design in a pragmatic manner, either for developing a new system or

DOI: 10.4018/978-1-4666-4301-7.ch012

reengineering an existing system via reverse engineering where needed. The *reengineering* of a system is actually examining and modifying the system that has been in-use for a long time, to improve it and turn it up-to-date in terms of business rules and system performance needs. Reverse engineering is proposed in (Landry Chouambe, Benjamin Klatt, and Klaus Krogmann, 2008) when the interfaces are not explicit or composite components are not supported in a system. An adaptive component model can be dynamic and evolve with the changing needs via adaptation to different execution contexts (Xin Peng, Yijian Wu, Wenyun Zhao, 2007). The component-based modeling offers an opportunity to speed up the systems development via Commercial Off-The-Shelf (COTS), Open Source, or In-House components readily available for inclusion during design stage (Pasquale Ardimento, Giovanni Bruno, Danilo Caivano, Giuseppe Visaggio, 2007). The component based modeling also promotes an incremental specification validation and runtime adaptability in a distributed component information systems (Nasreddin Aoumeur, Kamel Barkaoui, Gunter Saake, 2007). An *information system* is a computer application that efficiently processes, stores and relays information within an organization and across its customers, suppliers, and all other public relationship organizations. The component-based models are so flexible that these models can support dynamic decisions in concurrent via component selection and reuse at runtime (Biplav Srivastava, 2004). Likewise, component based modeling resulted in efficient system design, development, and reengineering.

This chapter presents a methodology for component-based modeling and development of an information system, starting from the requirements definition phase, identification of components and their interfaces. The methodology uses *business type* models and *use case* models for understanding the requirements and identifying the system components. A business type model is a conceptual map of all data and information

of interest for the information system. A use Case model is a description of steps that reflect upon a usage scenario or a system feature with respect to a system user or actor. The component interaction diagrams describe the interactions among objects in the systems perspective, and context diagrams for reflecting upon the business domain. While design follows a well defined methodology, system reengineering is proposed via component replacement strategies, where complexity of components is analyzed for appropriate resizing of components. The same methodology can be used for reengineering legacy information systems where parts of legacy system features can be turned into new components or replaced with existing components.

COMPONENT BASED SYSTEM MODELING

A component is a reusable piece of software that offers a defined service as a part of a system in which it is used. The challenge in component based architecture and design is to reuse as many existing components and designing new ones that are not already available. Component-based modeling includes several activities to start with an objective of defining the requirements of an information system at the technical and user level. The first phase of the component-based modeling methodology is to capture the requirements and preparing an appropriate *business type models* which serves as a basis for a *component architecture* by identify component specifications, interfaces and dependencies. The Figure 1 presents a series of activities performed during requirements definition phase of component-based systems development.

The system requirements are captured via *use case* model, and appropriate system components are identified (Cheesman, J. & Daniels J. 2001). The business type models define the information in terms of software specifications, whereas the Use Case modelling provides the requirements in

13 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/component-based-modeling-information-systems/77707

Related Content

Providing Automated Holistic Process and Knowledge Assistance during Software Modernization

Gregor Grambow, Roy Oberhauser and Manfred Reichert (2014). *Uncovering Essential Software Artifacts through Business Process Archeology* (pp. 20-63).

www.irma-international.org/chapter/providing-automated-holistic-process-and-knowledge-assistance-during-software-modernization/96613

Service Identification and Specification with SoaML

Michael Gebhart (2013). *Migrating Legacy Applications: Challenges in Service Oriented Architecture and Cloud Computing Environments* (pp. 102-125).

www.irma-international.org/chapter/service-identification-specification-soaml/72214

Performing Security on Digital Images

Abdallah Soualmi, Lamri Laouamer and Adel Alti (2019). *Exploring Security in Software Architecture and Design* (pp. 211-246).

www.irma-international.org/chapter/performing-security-on-digital-images/221718

An SMT-based Approach for Generating Coverage Oriented Metamodel Instances

Hao Wu (2016). *International Journal of Information System Modeling and Design* (pp. 23-50).

www.irma-international.org/article/an-smt-based-approach-for-generating-coverage-oriented-metamodel-instances/170518

Improving Memory Management Security for C and C++

Yves Younan, Wouter Joosen, Frank Piessens and Hans Van den Eynden (2010). *International Journal of Secure Software Engineering* (pp. 57-82).

www.irma-international.org/article/improving-memory-management-security/43926