# A Comparative Study of White Box, Black Box and Grey Box Testing Techniques

Mohd. Ehmer Khan

Department of Computer Science,
Singhania University,
Jhunjhunu, Rajasthan, India

Farmeena Khan

Department of Computer Science,
EIILM University,
Jorethang, Sikkim, India

*Abstract*—**Software testing is the process to uncover requirement, design and coding errors in the program. It is used to identify the correctness, completeness, security and quality of software products against a specification. Software testing is the process used to measure the quality of developed computer software. It exhibits all mistakes, errors and flaws in the developed software. There are many approaches to software testing, but effective testing of complex product is essentially a process of investigation, not merely a matter of creating and following route procedure. It is not possible to find out all the errors in the program. This fundamental problem in testing thus throws an open question, as to what would be the strategy we should adopt for testing. In our paper, we have described and compared the three most prevalent and commonly used software testing techniques for detecting errors, they are: white box testing, black box testing and grey box testing.**

*Keywords- Black Box; Grey Box; White Box.*

## I.    INTRODUCTION

Software testing identifies defects, flaws or errors in the application code that must be fixed. We can also define software testing as a process of accessing the functionality and correctness of a software through analysis. The main purpose of testing can be quality assurance, reliability estimation, validation and verification. Software testing is a fundamental component of software quality assurance and represents a review of specification, design and coding. The main objective of software testing is to affirm the quality of software system by systematically testing the software in carefully controlled circumstances, another objective is to identify the completeness and correctness of the software, and finally it uncovers undiscovered errors. [1] [2]

The three most important techniques that are used for finding errors are: [1]

*1)   White Box Testing Technique:* It is the detailed investigation of internal logic and structure of the code. In white box testing it is necessary for a tester to have full knowledge of source code.

*2)   Black Box Testing Technique:* It is a technique of testing without having any knowledge of the internal working of the application. It only examines the fundamental aspects of the system and has no or little relevance with the internal logical structure of the system.

*3)   Grey Box Testing Technique:* White box + Black box = Grey box, it is a technique to test the application with limited knowledge of the internal working of an application and also has the knowledge of fundamental aspects of the system.
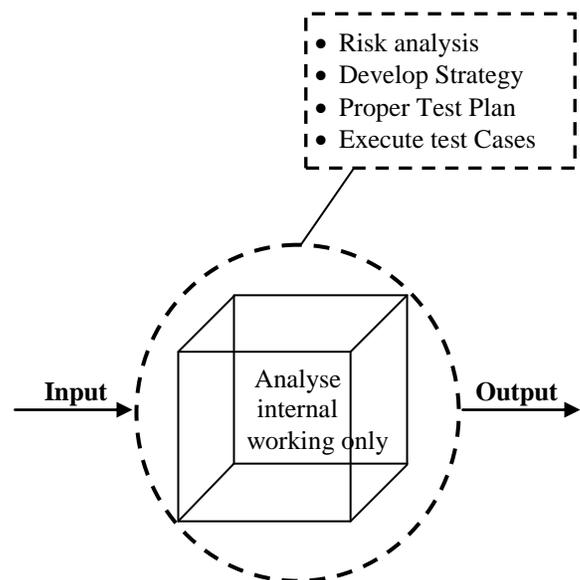
## II.    WHITE BOX TESTING TECHNIQUE



Figure 1.    Represent white box testing

White box testing is a test case design method that uses the control structure of the procedural design to derive test cases. White box testing can uncover implementation errors such as poor key management by analyzing internal workings and structure of a piece of software. White box testing is applicable at integration, unit and system levels of the software testing process. In white box testing the tester needs to have a look inside the source code and find out which unit of code is behaving inappropriately. [3]

Some of the advantages and disadvantages of white box testing technique are listed below: [3] [4]

Advantages

- It reveals error in hidden code by removing extra lines of code.

- Side effects are beneficial.

- Maximum coverage is attained during test scenario writing.

Disadvantages

- It is very expensive as it requires a skilled tester to perform it.
- Many paths will remain untested as it is very difficult to look into every nook and corner to find out hidden errors.
- Some of the codes omitted in the code could be missed out.

