

Automating Test Automation

Saurabh Sinha

(Joint work with Suresh Thummalapenta, Nimit Singhania, and Satish Chandra)



Functional Testing of Enterprise Applications





What is Test Automation?



- 1. Launch the application through the link <u>http://godel.in.ibm.com:8080/online-</u> <u>bookstore/Default.jsp</u>
- 2. Enter the intended book search name as "MySQL" at the "Title" Edit field and select "Category" as "All" by Drop down list and then Click "Search" Button
- 3. Select a title from the list of all Search Results displayed and then click either on the image of the book or on Name of the Book
- 4. Enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button



What is Test Automation?

Manual Test Cases Image: Comparison of the second seco	Test Image: Construction Image: Construction Image: Construction Automated Test Scripts ink					
bookstore/Default.jsp	public void setUp() throws Exception {					
2. Enter the intended book search nam at the "Title" Edit field and select "C	driver = new FirefoxDriver(); driver.manage().timeouts(). implicitlyWait(30, TimeUnit.SECONDS); }					
Button	<pre>public void test() throws Exception { driver.get("/online-bookstore/Default.jsp"); driver.findElement(By_name("name")) clear();</pre>					
3. Select a title from the list of all Sear	driver.findElement(By.name("name")).sendKeys("mysql");					
displayed and then click either on th book or on Name of the Book	driver.findElement(By.cssSelector("input[type=\"submit\"]")).click(); driver.findElement(By.xpath("//tr[2]/td[2]/a/font")).click(); driver.findElement(By.name("Login")).clear();					
4. Enter login "guest" and password "g login	driver.findElement(By.name("Login")).sendKeys("guest"); driver.findElement(By.name("Password")).clear(); driver.findElement(By.name("Password")).sendKeys("guest"); driver.findElement(By.cssSelector("input[type=\"submit\"]")).click();					
5. Enter the Quantity "1" and Click on Shopping Cart" Button	 }					



Is Test Automation a Significant Problem?





Test Automation Techniques

Record and replay?

Fragile, difficult to maintain scripts



Keyword-driven framework?

Effort in developing a general framework

High-level scripting language?

Limited expressivity



Automating Test Automation

- 1. Launch the application through the link <u>http://godel.in.ibm.com:8080/online-</u> <u>bookstore/Default.jsp</u>
- Enter the intended book search name as "MySQL" at the "Title" Edit field and select "Category" as "All" by Drop down list and then Click "Search" Button
- 3. Select a title from the list of all Search Results displayed and then click either on the image of the book or on Name of the Book
- 4. Enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button
- Verify "User Information" and Following "Item" Details of Selected Book, Details Order #, Item Price, Quantity, Total





A Manual Test Case and Tool-Agnostic Representation



- Enter the intended book search name as "MySQL" at the "Title" Edit field and select "Category" as "All" by Drop down list and then Click "Search" Button
- 3. Select a title from the list of all Search Results displayed and then click either on the image of the book or on Name of the Book
- 4. Enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button
- Verify "User Information" and Following "Item" Details of Selected Book, Details Order #, Item Price, Quantity, Total

Action-Target-Data (ATD) Tuples

<goto,

http://godel.in.ibm.com:8080/ online-bookstore/default.jsp, >



A Manual Test Case and Tool-Agnostic Representation





A Manual Test Case and Tool-Agnostic Representation





Challenge: Identifying Relevant Segments

- 1. Launch the application through the link <u>http://godel.in.ibm.com:8080/online-</u> <u>bookstore/Default.jsp</u>
- Enter the intended book search name as "MySQL" at the "Title" Edit field and select "Category" as "All" by Drop down list and then Click "Search" Button
- 3. Select a title from the list of all Search Results displayed and then click either on the image of the book or on Name of the Book
- 4. Enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button
- Verify "User Information" and Following "Item" Details of Selected Book, Details Order #, Item Price, Quantity, Total

Problem

- A manual test step can include multiple segments combined using conjunctions
- Splitting simply based on conjunctions results in invalid segments
- E.g.: Step 4 results in an invalid segment "password", which does not contain any verb

Solution

- Enumerate all possible candidate segmentations and explore each candidate systematically
- Backtrack if a chosen candidate does not lead to the interpretation of the entire test case



Challenge: Filtering Irrelevant Segments

- 1. Launch the application through the link <u>http://godel.in.ibm.com:8080/online-</u> <u>bookstore/Default.jsp</u>
- Enter the intended book search name as "MySQL" at the "Title" Edit field and select "Category" as "All" by Drop down list and then Click "Search" Button
- 3. Select a title from the list of all Search Results displayed and then click either on the image of the book or on Name of the Book
- 4. Enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button
- Verify "User Information" and Following "Item" Details of Selected Book, Details Order #, Item Price, Quantity, Total

Problem

 A test step can contain valid ATD tuples that do not correspond to actions on the user interface

Solution

- Use the application under analysis as a test oracle and ignore irrelevant ATD tuples
- Proceed automatically with the next ATD tuple of the test step



