ITM Regression Testing:

Step 1: Finding the regression faults and roll back the related code back to before modified.

We got a based version of ITM.

Step 2: Add monitors at based version of ITM. Run test cases and keep track changes of the pre-condition and post-condition of each function that was executed. The test case is not unit test case, it is usability testing.

**Adding monitors.**

We add monitors inside each function and record all the global variables that function used and its parameters.

If a function contains ajax:

The request url and request body are tracked as precondition of the function

The ajax call back , success callback function is tracked as postcondition of the function.

The ajax will return a data object which is the response of the request. This has been record as postcondition of the function.

If function operation on some global variables, we only focus on write to a global variable which means the function change the global variable ,record these as properties at post-condition of the function

If function does change the HTML DOM,

Just treated as HTML DOM CHANGE, we does keep what kind of change it is.

====

Step 3 Modified program and add monitors.

Modified program one case at a time.

Step 4: Create or modify test cases if necessary

Step 5: Run related test cases to check the performance.

Step 6: Compare the profile generated by running same test cases in based version of program and modified version of program, to find out the change in invariant.

Step 7: Test cases selection based on the affected invariant.

Expected Result:

Regression fault caused by database schema change.

**Problem : Context Missing db table**

JOT -summary is a new function add into ITM it requires a new table jot-summary, however some databases does not have these tables. This cause regression fault.When program running on the database does not have table jot-summary, that will cause the problem.

How invariant can detect this :

We know it require table jot-summary. We find any program refer to jot-summary table’s columns.We only focus at columns problem,idea and more.

|  |
| --- |
|  |

Total”318

Selected test cases :100-13 / 87

Regression fault :13 / 0

Context change. Program change . run on different context.

**Problem : Database content change**

Previous, update/import database function will fresh the all the data and import new data. However function cause some default setting value missed.

How invariant can detect this :

After update /Import the new data, we compare the database between manual updated /import with automatic updated/import found that grade information are missing, Then using grade found all the test cases that relevant to the grade.

Total”318

Selected test cases :143

Regression fault :59

|  |
| --- |
|  |

**Problem : global variable change**

Problem: A global variable is missing. The problem is there are two programmers working at same page. Programer A need to init global variable in programmer B’s code.

However, due to communication issue, the programmer B does not realize it and remove this initial statement code.

After change , re-run the test cases when found that a global variable was changed.

Case 1: threadid was removed

Total”318

Selected test cases :26

Regression fault :25

Case 2: data\_note\_str\_id was removed

Total”318

Selected test cases :14

Regression fault :14

Case 3: global thread\_id was removed

Total”318

Selected test cases :108

Regression fault :31

**Problem : session change**

The program remove a session . However, some pages still use this session value .

How invariant can detect this.

After program was modified by comparison we found that one global variable does not initialized ,find all the program refer to this variable ("threadid", threadid);

Total”318

Selected test cases :108

Regression fault :40

**Problem : KF URL Change.**

Problem: ITM integrated with KF, different KF program may run differently. When ITM runs with KF at HK server,The KF does not support HTTPS and ITM sent HTTPS request to get data. It fails.

--------------------

Different in context. KF URL is different.

Total”318

Selected test cases :46

Regression fault :16

**Problem : Database content Change.**

Problem: After using KF editor, we found that there are lot empty notes there.

How invariant can detect this.

Invariant can find that previous the notes must have some content in title, however, there are lot empty notes in database. Every function using notes need to be retest.

Note\_title,note\_content

Total”318

Selected test cases :159

Regression fault :16

**Problem : DataBase Change. Same with first one.**

The database has been changed , however, some api does not change. .

Note table add new column community\_id, some of database did update there schema, miss the column.

How invariant can detect this.

Find invariants relate to the this database.

Total”318

Selected test cases :158

Regression fault :25

**Problem : Change library JSPDF Which is generate PDF from HTML.**

Total”318

Selected test cases :3

Regression fault :3

**Problem : Javascript library change.**

**Update D3.js to a new version of D3.js**

Total”318

Selected test cases :88

Regression fault :67

Conclusion:

Many regression fault are due to database change or execution context change and inconsistency in operation.

The database changes will affect most part of program, after changes, we know which table and column has been changed. We can find its related invariant in program. However,it may not accurate. By searching we are finding the test case refer to the a invariant name. The invariant can has same name which cause over selection.

For example : Table thread\_note has column name community\_id. We are adding this column. When program runs in different db, we need to find out all the test cases refer to invariant has name community\_id. In project dashboard , there are variable has name community\_id, which come from session and is not refer to the record thread\_note’s community\_id. The change of thread\_note will not affect this at all. However, the algorithm does not tell different between two community\_id,to keep it safe. The alg sellect it both.

For the global variable change, different pages’ variable does not share variable with each other. That means for each page, the global variable’s scope is private. The change of global variable will not affect other pages. The case 3 and case 4’s is modification about global variable . The accurate rate is very high which due to the change global variable only used in this page and does not have variable in other page with same name. case 5 is not the lucky one, the variable name thread\_id is common used in many pages with made this invariant very popular and select 108 test cases, but only 31 was regression fault.