Some of the synonyms of white box testing are glass box testing, clear box testing, open box testing, transparent box testing, structural testing, logic driven testing and design based testing.

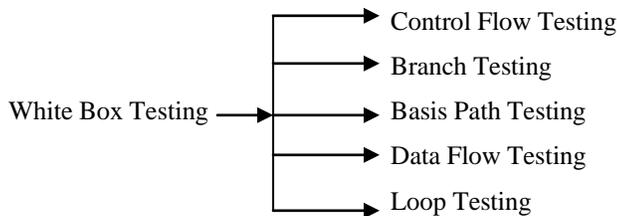Some important types of white box testing techniques are briefly described below: [3]



Figure 2.   Represent different forms of white box testing techniques

*1)   Control Flow Testing:* It is a structural testing strategy that uses the program control flow as a model control flow and favours more but simpler paths over fewer but complicated path.

*2)   Branch Testing:* Branch testing has the objective to test every option (true or false) on every control statement which also includes compound decision.

*3)   Basis Path Testing:* Basis path testing allows the test case designer to produce a logical complexity measure of procedural design and then uses this measure as an approach for outlining a basic set of execution paths.

*4)   Data Flow Testing:* In this type of testing the control flow graph is annoted with the information about how the program variables are define and used.

*5)   Loop Testing:* It exclusively focuses on the validity of loop construct.
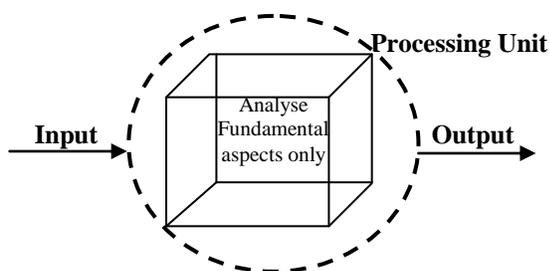
### III.   BLACK BOX TESTING TECHNIQUE



Figure 3.   Represent black box testing

Black box testing treats the software as a "Black Box" – without any knowledge of internal working and it only examines the fundamental aspects of the system. While performing black box test, a tester must know the system architecture and will not have access to the source code. [5]

Some of the advantages and disadvantages of black box testing technique are listed below: [4] [5]

Advantages

- Efficient for large code segment.
- Tester perception is very simple.
- Users perspective are clearly separated from developers perspective (programmer and tester are independent of each other).
- Quicker test case development.

Disadvantages

- Only a selected number of test scenarios are actually performed. As a result, there is only limited coverage.
- Without clear specification test cases are difficult to design.
- Inefficient testing.

Some of the synonyms of black box testing technique are opaque testing, functional testing, close box testing, and behavioural testing.

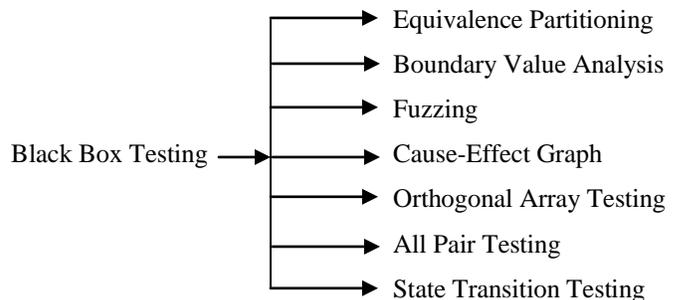Some important types of black box testing techniques are briefly described below: [5]



Figure 4.   Represent differnet forms of black box testing techniques

*1)   Equivalence Partitioning:* It can reduce the number of test cases, as it divides the input data of a software unit into partition of data from which test cases can be derived.

*2)   Boundary Value Analysis:* It focuses more on testing at boundaries, or where the extreme boundary values are chosen. It includes minimum, maximum, just inside/outside boundaries, error values and typical values.

*3)   Fuzzing:* Fuzz testing is used for finding implementation bugs, using malformed/semi-malformed data injection in an automated or semi-automated session.

*4)   Cause-Effect Graph:* It is a testing technique, in which testing begins by creating a graph and establishing the relation between the effect and its causes. Identity, negation, logic OR

and logic AND are the four basic symbols which expresses the interdependency between cause and effect.

