Faculty of Manufacturing Engineering

DEVELOPMENT OF TEST DATA NAVIGATOR TOOL
(TDN)

Usha Nantini A/P Perumal

Master in Manufacturing Engineering
(Manufacturing System Engineering)

2017
DECLARATION

I declare that this thesis entitle “Development of Test Data Navigator Tool (TDN)” is the result of my own research except as cited in the reference. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ..................................
Name : ..................................
Date : ..................................
APPROVAL

I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality as a partial fulfilment of Master of Manufacturing Engineering (Manufacturing Systems).

Signature : ....................................
Supervisor Name : ....................................
Date : .............................................
DEDICATION

To my beloved parents Mr. Perumal Doraisamy and Mrs. Bathmawathy Mutiah
ABSTRACT

This project, titled "Development Test Data Navigator Tool (TDN)", aims to employ a solid tool strategy, which can deliver time savings and, eventually, cost savings by almost getting rid of the manual activity involved in determining, transforming, setting up, and maintaining test data sets. The Test Data Navigator Tool (TDN) is a mechanism for tracking the records of Standard Test Data Format (STDF) log files as well as CAMSTAR test summaries, and determining the aberrations of the records and the availability of data. The TDN is formulated using Microsoft Visual Studio and the Visual Basic (vb.net) programming language. The system essentially has two kinds of modules which aid in mining a test data location by means of the file transfer process logging: search record module, and export and save module. The TDN tool emphasises on addressing two critical issues in case involvement of additional resources and the average time for resolving a missing test data or summary problem. Through this mechanism, the procedure of tracking the record of a single lot or multiple lots becomes quite easier, facilitated by the click of a mouse. This tool offers user-friendly data entry with features such as a list box, checkbox and dropdown button menu for making the entry of inputs simpler to comprehend and use. The project has three objectives: (i) To study and scrutinise the gaps in the present data searching approach utilising time study; (ii) To introduce and formulate a fresh framework for enhancing the present data searching approach; (iii) To carry out a usability study and substantiate the formulated framework. For accomplishing the first objective, time study and brainstorming have been employed. A feasibility study was then carried out pertaining to TDN development concepts such as project theory, fact-finding, and old system study analysis as well as literature review to formulate a fresh framework. Lastly, a tool assessment was carried out after the framework was completed.
ACKNOWLEDGEMENTS

I would like to express my profound gratitude to my parents Mr Perumal s/o Doraisamy and Mrs Batmawathi d/o Mutiah for giving me consistent support and continuous encouragement throughout my years of study and in the process of researching and writing this thesis. This achievement would have been impossible without them.

I would like to thank my thesis advisor Dr. Effendi bin Mohamad of the Faculty of Manufacturing Engineering at University Teknikal Malaysia Melaka. Mr. Prof. Effendi was always available whenever I faced problems or had a question regarding my research or writing. He ensured that this paper remained my own work, but pointed me in the right direction whenever he thought I needed it. I am indebted to him for his extremely valuable comments on this thesis.

I would finally like to thank the experts involved in the validation survey for this research project. Without their passionate participation and input, the validation survey could not have been conducted successfully.

Thank you
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECLARATION</td>
<td></td>
</tr>
<tr>
<td>APPROVAL</td>
<td></td>
</tr>
<tr>
<td>DEDICATION</td>
<td></td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>i</td>
</tr>
<tr>
<td>ABSTRAK</td>
<td>ii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>iii</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>vi</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>viii</td>
</tr>
<tr>
<td>LIST OF APPENDICES</td>
<td>xii</td>
</tr>
</tbody>
</table>

## CHAPTER

### 1. INTRODUCTION

1.1 Background of the study
1.2 Problem Statement
1.3 Objective of the study
1.4 Scope of the study
1.5 Significant of the study
1.6 Study Period (Project Schedule)
1.7 Summary

### 2. LITERATURE REVIEW

2.1 Introduction
2.2 Overview of the Semiconductor Manufacturing Process
2.2.1 Review on Backend of Line Testing Process
2.3 Problems with the Data
2.4 Important of Proper Tools in Test Data Analysis
2.5 An overview of Software Development Process Models
2.5.1 Waterfall Model
2.5.2 Iterative and incremental Model
2.5.3 Spiral Model
2.5.4 A spiral development model
2.5.5 Prototyping Model
2.6 Summary

