

بسم الله الرحمن الرحيم

Sudan University of Science and Technology

College of Graduate Studies

**Model Driven Architecture For Configuring
Test Script in Model Based Testing**

**استخدام المعمارية المبنية على النماذج لتكوين برنامج نصي
تنفيدي**

في الاختبارات المبنية على النماذج

By:

Abeer Abdalla Sid Ahmed Ibrahim

Supervisor:

Dr .Abd Algafar Hamid Ahmed Ali

Dec, 2013

ABSTRACT

MBT is the automatic generation of software test procedures, using models of system requirements and behavior. This type of testing requires significantly more up-front effort in building the model, but it offers more advantages over traditional software testing methods. The increase of testing effort is growing according to the systems sizes and testing cannot keep pace with the development of complexity and size of systems. MBT involve three main steps: Modeling, test case generation and test execution .Test scripts are a necessary part of the third step where the test case has been generated need to concertize. So for large and complex system writing test script for each test cases for the systems is consuming time and effort, and complex to maintain it.

This research finds an approach for automated testing to reduce the time and effort for writing test script and to improve the quality of the products. Previous research in this area has test automation methods, but they are focused in generation of test cases, while this research is focuses on generation of test scripts.

Recently Model Driven Architecture has been emerged as an effective methodology for systems development. The aim of MDA is to decrease the cost of development and increase the quality of the product through automation and making artifact reusable. This is realized through developing Platform independent metamodel (PIM), platform specific metamodel(PSM) and

transformation or mapping rules. Then MDA has machinery to automate the mapping process using standard language like QVT. It is proposed MDA can work for test scripts generation where a PIM metamodel represent the test cases and PSM metamodel represent the test scripts are being developed. An open source MDA toolset, EMF, has been used with a case study to show the possibility of proposed approach.

The result shows test script could be generated from test case abstraction level without need for programming skills because of high abstraction level MDA has afford. So the develop time could be reduced and effort. MDA in test automation improve the test efficiency where any changes to either end could be synchronized automatically and it will simply maintenance.

المستخلص

الاختبار المبني على النموذج عبارة عن توليد اتوماتيكي لاجراءات اختبار النظم ,باستخدام نماذج وسلوك متطلبات النظام. هذه المتطلبات تحتاج الى جهد فى بناء النموذج ,لكنها توفر مزايا اضافية على الطرق القديمة لاختبار النظم. ان الجهد المبذول يتزايد بتزايد حجم الانظمة والاختبار لايتواكب مع تطور وتعقيد حجم الانظمة.

الاختبارات المبنية على النموذج تتكون من ثلاث خطوات اساسية: النمذجة ,توليد حالات الاختبار وتنفيذ حالات الاختبار.ان نص الاختبار يعتبر جزء مهم فى الخطوة الثالثة بعد توليد حالات الاختبار.كتابة نصوص الاختبار لكل حالات الاختبار للانظمة يستهلك كثيرا من الزمن والجهد وتصعب صيانتها وذلك لكبر وتعقد الانظمة.

هذه الاطرحه تبحث عن منهجية لاتمتة الاختبار لتقليل الزمن والجهد اللازم لكتابة نصوص الاختبار ولزيادة جودة المنتجات. ان الاطروحات السابقة كانت فى مجال طرق الاختبار الالى تركز على توليد حالات اختبار بينما هذا البحث يركز على توليد نصوص اختبار.

ظهر مؤخرا بما يسمى هيكلية قيادة النموذج وهى وسيلة فعالة فى تطوير الانظمة. الهدف من هذه الهيكلية تقليل تكلفة التطوير وزيادة جودة المنتجات عن طريق الاتمتة وجعلها قابلة للاستخدام.ويمكن ذلك عن طريق منصة النموذج المستقل, منصة النموذج الخاصة وقواعد التحويل. هيكلية قيادة النموذج لديها اليه لاتمتة عملية التحويل باستخدام لغات قياسية مثل QVT.تم اقتراح هيكلية قيادة النموذج لتوليد نصوص اختبار, حيث منصة النموذج المستقل تمثل حالات الاختبار, ومنصة النموذج الخاص تمثل نصوص الاختبار. ادوات المصدر المفتوح لهيكلية قيادة النموذج تطبق فى دراسة حالة لتوضيح مدى امكانية المنهجية المقترحة.

