

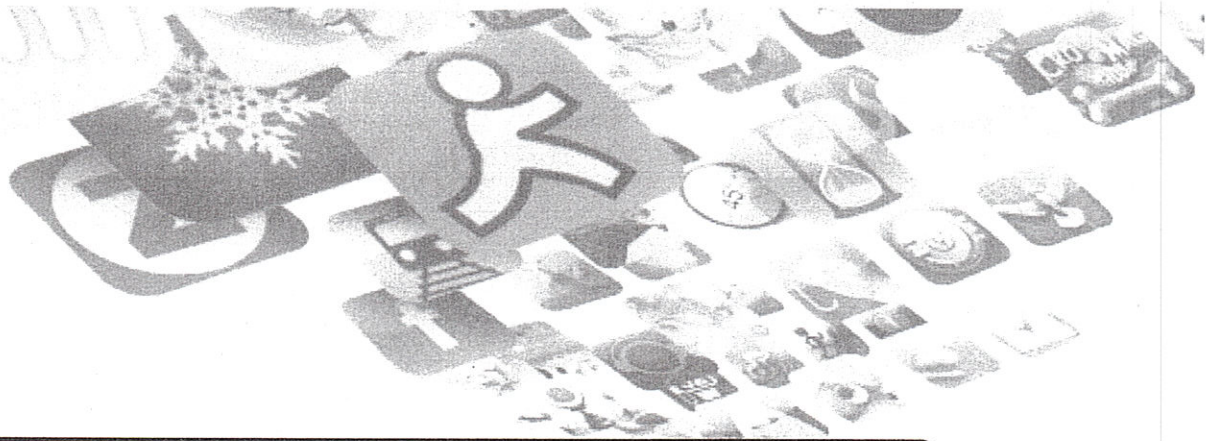
2014 Summer International Academic Conference

Religion and Infotech of the Northeast Asian Nomadic People

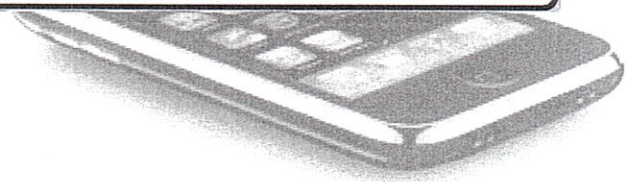
DAY : 8 July(Tue) 2014

- ◆ CONFERENCE SITE : Hulunbuir College Library Seminar Room, China
- ◆ HOST : Institute for Mongolian Studies of Dankook University, Hulunbuir College
- ◆ SUPPORTER : Dankook University, National Research Foundation of Korea





2014 정보기술 연구소 하계 학술대회



동북아 유목민의 정보기술과
소프트웨어 개발 가시화 관리 방안



DANKOOK UNIVERSITY

INFORMATION TECHNOLOGY RESEARCH CENTER

2014년 7월 8일

장소: 헬런 보이르 대학교



세션 1

품질 보증 및 관리를 위한 비 개발자의 산출물 접근 관리 개선 방안 (Improvement method of non-developers access management for quality assurance and control of deliverables)	박용범 (단국대) 이은승 (단국대)
유스케이스 식별을 통한 테스트 케이스 추출 패러다임 (Test Case Extraction Paradigm though Identifying Use Cases)	김영철 (홍익대) 박보경 (홍익대)
사물인터넷을 위한 3차원 지오펜스 프레임워크 설계 (3D Geofence Framework Design for the Internet of Things)	전병국 (강릉원주대)
대몽골제국의 과학기술과 백성분류체계 (A Study on Scientific Technology and Classification System of People in the Great Mongol Empire)	박원길 (칭기스칸 연구센터 소장)