Challenge: Disambiguating Targets



Problem

- The target of a test step may match multiple elements in the user interface
- Reason: testers often do not mention the target properly or the user interface contains multiple elements with the same label

Solution

- Explore all candidate target elements systematically
- Backtrack if a chosen candidate prohibits interpretation of the entire test case



Enter the intended book search name as "MySQL" at the "Title" Edit field and select "Category" as "All" by Drop down list and then Click "Search" Button

Enter the intended book search name as "MySQL" at the "Title" Edit

select "Category" as "All"

Drop down list

then Click "Search" Button













© 2012 IBM Corporation



Synthesis of the Tool-Agnostic Representation



Use the application as oracle

- 1. Can perform ATD on the current page
- 2. Subsequent steps succeed

If the exploration of a candidate flow fails, backtrack to explore an alternative flow



1 <Select, Category, All>

1 <Drop, Down, List> 2 <Drop, List, Down>

1 <Click, Search, Button>
2 <Click, Search, _>
3 <Click, Button, Search>
4 <Click, Button, _>





4. enter login "guest" and password "guest", and click login

5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button





- 4. enter login "guest" and password "guest", and click login
- 4.1.1 enter login "guest"
- 4.1.2 password "guest"
- 4.1.3 click login
- 4.2.1 enter login "guest"
- 4.2.2 password "guest", and click login
- 4.3.1 enter login "guest" and password "guest" 4.3.2 click login
- 4.4.1 enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button





- 4. enter login "guest" and password "guest", and click login
- 4.1.1 enter login "guest"
 - ATDTuple: <enter, login, guest>
- 4.1.2 password "guest"
- 4.1.3 click login
- 4.2.1 enter login "guest"
- 4.2.2 password "guest", and click login
- 4.3.1 enter login "guest" and password "guest" 4.3.2 click login
- 4.4.1 enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button









4. enter login "guest" and password "guest", and click login
4.1.1 enter login "guest"

ATDTuple: <enter, login, guest>

- 4.1.2 password "guest" ATDTuple: none
- 4.1.3 click login
- 4.2.1 enter login "guest"
- 4.2.2 password "guest", and click login
- 4.3.1 enter login "guest" and password "guest" 4.3.2 click login
- 4.4.1 enter login "guest" and password "guest", and click login
- 5. Enter the Quantity "1" and Click on "Add to Shopping Cart" Button















© 2012 IBM Corporation





© 2012 IBM Corporation







Technical Issues

- State restoration problem
- Optimizations for intelligently exploring the search space
 - Look-ahead static checking
 - Local backtracking
 - Active Learning
- Guided automation for incomplete manual test cases
 - Addresses the issue of incomplete manual test cases
 - Steps do not have all necessary information
 - Missing test steps

Login to the application as guest

- Highlights the failing test step and seeks human feedback
 - Reuses the feedback in the future to resolve similar steps



ATA Architecture





Empirical Evaluation

- Empirical Studies
 - Three web applications:
 - Two open-source web applications: *BookStore* and *BugTracker*
 - One enterprise application
 - Research questions
 - How often can ATA automatically interpreted manual test steps, with no human intervention?
 - How often does ATA effectively reuse the human feedback received for the steps that it cannot interpret?
 - Do optimizations help improve the efficiency?
- Evaluation in a Production Environment
 - IBM-internal enterprise web application



RQ1: Interpreting Manual Test Steps

		Test Steps		Successful Steps		Modified Steps			Reused Steps	
	Tests	Action	Verification	Action	Verification	Action	Verification	Inserted Steps	Action	Verification
BookStore	10	49	57	32 (65%)	37 (65%)	5 (10%)	6 (10%)	2	12 (24%)	14 (25%)
BugTracker	10	48	49	26 (54%)	24 (49%)	9 (18%)	15 (31%)	3	13 (27%)	10 (20%)
Арр	13	62	28	42 (68%)	18 (64%)	8 (13%)	10 (36%)	0	12 (19%)	0
	33	159	134	100 (63%)	79 (59%)	22 (14%)	31 (23%)	5	37 (23%)	24 (18%)

- Total number of manual tests: 33 (159 action steps, 134 verification steps)
- Steps that required human intervention:
 - Action steps: 22 / 159 (14%)
 - Verification steps: 31 / 134 (23%)
- Steps with no human intervention:
 - Action steps: 137 / 159 (86%)
 - Verification steps: 103 / 134 (77%)



Summary and Future Work

- A technique for automating test automation that
 - Uses a combination of natural-language processing, backtracking exploration, runtime interpretation, and learning
 - Uses the application as the oracle to determine the correctness of a potential interpretation
- Overcomes many of the limitations of conventional automation techniques
- Empirical results indicate that the technique can be effective in practice
- Evidence from an evaluation in a production environment indicates substantial productivity improvement
- Further evaluation in production environments
- Persistent state restoration
- Improved change resiliency
- Why do such large manual test suites exist? What features do they cover? Can the tests be reduced without compromising "coverage"?