



شبكة المعلومات الجامعية

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



شبكة المعلومات الجامعية
@ ASUNET



شبكة المعلومات الجامعية التوثيق الالكتروني والميكروفيلم



شبكة المعلومات الجامعية

جامعة عين شمس

التوثيق الالكتروني والميكرو فيلم

قسم

نقسم بالله العظيم أن المادة التي تم توثيقها وتسجيلها
علي هذه الأفلام قد أعدت دون أية تغيرات



يجب أن

تحفظ هذه الأفلام بعيدا عن الغبار

في درجة حرارة من ١٥-٢٥ مئوية ورطوبة نسبية من ٢٠-٤٠%

To be Kept away from Dust in Dry Cool place of
15-25- c and relative humidity 20-40%

بعض الوثائق الأصلية تالفة

بالرسالة صفحات لم ترد بالاصل



Cairo University
Faculty of Computers & Information
Department of Computer Science

Automated Reverse Engineering of Graphical User Interfaces

BY
Hani Samir Girgis

Under the supervision of

Prof. Dr. Reem Bahgat
Dr. Amr Kamel

A Thesis Submitted to the
Faculty of Computers and Information
Cairo University
In partial fulfillment of the Degree of
Master of Science
In
COMPUTER SCIENCE

October 2007

Approval Sheet

Name: Hani Samir Girgis

Degree: Master of Science

Title of Thesis: Automated Reverse Engineering of Graphical User Interfaces

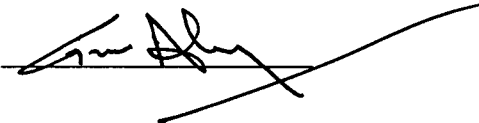
Examining Committee:

Prof. Gamal Mohamed Ali

Examiner

Dept. of Computer and Systems Engineering

Faculty of Engineering, Ain Shams University

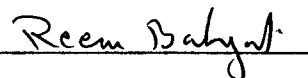
Signature: 

Prof. Reem Bahgat

Supervisor and Examiner

Dept. of Computer Science

Faculty of Computers and Information, Cairo University

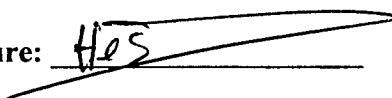
Signature: 

Dr. Hisham Ahmad Hassan

Examiner

Dept. of Computer Science

Faculty of Computers and Information, Cairo University

Signature: 

Date Approved: 17/2/2008

Certification

I certify that this work has not been accepted in substance for any academic degree and is not currently being submitted in candidature for any other degree.

Any portions of this thesis for which I am indebted to other sources are mentioned and explicit references are given.

Student Name: Hani Samir Girgis

Signature: Hani Samir

Table Of Contents

Table Of Contents.....	ii
List of Tables.....	v
List of Figures	vi
Abbreviations.....	viii
Abstract	ix
Chapter 1. Introduction	1
1.1 Introduction	2
1.2 Problem Definition	3
1.3 Objectives	5
1.4 Poposed Solution	6
1.5 Thesis Organization.....	7
Chapter 2. Background.....	8
2.1 Introduction	9
2.2 Redevelopment	9
2.3 Reengineerng:.....	10
2.3.1 White-Box Reengineering	10
2.3.2 Black-Box Reengineering	11
2.3.3 Choosing the Appropriate Migration Approach	11
2.4 Interaction-based Reengineering	12
2.4.1 Motivation	12
2.4.2 A Methodology for Interaction Reengineering	14
2.4.3 Applications of Interaction Reengineering.....	17
2.4.3.1 From Character-based UIs to Web and WAP Interfaces.....	17
2.4.3.2 From Websites to Web Services.....	20
2.4.3.3 From Form-based User Interfaces to Web Services.....	23
2.4.4 Advantages and Limitations	26
2.5 User Interface Reverse Engineering.....	28
2.5.1 User Interface Reverse Engineering Related Work.....	28
2.5.2 Other Applications of User Interface Reverse Engineering	32
2.5.3 Dynamic Analysis	32