*5) Orthogonal Array Testing:* OAT can be applied to problems in which the input domain is relatively small, but too large to accommodate exhaustive testing.

*6) All Pair Testing:* In all pair testing technique, test cases are designs to execute all possible discrete combinations of each pair of input parameters. Its main objective is to have a set of test cases that covers all the pairs.

*7) State Transition Testing:* This type of testing is useful for testing state machine and also for navigation of graphical user interface.

## IV. GREY BOX TESTING TECHNIQUE

Grey box testing technique will increase the testing coverage by allowing us to focus on all the layers of any complex system through the combination of all existing white box and black box testing.
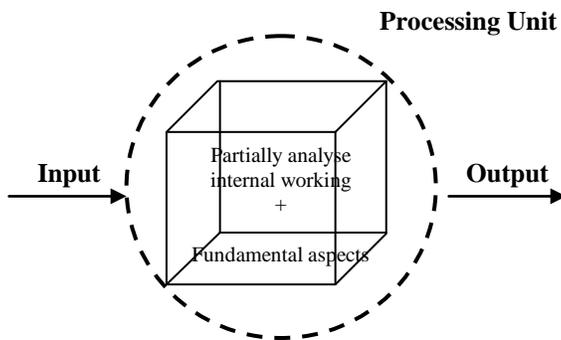


Figure 5.    Represent grey box testing

In grey box testing the tester must have knowledge of internal data structures and algorithm, for the purpose of designing test cases. Examples of grey box testing technique are: [6]

- Architectural Model
- Unified Modeling language (UML)
- State Model (Finite State Machine)

In grey box testing the codes of two modules are studied (white box testing method) for the design of test cases and actual test are performed in the interfaces exposed (black box testing method).

Some of the advantages of grey box testing technique are listed below: [4] [6]

- Grey box testing provides combined benefits of white box and black box testing techniques.
- In grey box testing, the tester relies on interface definition and functional specification rather than source code.
- In grey box testing, the tester can design excellent test scenarios.
- The test is done from the user's point of view rather than designer's point of view.
- Create an intelligent test authoring.

- Unbiased testing.

Some of the disadvantages of grey box testing technique are listed below:

- Test coverage is limited as the access to source code is not available.
- It is difficult to associate defect identification in distributed applications.
- Many program paths remain untested.
- If the software designer has already run a test case, the tests can be redundant.

The other name of grey box testing is translucent testing. Different forms of grey box testing techniques are briefly described below: [6]
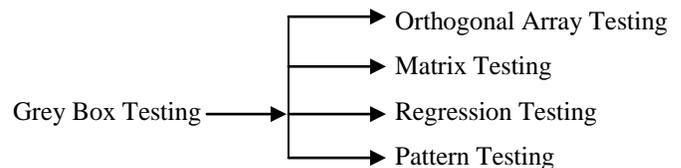


Figure 6.    Represent different forms of grey box testing techniques

*1) Orthogonal Array Testing:* This type of testing use as subset of all possible combinations.

*2) Matrix Testing:* In matrix testing the status report of the project is stated.

*3) Regression Testing:* If new changes are made in software, regression testing implies running of test cases.

*4) Pattern Testing:* Pattern testing verifies the good application for its architecture and design.

## V. FUTURE OF SOFTWARE TESTING

With the changing trends in the software industry, software testing too changes. The existing new technologies like Service Oriented Architecture (SOA), wireless technologies, mobile services etc. has opened new path to testing. Some of the changes which we will see in the industry over the next few years are listed below: [8]

- Testers will provide light weight models that developers can run against their codes.
- Early review and modeling will exposes many ambiguous bugs.
- As in the future developer's code is full of testability hooks, errors will be more detectable.
- Static analyser (detection tools) will come in main stream.
- Useful matrices such as spec coverage, model coverage and code coverage drives the projects.
- Combinatorial tools will allow testers to prioritize their testing.
- The testers will provide visible and value added services throughout the software development process.
- Tester can develop test harnesses stubs and drivers written in and interacting with a variety of programmatic languages.