세션 2

함수적 접근 경로 이용한 빅 데이터 테이블화 (BigData Tabilization with Functional Access Path)	김영철 (홍익대) 양호석 (홍익대)
의료 정보 시스템의 웹 접근성 연구 (Web Accessibility for Medical Information System)	김영철 (홍익대) 서채연 (홍익대)
Twister를 활용한 MapReduce 응용 (Utilizing MapReduce using Twister)	강윤희 (백석대)
XPath을 이용한 테스트 케이스 정형화 (Test case formalization using XPath)	김영철 (홍익대) 강건희 (홍익대)

박사학위 초청 세션

Good Software(GS)인증모델 기반의 이종 성숙도 모델 개선 (Improving Heterogeneous Maturity Models Based on Assessing Good Software(GS) Certification Model)	김기두 (홍익대)
Test case Generation from Cause-Effect Graph based on Model Transformation	김동호 (홍익대)

산업체 초청 세션

크로스 플랫폼을 이용한 웹 어플리케이션 활용에 대한 연구 (Studies utilizing the web application using Cross-Platform)	조용균, 최순봉 (주) 빈플릭스 SLRC
---	------------------------------

Test case formalization using XPath

Geon_hee Kang^{1,1} , R.YoungChul Kim¹

¹SE Lab, Dept. of Computer Information Communication, Hongik University
{kang, bob}@selab.hongik.ac.kr,

Abstract. The previous test case formalization doesn't make enough to read the extraction of test cases as XML (MS Words) representations from models (such as UML diagrams). But it is very difficult to read MS Words of XML (test cases), and also not reuse them. With XPath, it can formalize test case (like Excel format) to easily read them.

Keywords: XPath, Test Case, Formalization

1 Introduction

기존의 테스트케이스 정형화 연구[1]에서는 테스트 케이스 상태 다이어그램을 통하여 XML형식으로 추출한 후 그 데이터를 통하여 MS WORD형식의 파일을 생성하였다. 물론 XMI파일을 가지고 재사용을 하여도 상관 없지만 MS WORD형식의 파일을 사용하게 된다면 문서는 가지고 있으나 테스트 케이스 파일(XMI)을 가지고 있지 않다면 재사용하기 힘든 상황이 발생하게 된다. 그래서 쉽게 테스트케이스를 수정, 보완 할 수 있고 문서만 가지고 있다면 다시 테스트 케이스를 추출할 수 있는 Xpath[2]를 통한 Excel유형 방식의 정형화를 제안한다. XPath[2]는 Xml방식의 데이터를 경로식을 사용하여 표현한다. 그렇기 때문에 경로를 사용하면 해당 경로의 데이터를 간단하게 불러올 수가 있다. 그리고 JAVA와 C등 여러 가지 언어에서 사용할 수 있는 API가 있기 때문에 다양한 분야에서 사용될 수 있다. Jxl[3]은 JAVA 상에서 MS Excel형식의 문서를 편집, 수정, 생성할 수 있는 API이다.

본 논문의 순서는 2장 본론에서 XPath 사용을 통한 테스트 케이스 정형화에 대해 설명하고 기존의 상태다이어그램을 통하여 추출된 XMI문서를 가지고 적용사례를 보여준다. 3장에서는 결론을 언급한다.

2 Test case formalization using XPath

fig. 1. Formal tools of test cases (XmlToXls)

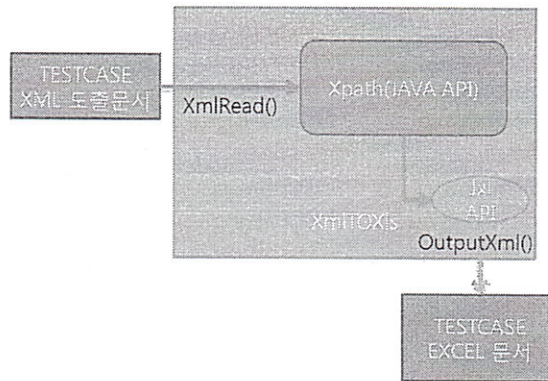


fig. 1.는 XmlToXls(테스트케이스 정형화 도구)의 구조도이다. XML로 구성되어 있는 테스트 케이스 문서를 XmlToXls에서는 XPath[2]로 읽어 들여 Jxl[3] API를 통하여 엑셀문서로 출력한다

