Review of Literature

In order to meet the objectives of the research various research papers are studied. The papers were focused on the different approaches to gather evidences and to find feasible testing strategies for a ETL testing. In the course of evaluating testing strategies empirical investigation of test cases are also carried out. The research papers referred lead path for fulfilling the objectives of this research.

Tore Dybå et al.[12] suggest that EBSE aims to improve decision making related to software development and maintenance by incorporating current best evidence from research with practical experience and human values. EBSE provides mechanisms to support various parts of Software Process Improvement. In particular, EBSE focuses on finding and appraising an appropriate technology for its suitability in a particular situation. There are several information sources that can be used. For example, getting viewpoints from the customers or the software’s users, asking a colleague or an expert, using own experience, or search for research-based evidence. In research-based evidence reports, articles, and other document that describe a study conducted and reported according to certain guidelines are included. The researchers here have used only the textual material available which does not prove to be sufficient enough.

Data sources like the Internet, Open forums and electronic databases should also be used for gathering data and seeking evidences. A large amount of data can be collected through these sources and it will then be classified under various heads and then will it be used for inference drawing.

Sackett et al.[13] recommend that individual doctors review the way in which they practice and teach Evidence Based Medicine in order to improve their individual performance. For EBSE, this would involve propagating successful technologies throughout a company and preventing the spread of technologies that are unsuccessful. In this paper adequate research was done in the field of medicine and the author suggests the same practices to be carried out in Software Engineering where engineers should first evaluate his decision based upon the evidences gathered and then bring it into practice. Adoption of new technology before the commencement of a new project should be based on the evidences gathered.

Though it is feasible in the field of medical science since individual doctors are responsible for reporting unanticipated side-effects of drugs, but when compared to software engineering in a competitive industry, barely any company will assist their competitors by reporting good and bad experiences with new technologies. Thus evidences should be gathered using the available literature and data gathered through primary and secondary data collection tools.

Juristo et al [14] paper on Unit Testing Techniques states that testing is an invariable part of all systems engineering project. The human efforts that are spent on testing sometimes even exceed half the project total cost. Inadequate testing has been the source of many systems failures. Thus if test cases are designed keeping in view the real world practices then analysis of these can produce useful insights. Apart from mitigating defects in programs, testing is performed in peripheral areas such as performance of software, reliability and security. Testing does not simply focuses on the technology
alone but on socio-technical issues such as acceptability, usability and fitness of software. It focuses on the whole systems rather than software in isolation. The authors did not attempt to create a new definition of testing in this paper, but rather to navigate this fuzzy concept by looking at what testing is and how it is meaningful from a practitioner point of view. For this study, the authors chose to use the definitions provided in the IEEE’s Software Engineering Body of Knowledge—usually referred to as the Swebok. Juristo et al. refined and extended the Swebok classification system to include more detailed categories for code coverage and test set classification. The source of evidence however was confined to papers published in the IEEE and ACM electronic databases.

Wasif Afzal et al [15] use Search-based software engineering (SBSE) as a tool for the application of optimization techniques in solving software engineering problems. This paper discussed the applicability of optimization techniques in solving software engineering problems. The growing interest in evidence based or search based search technique can be attributed to the fact that generation of software tests is generally considered as an undeniable problem, since there are many possible combinations of a program’s input. The researchers here performed a SLR and came to the conclusion that metaheuristic search techniques have been applied for non-functional testing of execution time, quality of service, security, usability and safety. A variety of metaheuristic search techniques were found to be applicable for non-functional testing including simulated annealing, tabu search, genetic algorithms, ant colony methods, grammatical evolution, genetic programming and swarm intelligence methods.

Martin Johansen et al [16] suggested that evidence gathering is most suitable for software products which cater to similar type of software requirements. These type of softwares share a considerable amount of code and testing products individually is redundant for product lines. Since the products in a product line share a large amount of code, it should be possible to improve testing by utilizing the specification of the similarities and differences between the products. A strategy for testing a software product line is a description of which steps a test engineering should follow to reduce the effort on testing the product line significantly below the effort required to test each product delivered to a customer.

Sarvnaz Karimi and Falk Scholer [17] credit systematic review as the key tool used in evidence based policy. The SLR synthesize available research on the topic of investigation. Though many researches hold their base by searching the literature but a SLR is carried out to agreed standards: using clear protocols in carrying out the process, focusing on specific questions, identifying as much of the relevant literature as possible, critically appraising the quality of the research included in the review; synthesizing research findings from included studies; being as objective as possible to remove bias, and, updating the review so that it remains relevant.
Mostafa Kandil[18] et al draw a line between testing done in software and testing a web application. The functional requirements of a web application are different thus separate testing strategies should be followed to give a proper coverage to WA’s functional requirements. Due to the heterogeneous nature and different quality criteria of Web Applications, its components and user expectations, new demands emerge for testing of those systems to ensure a high reliability level. The most important aspect of WA’s is that it deals with large numbers of users, clients and stakeholders around the world. Many WA’s require high quality security checks such as banking systems, governmental, ticketing system, e-commerce etc. The researchers here only contribute in WA’s and GUI applications, which can be scaled for other aspects of web applications as well like data flow, user acceptability and so on.

