UNIVERSITI PUTRA MALAYSIA

MEDIATOR CONNECTOR FOR COMPOSING LOOSELY COUPLED SOFTWARE COMPONENTS

Hamid Sanatnama

FSKTM 2009 6

MEDIATOR CONNECTOR FOR COMPOSING LOOSELY COUPLED SOFTWARE COMPONENTS

By

Hamid Sanatnama

Thesis Submitted to the School of Graduate Studies, University Putra Malaysia in Fulfilment of the Requirement for the Degree of Doctor of Philosophy

June 2009

DEDICATION

To my dearest wife and my lovely children for the endless support, encouragement and patience, and also those who made this study possible.

Abstract of thesis presented to the Senate of University Putra Malaysia in fulfillment of the requirement for the degree of Doctor of Philosophy

MEDIATOR CONNECTOR FOR COMPOSING LOOSELY COUPLED SOFTWARE COMPONENTS

By

HAMID SANATNAMA

June 2009

Chairman: **Associate Professor Abdul Azim Abdul Ghani, PhD**

Faculty: **Computer Science and Information Technology**

Component-Based Software Development (CBSD) is an approach that has many benefits, such as improving application developer productivity, reducing costs and complexity by reusing of existing codes. Programming within this approach is like assembly (i.e. composing software out of prefabricated components) rather than development, which reduces skill requirements, and allows expertise focuses on domain problems. The foundation of any CBSD methodology is its underlying component model, which defines what components are, how they can be constructed, and specifies the standards and conventions that are needed to enable composition of independently developed component. The current component models do not support composition in both design and deployment phase. They also focus on the specification and packaging of components but provide almost no support for the easy composition of components. Component in these models uses either direct or indirect message passing as connection schemes, which leads to tightly coupling (i.e. components mix computation with

control). It is conclude that this research has proposed an effective way for component composition which provides loosely coupling between composed components. For system maintenance and evolution, this decoupling should make it simpler to manage changes in the components, and also changes in the connector separately. This research has resulted in the proposed of mediator connector which is similar to a communication hub. It initiates method calls and manages the returns, and also provides total loosely coupling between components and also itself. Mediator connector is a framework and can be reused without any modification. The components composition using mediator connector belongs to the deployment phase. Our approach is based on interactions between components as a subset of behavior in a system. In order to minimize coupling between components and mediator connector, we have designed and developed an XML-based language, called Component Interaction Markup Language (CIML), where components as well as their interactions are described in a CIML document which are used by mediator connector. For evaluation of mediator connector in order to measure the loosely coupling it provides, four case studies have been tested. To measure coupling we applied Coupling Between Object (CBO) software metric. The result shows that mediator connector provides totally loosely coupling between software components composing in the deployment phase.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

PENGHUBUNG PENENGAH UNTUK MENGKOMPOS KOMPONEN PERISIAN KEBERGANTUNGAN LONGGAR

Oleh

Hamid Sanatnama

June 2009

Pengerusi: **Profesor Madya Abdul Azim Abd. Ghani, PhD**

Fakulti: **Sains Komputer dan Teknologi Maklumat**

Pembangunan Perisian Berasaskan Komponen (PPBK) adalah satu pendekatan yang mempunyai banyak faedah, seperti mempertingkatkan produktiviti pembangun aplikasi, mengurangkan kos dan kompleksiti dengan mengguna semula kod yang ada. Pengaturcaraan dalam kaedah ini adalah seperti pemasangan (iaitu mencantumkan perisian menggunakan komponen yang telah terhasil) tidak seperti pembangunan, yang mana mengurangkan keperluan terhadap kemahiran, dan membenarkan pakar untuk memfokus kepada permasalahan utama. Asas kepada mana-mana metodologi CBSD adalah model komponen asasnya, yang mana mentakrifkan apakah dia komponen, bagaimana untuk membinanya, dan menspesifikasi piawaian dan peraturan yang diperlukan untuk membolehkan pembentukan komposisi komponen yang dibangunkan berasingan. Model komponen semasa tidak menyokong komposisi dalam kedua-dua fasa reka bentuk dan penyerahan. Mereka memfokus kepada spesifikasi dan pembungkusan komponen tetapi hampir tidak menyokong untuk mempermudahkan komposisi bagi komponen. Komponen dalam model ini menggunakan penyerahan mesej sama ada