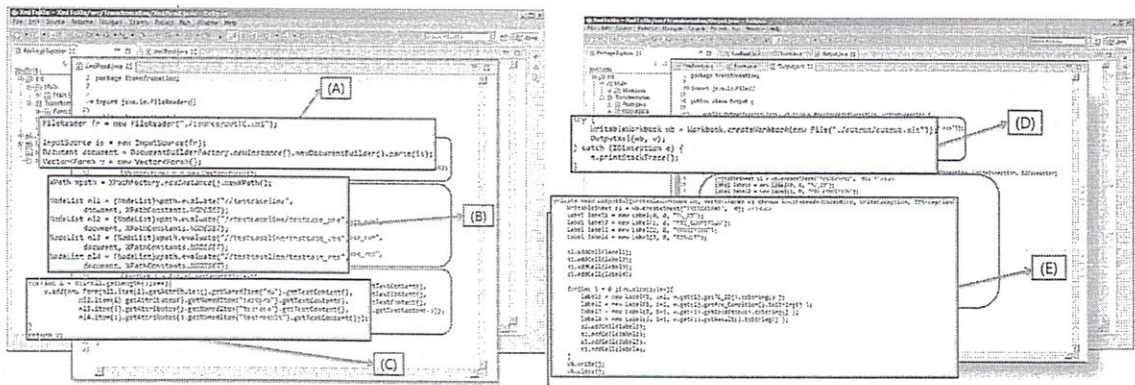


fig. 2. XmlRead(), Output()

fig. 2. 에서의 (A)에서는 File Writer객체를 생성하여 OutTc.xml파일을 읽은 다음 Input Source객체에 입력하여준다 그 이후 Document 객체에 빌더를 파싱 시켜주는 부분이다. 그리고 Form 이라는 구조를 가진 Vector를 사용하여 데이터를 저장하기 때문에 Vector<Form> 객체를 선언하여 준다. Form 구조는 TestCase ID, Pre_condition, Condition, Result 로 구성되어 되어있다. (B)부분 에서는 XPath객체를 선언하여 주고 4개의 노드 리스트 (Test Case ID, Pre_condition, Condition, Result)를 선언한다. (C) 부분에서는 선언한 노드 리스트를 통하여 각 영역에 맞는 데이터를 Vector에 저장을 하게 되고 마지막으로 메소드

에서 Vector를 반환한다. (D)에서는 Workbook객체를 선언을 하여주고 파일의 경로를 지정한다. (E)부분은 OutputXls 메소드 이다. WritableWorkBook객체와 Vector객체를 매개변수로 받고 Vector에 저장되어 있는 Test Case ID, Pre_condition, Condition, Result의 반복문을 통하여 WritableSheet객체에 저장한다. write 메소드를 통해 엑셀 시트에 입력하게 된다.

3. Case Study

본 연구는 직접 개발된 메타모델링 모델 변환을 적용한 UML상태 다이어그램 기반의 자동 테스트 케이스 발생 도구[1]를 이용하여 추출이 된 상태 다이어그램을 이용한 테스트케이스의 XMI파일을 이용하였다.