النتيجة تظهر ان نصوص الاختبار يمكن تولديها من حالات الاختبار المجردة المستوى بدون الحوجة الى المهارات البرمجية وذلك للتجريد العالى للمستوى المقدمة عن طريق هيكلية قيادة النموذج , والتي تقلل من الجهد والزمن اللازم للتطوير. هيكلية قيادة النموذج فى اتمة الاختبار تحسن من كفاءة عملية الاختبار حيث يمكن مزامنة اى من التغييرات وتبسط عملية الصيانة.

TABLE OF CONTENTS

	ABSTRACT	ii
	TABLE OF CONTENTS	
iv		
	L IST OF FIGURES	
	vii	
	LIST OF TABLES	
	viii	
	LIST OF ABBREVIATION S	ix
CHAPTER 1	INTRODUCTION	1
	1.0 Introduction	
1		
	1.1 Problem statement	
3		
	1.2 Objectives	4
CHAPTER 2	LITERATURE REVIEW	5

5	2.0 Introduction	
	2.1 Test script	5
	2.2 Good script	6
	2.3 Limitations of writing test script	6
	2.4 Scripting techniques	7
	2.4.1 A linear script	7
	2.4.2 Structured scripting	7
	2.4.3 Shared scripts	7
	2.4.4 Data-driven scripts	8
	2.4.5 The keyword-driven script	8
	2.5 Introduction to Model Based Testing	8
	2.6 The advantage of Model Based Testing	9

	2.7 Test case concretization	
10		
	2.8 Automated software testing	
11	2.9 Metamodel	
	12	
	2.10 Model Driven Architecture(MDA)	
13	2.11 The Benefits of MDA	
13		
	2.12 MDA structure	15
	2.13 Transformations	
18		
	2.14 Query View Transform (QVT)	
18		
	2.15 Related work	
20		
CHAPTER 3	RESEARCH METHODOLOGY	
23		
	3.0 Introduction	
23		
	3.1 Methodology steps	
23		
	3.1.1 Developing PIM metamodel	
24		

	3.1.2 Developing PSM metamodel	
26		
	3.1.3 Developing mapping rules	
30		
	3.1.4 Automatic transformation from PIM to PSM	
31		
CHAPTER 4	CONCLUSION	
35		
	4.0 Discussion and conclusions	
35		
	REFERENCES	

LIST OF FIGURES

2.1	Models, languages, metamodels and metalanguages	13
2.2	Transformation	18
3.1	Test case metamodel (PIM)	24
3.2	Withdrawal test case	26
3.3	Test script metamodel (PSM)	27
3.4	Test script using SpecExplorer	29

LIST OF TABLES

2.1	The benefits of MDA	15
2.2	The comparison among MDA approach	
22		
3.1	Mapping rules table	30

LIST OF ABBREVIATIONS

3G	Third Generation
4G	Forth Generation
API	Application Programming Interface
AI	Artificial Intelligence
ATL	Atlas Transformation Language
ATM	Automated Teller Machine
CIM	Computation Independent Model
EMF	Eclipse Modeling Framework
FSM	Finite State Machine
IEEE	Institute of Electrical and Electronics Engineers
MBT	Model-Based Testing
MOF	Meta Object Facility
MDA	Model Driven Architecture
MOF	Meta Object Facility
OMG	Object Management Group
PIM	Platform Independent Model
PIN	Personal Identification Number
PIT	Platform Independent Test model
PSM	Platform Specific Model
QVT	Query/Views/Transformations
SUT	System Under Test
TTCN-3	Testing and Test Control Notation Version 3
UML	Unified Modeling Language
UML2	Unified Modeling Language version 2
XML	Extensible Markup Language
XMI	XML Metadata Interchange