- Tomorrows' tester will be professionally more educated, examine and accredited professional.

TABLE I.    COMPARISON BETWEEN THREE FORMS OF TESTING TECHNIQUES [6] [7]

| S. No. | Black Box Testing | Grey Box Testing | White Box Testing |
|---|---|---|---|
| 1. | Analyses fundamental aspects only i.e. no proved edge of internal working | Partial knowledge of internal working | Full knowledge of internal working |
| 2. | Granularity is low | Granularity is medium | Granularity is high |
| 3. | Performed by end users and also by tester and developers (user acceptance testing) | Performed by end users and also by tester and developers (user acceptance testing) | It is performed by developers and testers |
| 4. | Testing is based on external exceptions – internal behaviour of the program is ignored | Test design is based on high level database diagrams, data flow diagrams, internal states, knowledge of algorithm and architecture | Internal are fully known |
| 5. | It is least exhaustive and time consuming | It is somewhere in between | Potentially most exhaustive and time consuming |
| 6. | It can test only by trial and error method | Data domains and internal boundaries can be tested and over flow, if known | Test better: data domains and internal boundaries |
| 7. | Not suited for algorithm testing | Not suited for algorithm testing | It is suited for algorithm testing (suited for all) |

In the near future we will see a shift towards new techniques and testing transformed business operations, the way people interact with the systems and information it provides, and therefore mitigating the risk and increasing the benefits of business change.

## VI. CONCLUSION

We can define software testing as an activity aimed at evaluating an attribute, or capability of a program to determine, that it meets its required specification. Software testing can provide an independent view of the software to allow the business to appreciate and understand the risk of software implementation.

To carry out software testing in a more effective manner, in our paper we have described and compared three main software testing techniques.

## REFERENCES

[1]  Mohd. Ehmer Khan, "Different Forms of Software Testing Techniques for Finding Errors," IJCSI, Vol. 7, Issue 3, No 1, pp 11-16, May 2010

[2]  Software Testing Methodologies by Azhar available at http://azhar-paperpresentation.blogspot.com/2010/04/software-testing-methodologies.html

[3]  Mohd. Ehmer Khan, "Different Approaches to White Box testing Technique for Finding Errors," IJSEIA, Vol. 5, No. 3, pp 1-13, July 2011

[4]  Software Testing Methods available at http://www.tutorialspoint.com/software_testing/testing_methods.htm

[5]  Mohd. Ehmer Khan, "Different Approaches to Black Box Testing Technique for Finding Errors," IJSEA, Vol. 2, No. 4, pp 31-40, October 2011

[6]  Grey Box Testing from Wikipedia available at http://en.wikipedia.org/wiki/Gray_box_testing

[7]  Comparison among Black Box Testing, Gray Box Testing & White Box Testing available at http://www.softwaretestinggenius.com/articalDetails?qry=404

[8]  Predicting the Future of Testing by Harry Robinson available at http://www.stickyminds.com/sitewide.asp?Function=edetail&ObjectType=COL&ObjectId=6887

AUTHORS PROFILE

**Mohd. Ehmer Khan**

I completed my B.Sc in 1997 and M.C.A. in 2001 from Aligarh Muslim University, Aligarh, India, and pursuing Ph.D (Computer Science) from Singhania University, Jhunjhunu, India. I have worked as a lecturer at Aligarh College Engineering & Management, Aligarh, India from 1999 to 2003. From 2003 to 2005 worked as a lecturer at Institute of Foreign Trade & Management, Moradabad, India. From 2006 to present working as a lecturer in the Department of Information Technology, Al Musanna College of Technology, Ministry of Manpower, Sultanate of Oman. I am recipient of PG Merit Scholarship in MCA. My research area is software engineering with special interest in driving and monitoring program executions to find bugs, using various software testing techniques.

**Farmeena Khan**

I completed my BCA in 2003 from Dr. B. R. Ambedkar University, Agra, India and MBA in 2011 from Janardan Rai Nagar Rajasthan Vidyapeeth University, Rajasthan, India, and pursuing M.Sc (Computer Science) from EIILM University, Jorethang, Sikkim, India.