fig. 4. Part of the TestCase XMI file

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<testcase:TestCaseModel xmi:version="2.0" xmlns:xmi="http://www.omg.org/xmi/2.0" xmlns:tc="http://www.omg.org/spec/TC/2.0" >
  <testcaseline no="TC1">
    <testcase_pre testpre="solenoid_OFF_OFF_OFF=F,Lever_ID or 2|step == AND" />
    <testcase_con testcon="AND" />
    <testcase_res testresult="soleniod_ON_OFF_OFF=F" />
  </testcaseline>
  <testcaseline no="TC2">
    <testcase_pre testpre="solenoid_OFF_OFF_OFF=F,Lever_ID or 2|step == AND" />
    <testcase_con testcon="AND" />
    <testcase_res testresult="soleniod_ON_OFF_OFF=F" />
  </testcaseline>
  <testcaseline no="TC3">
    <testcase_pre testpre="solenoid_OFF_OFF_OFF=F,Lever_ID or 2|step == AND" />
    <testcase_con testcon="AND" />
    <testcase_res testresult="soleniod_ON_OFF_OFF=F" />
  </testcaseline>
</testcase:TestCaseModel>
```

fig. 4. 는 상태 다이어그램을 이용한 테스트 케이스 XMI파일의 일부 이다. 이 XMI의 구조를 XPath표현식으로 보면 다음과 같이 표현할 수 있다 .

```
//testcaseline/testcase_pre
testcase_con
testcase_res
```

fig. 5. XPath NodeList & Extract Excel

```
NodeList n11 = (NodeList)xpath.evaluate("//testcaseline",
document, XPathConstants.NODESET);
NodeList n12 = (NodeList)xpath.evaluate("//testcaseline/testcase_pre",
document, XPathConstants.NODESET);
NodeList n13 = (NodeList)xpath.evaluate("//testcaseline/testcase_con",
document, XPathConstants.NODESET);
NodeList n14 = (NodeList)xpath.evaluate("//testcaseline/testcase_res",
document, XPathConstants.NODESET);
```



TC_ID	PRE_CONDITION	CONDITION	RESULT
TC1	solenoid_OFF_OFF_OFF=F,Lever_ID or 2 step == AND	AND	soleniod_ON_OFF_OFF=F
TC2	solenoid_OFF_OFF_OFF=F,Lever_ID or 2 step == AND	AND	soleniod_ON_OFF_OFF=F
TC3	solenoid_OFF_OFF_OFF=F,Lever_ID or 2 step == AND	AND	soleniod_ON_OFF_OFF=F
TC4	solenoid_OFF_OFF_OFF=F,Lever_ID or 2 step == AND	AND	soleniod_ON_OFF_OFF=F
TC5	solenoid_OFF_OFF_OFF=F,Lever_ID or 2 step == AND	AND	soleniod_ON_OFF_OFF=F

fig. 5. 는 XmlToXsl 내부 코드에서 적용대상에 맞게 XPath 표현식을 통하여 NodeList를 정

의를 하였다. n1은 TestCase의 ID, n2는 pre_condition, n3는 condition, n4는 result의 정보를 가져오기 위한 NodeList이다. XPath표현식을 통해서 데이터를 불러내어 엑셀에 저장을 하게 된다.

4 Conclusion

Xpath[2]와 Jxl[3]을 이용하여 XML로 구성되어 있는 Test Case의 정보를 테스터가 가독성이 좋고, 문서화가 쉬운 엑셀파일로 추출하여 정형화 할 수 있었다. 현재는 Xmi를 읽어서 엑셀파일로 추출만 가능하다. 그러나 지금의 알고리즘을 역 이용한다면 충분히 Xmi파일로도 변환이 가능하다. 결국 문서화 된 파일이 다시 TestCase의 입력파일로서 들어갈 수 있을 것으로 판단된다.

Acknowledgments. This work was supported by the IT R&D Program of MKE/KEIT [10035708, "The Development of CPS (Cyber-Physical Systems) Core Technologies for High Confidential Autonomic Control Software"] and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2011601)

References

1. Kim Jae Seong : A Study on Automatic Test Case Generation Based on UML State Diagram with Meta-modeling and Model Transformation 65-66 (2013)
2. Lim Seong Taek : XML document standards : XML1.1, XSL1.1, XSLT2.0, XPath2.0, XForms1.1, 217- 227 (2010)
3. Andy Khan : Java Excel API - A Java API to read, write, and modify Excel spreadsheets, <http://jexcelapi.sourceforge.net/>