2.5.4 Aspect Oriented Programming	33
2.5.5 User Interface Representation Language - XUL	34
2.6 Conclusion	35
Chapter 3. GUI Reverse Engineering.....	39
3.1 Architecture of Executing Java-Swing GUI Applications	40
3.2. GUI Representations	44
3.2.1 Types of GUI Representation for an Executing GUI	45
3.3 The Proposed GUI Representations	46
3.3.1 GUI Sea	47
3.3.2 GUI Changes XML	49
3.4 The Proposed GUI Reverse Engineering Method	50
3.4.1 Introduction	50
3.4.2 The GUIRE Aspect.....	51
3.4.2.1 Window and Widget Extraction (WWE)	52
3.4.2.2 Active Events Detection (AED)	56
3.4.2.3 GUI Inputs Detection (GID).....	64
3.4.3 The Change-Monitor Aspect	68
3.4.3.1 GUI Changes Detection (GCD).....	69
3.4.3.2 XUL Extraction (XULEx).....	72
Chapter 4. Swing2Script Method: Interaction-Based Proposed Solution	76
4.1 Introduction	77
4.2 Application Architecture after the Swing2Script Migration	82
4.2.1 The change-monitor aspect.....	82
4.2.2 XUL Window Structure and Behavior	83
4.2.3 GUI Robot	84
4.3 The Swing2Script Migration Process.....	84
4.3.1 The Reverse Engineering Phase	85
4.3.2 XUL Window Behavior Generation Phase	86
4.3.3 Server Deployment Phase.....	89
4.4 The Swing2Script Interaction Process.....	84

Chapter 5. Case Study	92
5.1 Introduction	93
5.2 The Migration Process Overview	93
5.3 Run-time Interaction in the Migrated Application	97
Chapter 6. Conclusion and Future Work.....	101
6.1 Conclusion.....	102
6.2 Future Work.....	106
References	108

List of Tables

Table 3.1 Comparison between GUI State Independent and Dependent Representation	46
Table 3.2 Corresponding Swing and XUL widgets.	73

List of Figures

Figure 2.1 Comparison of Web Migration Techniques.....	12
Figure 2.2 A General Methodology for Interaction Reengineering	15
Figure 2.3 The CelLEST UI Reengineering Process.....	19
Figure 2.4 Interaction Reengineering of Websites to Web Services.....	21
Figure 2.5 Interaction Reengineering of Form-based CUIs to Web Services.....	24
Figure 2.6 System Architecture for Form-based CUI Reengineering to Web Services	26
Figure 2.7 GUI Ripper Context Diagram	30
Figure 3.1 SW Architecture of an executing monolithic Java-Swing Desktop Application .	40
Figure 3.2 GUI State Change during execution	41
Figure 3.3 The effect of Event Handler on GUI State.....	43
Figure 3.4 Types of runtime GUI Representations	45
Figure 3.5 The meta model of the GUI Sea.....	47
Figure 3.6 Login Window	48
Figure 3.7 Example of GUI Sea model	48
Figure 3.8 Window Island of the login window.....	49
Figure 3.9 The meta model of the GUI Changes XML.....	50
Figure 3.10 GUIRE Aspect weaved into Java Application.....	52
Figure 3.11 Java-Swing window containment structure	52
Figure 3.12 Procedure for widget extraction.....	53
Figure 3.13 Window Appearance detection pointcut.....	54
Figure 3.14 Event Listeners and Event Handlers in Java-Swing	57
Figure 3.15 Procedure for detecting Event Listeners	58
Figure 3.16 Procedure for detecting execution of code in Event Handlers.....	60
Figure 3.17 Code detection using AspectJ Recording Technique.....	64
Figure 3.18 Procedure for detecting GUI Input	66
Figure 3.19 GUI Input detection using AspectJ Recording Technique.....	68
Figure 3.20 Change-Monitor AspectJ weaved into Java Application.....	69
Figure 3.21 Procedure for detecting GUI Changes	71

Figure 3.22 EH GUI Change detection pointcut	72
Figure 3.23 Procedure for Window Structure Extraction.....	74
Figure 4.1 Traditional Screen Scrapping.....	78
Figure 4.2 The Software Architecture of a Migrated Application	81
Figure 4.3 Window Behavior Generator	87
Figure 4.4 Steps for generating Window Behavior JavaScript	88
Figure 4.5 Swing2Script Interaction Process	90
Figure 5.1 A Frame of the User Interface of the Swing Application	92
Figure 5.2 A Window Island file for the address book frame	94
Figure 5.3 A Window Behavior JavaScript file for the address book frame	96
Figure 5.4 A Frame of the User Interface of the Migrated Application	97
Figure 5.5 The XUL Window Representation	98
Figure 5.6 The GUI Event XML as communicated by the client to the server.....	99
Figure 5.7 The GUI changes XML as communicated by the server to the client	100

Abbreviations

AOP: Aspect Oriented Programming

CUI: Character-based User Interface

EH: Event Handler

FUI: Form-based User Interface

GUI: Graphical User Interface

RE: Reverse Engineering

XUL: XML User interface Language