### 3. METHODOLOGY

3.1 Requirements development
3.1.1 Requirements gathering methodology
3.2 System and software design
3.2.1 Project Facilities Requirement
3.2.2 Software Requirement
3.2.3 Hardware Requirement
3.3 TDN conceptual model
3.4 The Proposed Architectural description
3.4.1 Design of the Presentation Layer

iv
4. RESULT AND DISCUSSION

4.1 To study and analysis the gap in current data searching method using time study
   4.1.1 Validation of result (Manual searching method vs TDN tool)
   4.1.2 Discussion

4.2 To introduce and develop a new framework to improve the current data searching method.
   4.2.1 User Satisfaction with the implementation
   4.2.2 System speed
   4.2.3 On-Job-Training-Instruction - OJTI

4.3 To conduct a usability study and validate on the develop framework
   4.3.1 Descriptive results
   4.3.2 Regression Analysis
     4.3.2.1 System quality
     4.3.2.2 User satisfaction
     4.3.2.3 Intention to Use
     4.3.2.4 Perceived Net Benefit

4.4 Validating the TDN
   4.4.1 Reliability
     4.4.1.1 Internal Consistence
     4.4.1.2 Test-Retest Reliability
   4.4.2 Content Validity

4.5 Summary

5. CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

REFERENCES

APPENDICES
## LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Advantage and disadvantage of waterfall life cycle model</td>
<td>23</td>
</tr>
<tr>
<td>2.2</td>
<td>Advantage and disadvantage of iterative and incremental model</td>
<td>26</td>
</tr>
<tr>
<td>2.3</td>
<td>Advantage and disadvantage of spiral development model</td>
<td>28</td>
</tr>
<tr>
<td>2.4</td>
<td>Advantage and disadvantage V life cycle model</td>
<td>31</td>
</tr>
<tr>
<td>2.5</td>
<td>Advantage and disadvantage V life cycle model</td>
<td>34</td>
</tr>
<tr>
<td>3.1</td>
<td>Requirement Engineering Process definition</td>
<td>36</td>
</tr>
<tr>
<td>3.2</td>
<td>Requirement gathering process detail explanations</td>
<td>41</td>
</tr>
<tr>
<td>3.3</td>
<td>Project Facility Specifications</td>
<td>43</td>
</tr>
<tr>
<td>3.4</td>
<td>Tester Platform</td>
<td>44</td>
</tr>
<tr>
<td>3.5</td>
<td>Software Requirement and Function</td>
<td>44</td>
</tr>
<tr>
<td>3.6</td>
<td>Language Used and Function</td>
<td>45</td>
</tr>
<tr>
<td>3.7</td>
<td>Hardware Requirement</td>
<td>45</td>
</tr>
<tr>
<td>3.8</td>
<td>Hardware used and function</td>
<td>45</td>
</tr>
<tr>
<td>3.9</td>
<td>Conceptual model of the TDN tool</td>
<td>46</td>
</tr>
<tr>
<td>3.10</td>
<td>Black box testing list</td>
<td>53</td>
</tr>
<tr>
<td>3.11</td>
<td>TDN functional testing</td>
<td>55</td>
</tr>
<tr>
<td>3.12</td>
<td>Summary of Test Data Navigator Tool</td>
<td>57</td>
</tr>
<tr>
<td>4.1</td>
<td>Pre and post implementation result</td>
<td>90</td>
</tr>
<tr>
<td>4.2</td>
<td>Average age of end-users</td>
<td>92</td>
</tr>
</tbody>
</table>
4.3 Gender, segment, role and separation distribution for end-users 93
4.4 Scale properties 94
4.5 The test-retest reliability 99
4.6 Correlations between end-user satisfaction and end-user participation / perceived system success 100
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>ABC’s Tester Data Flow Diagram</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Current Framework Search Data</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>Test Data Availability Trend</td>
<td>5</td>
</tr>
<tr>
<td>1.4</td>
<td>Test Data Incident Ticket Trend</td>
<td>5</td>
</tr>
<tr>
<td>2.1</td>
<td>Semiconductor Manufacturing Processes</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>Graphical presentation of production flow</td>
<td>11</td>
</tr>
<tr>
<td>2.3</td>
<td>Software process</td>
<td>19</td>
</tr>
<tr>
<td>2.4</td>
<td>Software development life cycle</td>
<td>21</td>
</tr>
<tr>
<td>2.5</td>
<td>An example of the waterfall life cycle model</td>
<td>22</td>
</tr>
<tr>
<td>2.6</td>
<td>An example of the incremental life cycle model</td>
<td>24</td>
</tr>
<tr>
<td>2.7</td>
<td>Iterative and incremental software model</td>
<td>25</td>
</tr>
<tr>
<td>2.8</td>
<td>A spiral development model</td>
<td>27</td>
</tr>
<tr>
<td>2.9</td>
<td>A spiral development model</td>
<td>28</td>
</tr>
<tr>
<td>2.10</td>
<td>V life cycle model</td>
<td>30</td>
</tr>
<tr>
<td>2.11</td>
<td>Throwaway Prototyping Methodology</td>
<td>32</td>
</tr>
<tr>
<td>2.12</td>
<td>Evolutionary Prototyping Methodology</td>
<td>33</td>
</tr>
<tr>
<td>2.13</td>
<td>Incremental prototyping Methodology</td>
<td>33</td>
</tr>
<tr>
<td>3.1</td>
<td>Test Data Navigator new system</td>
<td>36</td>
</tr>
<tr>
<td>3.2</td>
<td>TDN system searching flow</td>
<td>37</td>
</tr>
</tbody>
</table>
3.3 TDN design and implementation stages 38
3.4 Requirement Engineering Process 39
3.5 Requirement gathering process flow for conducting the project 41
3.6 Typical Phase of Design 43
3.7 Conceptual model of the TDN tool 46
3.8 Interfacing between layers 47
3.9 The three-Layered architecture for the TDN tool 48