secara langsung atau tidak langsung sebagai skema penghubungnya, yang menyebabkan kebergantungan yang tinggi (iaitu pengiraan komponen berserta kawalan). Kaedah yang dijelaskan di dalam tesis ini adalah hasil daripada penyelidikan untuk mencadangkan kaedah yang lebih berkesan bagi komposisi komponen yang mana menghasilkan kebergantungan yang longgar di antara komponen yang dibangunkan. Bagi penyelenggaraan sistem dan evolusi, penyah-kebergantungan sepatutnya menyebabkan pengurusan perubahan dalam komponen yang lebih mudah, dan juga perubahan dalam penghubung secara berasingan. Penyelidikan ini membuahkan hasil kepada cadangan satu penengah penghubung yang serupa seperti hub komunikasi. Ia memberi nilai awal kepada panggilan 'method' dan menguruskan 'returns', dan juga menyediakan kebergantungan yang sangat longgar antara komponen dan dalamannya. Penengah penghubung adalah sebuah rangka kerja dan boleh diguna semula tanpa sebarang pengubahsuaian. Komposisi komponen menggunakan penengah penghubung dimiliki oleh fasa penyerahan. Pendekatan kami adalah berdasarkan interaksi antara komponen sebagai subset bagi perlakuan dalam sesebuah sistem. Bagi tujuan meminimakan kebergantungan antara komponen dan penengah penghubung, kami telah merekabentuk dan membangunkan sebuah bahasa berasaskan XML, dipanggil Component Interaction Markup Language (CIML), di mana komponen serta interaksinya dijelaskan di dalam dokumen CIML yang akan digunakan sebagai penengah penghubung. Tesis ini tidak merangkumi komposisi komponen pada fasa reka bentuk, tetapi kami percaya bahawa penengah penghubung dan CIML boleh digunakan walaupun pada komposisi fasa reka bentuk. Bagi tujuan ini kami juga mencadangkan rangka kerja yang menyokong komposisi di kedua-dua fasa reka bentuk dan penyerahan menggunakan penengah penghubung dan CIML. Bagi penilaian penengah penghubung untuk mengukur

kelonggaran kebergantungan yang disediakan, empat kajian telah dilakukan. Untuk mengukur kebergantungan kami menggunakan metrik perisian 'Coupling Between Obejct' (CBO). Keputusan menunjukkan bahawa penengah penghubung menghasilkan kebergantungan yang sangat longgar antara komponen perisian yang dikompos semasa fasa penyrahan.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the kindness of several individuals. I would like to express my gratitude and deepest appreciation to my supervisor, Assoc. Prof. Dr. Abdul Azim Abdul Ghani for his kindness and significant mentoring all the time and for offering his guidance and insightful comments in the aspects of developing mediator connector using CIML. Assoc. Prof. Mohd. Hasan Selamat also deserve a great deal of thanks not only for his practical guidelines, but also for sponsoring my PhD study through the Graduate Research Assistant (GRA) scheme for his research, and Dr Rodziah Atan for her kindness, and helpful advices specially for starting my thesis writing.

I am also deeply thanking all the staffs working in Faculty of Computer Science and Information Technology in UPM, especially technical staff.

My special thanks go to my dearest wife, Mahnaz Radboo, my children Elnaz and Iman for their understanding and patience throughout the course of my research and the preparation of this dissertation. I would also like to express my uttermost appreciation to Assoc. Prof. Dr. Faramarz Sadeghi, and Assoc. Prof. Dr. Hamid Khosravi from Faculty of Mathematic and Computer Science in Shahid Bahonar University of Kerman, who made the opportunity for me to continue my study. And also a great deal of thanks to all others who have helped me in one way or another, specially my best friend Reza Meimandi Parizi.

This thesis submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee are as follows:

Abdul Azim Abdul Ghani, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Mohd. Hasan Selamat
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Rodziah Binti Atan
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

AINI IDERIS, PhD
Professor/Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

DECLARATION

I hereby declare that the thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UPM or other institutions.

_____

HAMID SANATNAMA

Date: 30 June 2009

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

API	Application Programming Interface
CIML	Component Interaction Markup Language
CBSD	Component-Based Software Development
GUI	Graphical User Interface
JAXP	Java API for XML Processing
JDOM	Java Document Object Model
JVM	Java Virtual Machine
SAX	Simple API for XML
W3C	World Wide Web Consortium
XML	eXtensible Markup Language
XSD	XML Schema Definition
XSLT	eXtensible Stylesheet Language Transformation

CHAPTER 1

INTRODUCTION

1.1 Background

The design process in most engineering disciplines is based on reuse of existing systems or components. Mechanical or electrical engineers do not normally specify a design where every component has to be manufactured specially. They base their design on components that have been tried and tested in other systems.

Reuse-based software engineering can be compared to software engineering strategy where the development process is geared to reusing existing software. Although the benefits of reuse have been recognized for many years (Mcllory, 1968), it is only the past 10 years that there has been an evolution from original software development to reuse-based development.

Many techniques have been developed to support software reuse during the past 20 years (Sommerville, 2004). Reuse is possible at different levels (from simple function to complete application). There are a number of ways to support software reuse:

- Legacy system wrapping
- Design patterns
- Program libraries

- Configurable vertical application
- Components Off The Shelf (COTS) integration
- Application product lines
- Program generators
- Component frameworks
- *Component-based Software Engineering* (CBSE)
- Aspect-oriented software development
- Service-oriented systems

Component-based Software Engineering (CBSE) emerged in the late 1990s as a reuse-based approach to software system development. CBSE is the process of defining, implementing and integrating or composing the loosely coupled and independent developed components into system.

The foundation of any Component-Based Software Development (CBSD) methodology is its underlying component model, which defines what components are, how they can be constructed, how they can be composed or assembled. Components communicate with their environment only through their interfaces, so it is the interface which provides all the information needed.

1.1.1 Composition

Generally *composition* means taking two or more constructs and putting them together in some way. Composition is a constructive operation—its result is a new thing that has