Experiments carried out by Ciupa [19] et al show that the relative number of faults which are detected through random testing is certain that is their numbers can be predicted, but different runs of the random test case generator detect different faults. The experiments carried out suggest that to find faults quickly random testing is supposedly the best suitable technique. The nature of faults was also evaluated here and different strategies were applied in the data sets. The researchers reached to the conclusion that number of faults can be predicted and they also vary from one technique to another but the nature of faults detected are more or less the same in all techniques.

To reduce the cost of ETL testing it is advisable to automate the testing process as much as possible, and automating test case generation is an important phase of this automation. A possible strategy in the automation of test case generation is the application of met heuristic search (MHS) algorithms.

Considerable amount of research has been carried out in MHS algorithms and there is still a large potential in this area as search based testing proves to be a cost effective and workable option in testing field [20].

To get more test cases a mapping study can be done by the researchers and they will have to systematically classify the research papers and from them conclusions can be drawn. Evidence based datasets are also generated which can also be referred.

There is a severe decoupling between research in the computing field and the state of the practice in the field. The gap between empirical ETL testing strategies and software testing practice might be lessened if more attention were paid to two important aspects of evidence. The first is that evidence from case or field studies of actual software testing practice are
essential in order to understand and inform that practice. The second is that the nature of evidence should fit the purpose to which the evidence is going to be put.

Thus a study of all the mentioned papers indicates that there is a lot of potential in the field of ETL testing using the EBSE approach. The research focuses on the use of evidence based approach for testing which includes test case generation and also determination of the best approach for testing using various techniques. The main intention of this research proposal is primarily on testing strategies.

Motivation

After extensive literature search, evidence based software engineering approach appears to be a viable alternative for many ETL testing practices. It is a very current and innovative approach which is being followed and is able to eliminate some of the defects that the ETL testers are facing.

Usually research is problem solving agents. They solve the problems of variety of stakeholders like farmers, human, citizens, livestock, entrepreneurs, etc.. It is always intended towards solving a particular problem or exploring something which is not known. So, real benefits and impacts are anticipated before starting of the project.

True benefit or impact can be measured for individual output which can be cumulatively measured for the institute, then cumulatively for the organization and ultimately for the country.

My research currently doesn’t have a huge benefit practically to society, however advancing knowledge is generally beneficial. Within the next couple of years my research should show more benefit to society as when the technology gets implemented in organization this will ensure less money is wasted.

I’m hoping that by the end of my PhD there will be a working prototype that I can test out somewhere.

Research indicates that testing being a cumbersome as well as integral part of software development when combined with EBSE leads to more optimum results.

Plan of work and Methodology

By reading relevant research papers it is being concluded that for evidence based approach the most feasible approach is systematic literature review, mapping studies and evidence gathering as they provide the researcher an unbiased review. In due course of time Narrative synthesis and Meta analysis tools will also be explored and the optimum evidence gathering tool will be selected for further collection.

The process of Systematic Literature Review is widely used to aggregate the results of primary studies. Conducting such a review involves an exhaustive search of the relevant literature, selection and data extraction. All of this is conducted in accordance with a review protocol, which is composed ahead of performing the study and specifies in detail how the study is to be performed and the reasoning behind any choices.
**Work Plan**

The research plan is divided into following seven phases:

<table>
<thead>
<tr>
<th>Phase</th>
<th>Activity</th>
<th>Approx time (in months)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Literature survey on evidence based software engineering. And its classification</td>
<td>8-12</td>
</tr>
<tr>
<td>2.</td>
<td>Gathering primary data of software projects and Searching empirical databases</td>
<td>4-5</td>
</tr>
<tr>
<td>3.</td>
<td>Refining and implementing search strategies for system properties</td>
<td>3-5</td>
</tr>
<tr>
<td>4.</td>
<td>Assessment of drawn inferences from the proposed study</td>
<td>3-4</td>
</tr>
<tr>
<td>5.</td>
<td>Develop new / refine existing methods for improvement in testing strategy.</td>
<td>2-3</td>
</tr>
<tr>
<td>6.</td>
<td>Implementation, comparison with existing techniques and drawing conclusions</td>
<td>4-5</td>
</tr>
<tr>
<td>7.</td>
<td>Paper/ Thesis Writing</td>
<td>3-5</td>
</tr>
</tbody>
</table>

**Schedule of the activity:**

- Survey of work
- Design and develop methodologies and publish research papers
- A comparative study of testing methods and automated testing tools
- Submission of thesis