Design of the Presentation Layer
3.10 Software Testing Methodology 49
3.11 Web page programming options 52
3.12 Test Data Navigator Tool Main Interface 60
3.13 New system searching method 61
3.14 Target Framework 62
3.15 Project property group 63
3.16 Target Framework 64
3.17 Target Framework 64
3.18 Application keyword configuration 65
3.19 Search And Check Lot ID Class 66
3.20 Help Class 67
3.21 Interface and Coding 68
3.22 Build Exe File 69
3.23 Test Data Navigator Interface 70
3.24 Interface 71
3.25 Specification of TDN Part B 71
3.26 Specification of TDN Part C 72
3.27 Specification of Data Management Searching Tool's Part C
3.28 Example of Abnormalities of STDF Log’s Record
3.29 Example of Exported Results in Microsoft Excel
3.30 Target Audience
3.31 Organizational Structure
4.1 Time improvement by per lot
4.2 Time improvement trend before and after implementation
4.3 Data missing monthly trend
4.4 Time taken for investigate root cause (Hours/Week)
4.5 KPI data loading speed BEAR site with SB SBA before TDN
4.6 KPI data loading speed BEAR site with SB SBA after TDN implementation
4.7 DDM Trend
4.8 TDN implementation evaluation model
4.9 Test Data Navigator Main Interface
4.10 Specification of Test Data Navigator Tool’s Part A
4.11 Specification of Test Data Navigator Tool’s Part B
4.12 Specification of Test Data Navigator Tool’s Part C
4.13 Specification of Test Data Navigator Tool’s Part D
4.14 Survey process used
4.15 Respondents working experience
4.16 Model I Multiple Regression Analysis. Dependent variable: TDN System
4.17 Model II Multiple Regression Analysis. Dependent variable: User Satisfaction
4.18 Model II Multiple Regression Analysis. Dependent variable: Intention to Use

4.19 Model III Multiple Regression Analysis. Dependent variable: Perceived Net Benefit

4.20 Model I Multiple Regression Analysis. Dependent variable: TDN System
# LIST OF APPENDICES

<table>
<thead>
<tr>
<th>APPENDIX</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Gantt chart for MP1 and MP2</td>
<td>108</td>
</tr>
<tr>
<td>B</td>
<td>Survey of TDN performance evaluation (Part 1)</td>
<td>109</td>
</tr>
<tr>
<td>B2</td>
<td>Survey of TDN performance evaluation (Part 2)</td>
<td>110</td>
</tr>
<tr>
<td>C</td>
<td>Target framework and property configuration</td>
<td>111</td>
</tr>
<tr>
<td>C2</td>
<td>System reference manager configuration</td>
<td>112</td>
</tr>
<tr>
<td>C3</td>
<td>App.configuration</td>
<td>113</td>
</tr>
<tr>
<td>C4</td>
<td>Search and check lot id class configuration</td>
<td>114</td>
</tr>
<tr>
<td>C5</td>
<td>Helper class configuration</td>
<td>115</td>
</tr>
<tr>
<td>C6</td>
<td>Search class configuration</td>
<td>116</td>
</tr>
<tr>
<td>C7</td>
<td>EXE file configuration</td>
<td>117</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

The data of a company is considered to be the most treasured and valuable asset of the company (Silwattananusarn et al., 2012). This holds true for most companies. So, for any company, the use of unstructured content or poor quality data will result in an insurmountable challenge or a progressive disaster, which resounds the age-old technology catchword “garbage in, garbage out”. This is very important for all data integration schemes and initiatives (Caelli et al., 2013). There is a profound necessity to analyse and cleanse all the data before it can be used in any database for successful application (Vishnu et al., 2014). In general, to assure that the data retains its high quality over time, it is essential that organisations emphasise on the data cleansing procedures, and ensure validation and tracking of data quality levels over time (Richard et al., 1995). Else, the poor quality of data will inescapably increase the risk levels that would increase costs and reduce profitability, halt relations in the supply chain, and result in inferior customer relationship management (Manjunath et al., 2012). In automated data validation, the data accuracy is derived by comparing it against benchmark values and time, thereby improving the data quality in less time and with reduced cost (Rathika et al., 2014).

The quality of data is characterised by quality dimensions such as timeliness, integrity, accuracy, validity, accessibility, and completeness (Manjunath et al., 2012). The availability of the expected data is referred to as the completeness of data. The consistency of data without any conflict information is known as data synchronisation. Data validity
refers to the correctness and reasonableness of data. This study stresses on suggesting a framework that is capable of automatic data quality checks. By enabling companies to keep a track of the data quality, it will be possible for companies to instantly identify and amend the potential problems before the data quality significantly declines.

The objective of this study was to implement a Test Data Navigator Tool (TDN) that substitutes the manual searching procedure, and thereby reduces the test data and test summary availability status confirmation’s period. The new system is designed to track the records of both CAMSTAR test summaries and Standard Test Data Format (STDF) log files. Therefore, it has the potential to identify abnormalities of the records, and still ascertain the data availability. For developing the TDN software, various types of software development models have been reviewed, out of which the software development life cycle approach has been selected. This model endeavours to deliver a set of useful guidelines for the formulation of new programs or systems. The Microsoft Visual Studio and Visual Basic (vb.net) programming language was used to develop the TDN tool.

1.1 Background of the study

When a product is sent for testing, before the testing starts, the tester will key in the information of testing such as test type, tester ID and test mode into the recording system. This set of information is known as the Start lot. On completion of the testing, the tester will provide several files such as the STDF log, and information and summary files, known as the End lot. These files will be saved in a local hard drive in the testing environment.
The test results will be printed as a hardcopy summary. In the meantime, a soft copy of the collective test summary will be sent to the Camstar’s folder for future reference. The Camstar Enterprise Platform refers to a global-ready, growth-ready enterprise manufacturing execution system (MES) that enables visibility, control, and continuous improvement. Moreover, a copy of test results summary will be sent to a filer on the testing side so that it can be used by the Cerberus system for analysis of data. There are different departments that are involved in this end-to-end test data management. This chapter constitutes of the background of the study, the problem statement, the scope and objectives of the study, and the background of Company Infineon Sdn. Bhd. The significance and the estimated findings of this study are also presented at the end of this chapter.
1.2 Problem Statement

For ABC company, test data integrity is one of the key components of KPI measurement. On the basis of performance of different testers with respect to data availability and stability, a weekly data integrity report is produced manually. At present, different folders have to be searched to compute the records of log files and test summaries manually in order to verify the status of data availability and the root cause of missing data.

To confirm the status of data availability, only one Lot ID and tester type is provided to the data management team, using which they need to identify the tester station, tester mode, and generated test files’ names to proceed to the next step. Once it is confirmed that the generated files are missing, the Factory Integration (FI)’s department and EBS team have to be contacted to understand the root cause of the missing data and thus find a solution to resolve the problem.
The confirmation of the data availability status could take up to five minutes for a single case before any further action is taken. At times, it could even result in a few weeks of investigation and analysis before the problem is resolved.

Figure 1.3: Data missing Trend

The data missing trend and the problem identified after a long investigation is depicted in Figure 1.3. Due to late identification of the problem, the number of missing data increase. The test data incident ticket trend is presented below in Figure 1.4. The majority of the tickets failed or closed as a result of inappropriate investigation.

Figure 1.4: Test Data Incident Ticket Trend
Therefore, a new approach has been suggested in order to decrease the time required for the status confirmation to substitute the manual search process with a test data navigator tool. With the application of the search tool in the present environment, it is estimated that the time needed to find the data availability status will reduce and therefore the unnecessary steps included for data comparison will also be reduced.

1.3 Objective of the study

The objectives of the navigator tool that is being designed include:

i. To study and evaluate the gap in the present data search methodology with the help of time study

ii. To design and develop a new framework that improves the present data search methodology

iii. To conduct a usability study and validate the developed framework

1.4 Scope of the study

The analysis will be dependent on the following factors:

i. A balance of industry depth and low-cost test data management resources, which lead to reduced costs for addressing and managing test data requirements

ii. High quality and safe test data for the required target environments in a timely manner

iii. An adaptable delivery model for providing test data management services

iv. Minimised risks related to data integrity, data security and human error
1.5 Significance of the study

With the use of this system, the engineers and operators can ascertain the status of data availability through a convenient approach, in which the search time is considerably decreased. The reduction in search time is due to the existence of multiple Lot ID search and availability of time range setting, where the search records can be used for immediate and future reference.

The STDF search mode has the potential to search the contents of STDF log to find the records of each Lot ID and determine the abnormalities present in the record. With this approach, where the abnormalities detection function is used, the engineers can easily identify a problem in the system. This will reduce the time required to detect an abnormality and thereby resolve the problem.

Moreover, this tool also enables the elimination of non-value added activities in case the test data goes missing. The present practices in production consist of manual search in multiple tester filer log files or creation of incident ticket to IT department to validate data availability.

i. The manual search for single Lot ID requires up to 5 minutes.

ii. Substantial time and effort is required to handle search records of total weekly cases.

iii. It is necessary to call the help desk to create incident ticket to IT department to ascertain the availability of data.

iv. On the basis of ticket priority, IT takes 1–7 days to resolve a ticket.

Therefore, the outcome of the study will be shared with ABC management and it is expected that the new found information is likely to benefit the company and its employees.
1.6 Study Period (Project Schedule)

All the tasks are performed as per the monthly schedule phases. Every week, different sets of tasks are performed in different phases. The Gantt chart that represents the tasks schedule is depicted in the Appendices at the back of this report.

1.7 Summary

To summarise, this chapter presents a complete review, analysis and feasibility study of the Test Data Navigator Tool (TDN). It covers the objectives of project implementation, the scope of the project, the importance of the study, and the project schedule. The approach adopted for this project has been reviewed and the existing knowledge has been synthesized, and a theoretical analysis of the methodology is carried out through research and from a series of observations. The observation approach allows the identification of systemic deficiencies and security issues that are not found in the formal audit process. The research approach can be reviewed based on books, journals, experts’ opinion and Internet resources.