TROUBLESHOOTING PROCESS ANALYSIS AND DEVELOPMENT OF APPLICATION FOR DECISION MAKING ENHANCEMENT

Kashif Mahmood^a, Eduard Shevtshenko^a, Tatjana Karaulova^a,
Eva Branten^a, Meysam Maleki^b

^aTallinn University of Technology, Ehitajate tee 5, Tallinn 19086, Estonia

^bUNIDEMI, Universidade Nova de Lisboa 2829-516 Caparica, Portugal

Abstract

Troubleshooting process is often used to diagnose the cause of failed products in the production systems. This method is employed to repair rejected products and to find root causes of problems so that failed products can be usable again. Troubleshooting is one of the most important ways of identifying symptoms of specific problem in a product and also helps to provide remedial action regarding the causes of those symptoms. This research aims to analyze a troubleshooting process and provide enhancement in decision making procedure of troubleshooters which leads to increase the overall productivity of production process. A framework for business process improvement and data cleaner application are developed for this purpose that facilitates to improve the quality of data, quality of decision making process of troubleshooters and reduce the waste of materials. Moreover, AS-IS analysis of troubleshooting process is made based on that analysis TO-BE process flow is also described while a data cleaner application is developed to support the TO-BE process. The outcome of this research is mainly beneficial for manufacturing industry, however further studies may expand the findings to other streams such as service industry.

Keywords: Troubleshooting process; business process analysis; decision making; process improvement; troubleshooter application

This Publication has to be referred as: Mahmood, K[ashif]; Shevtshenko, E[duard]; Karaulova, T[atjana]; Branten, E[va] & Maleki, M[eyсам] (2016). Troubleshooting Process Analysis and Development of Application for Decision Making Enhancement, Proceedings of the 26th DAAAM International Symposium, pp.0663-0671, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-07-5, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/26th.daaam.proceedings.090

1. Introduction

In this section the background of the research will be discussed, followed by a problem statement that includes why it is necessary to analyse the troubleshooting process and what are financial benefits of that particular process improvement. Furthermore, the importance of testing and troubleshooting of a product will be a part of this section, moreover, the objectives of authors' applied research will also be highlighted at the end of the section.

1.1. Background

In the scope of production process the ultimate goal is to improve the process in order to achieve the highest production yield. For the sake of continuous improvement in production processes and product quality, companies try to reach less scarp, less rework, less consumption of surplus resources, time and money that leads to reliable and speed up a production process [1]. Although machines and equipment have specific capacity rates that cannot be exceeded, however it is a possible to speed up the work performed by humans. On the other hand optimization of material handling and storage in a production facility needs effective organization of the whole technology route depending on the manufacturing and lead times, production capacity and market requirements [2].

In this research we focus on the improvement of troubleshooting process of an electronics product manufacturer. Troubleshooting is the process to correct faults in an electronic system, it also analyse and solve problems in a tested product that is rejected after testing [3]. This research uses process mapping and business process management approach to analyse and enhance the decision making of troubleshooting process.

In order to increase the amount of product diagnosed by troubleshooters and fixed by repairmen, troubleshooters are using the application provided by an external company to see the error code of a given unit and to log the corrective action that were taken. As application is connected to a troubleshooting database, it shows and stores the corrective actions of the same error that were taken to eliminate the problem. Basically, corrective actions are the only data type inserted to database (through the application) by a troubleshooter, while the rest of data comes from the automated test stations. This makes the data not fully reliable due to the human error possibility.

1.2. Problem statement

On the production level where troubleshooter has to deal with the damaged product, there is a need to support the reliable data, so that the problem can be solved in a shorter time and the material waste would be minimal. Currently, troubleshooter is not able to monitor the accurate data displayed by troubleshooting application. Some proposed corrective actions are incorrect and are stored inaccurately in the troubleshooting process. The current solution that is in use does not has validating option for the input data and stored all the information provided by troubleshooter as it is. Inaccuracy causes additional expenses and time delays, which has a harmful effect on production outcome. It means there is a certain need for a solution that performs data filtering in order to define the most efficient corrective actions for a particular faults.

The proposed solution has to be helpful for a troubleshooting process and act as a supportive tool that facilitates finding the most efficient action, which needs to be performed in order to fix the malfunctioned unit. A business process improvement framework would be used to find out an appropriate solution.

There is missing a relationship between different failure patterns produced by test system and faults of the component in the products. It includes (but not limited) following research questions:

- How to create faults classifier and store the product failure, repair data (information sources, activities performed, performers)?
- How the structure of current information sources should be build (information sources, activities performed)?
- How statistical tools can be employed to analysis test results?

In this study the qualitative case study research methodology is used in order answer the research questions [4]. Information is collected by on site (production floor) observations and through interviews with testing and troubleshooting operators, technical staff (maintenance specialist) and production managers.

1.3. Product testing importance

There are number of reasons why product and component testing worth investigating, it includes [5, 6]:

- Quality and reliability – Testing makes it possible to predict the future failures in the field and ensures the present product in factory as fully functional unit.
- Customer Satisfaction – While a reliable product is not sufficient to considerably affect customer satisfaction in a positive manner, a problematic or non-working product will definitely leave a negative effect on customer satisfaction. Therefore, a high quality level is mandatory for customer satisfaction and manufacturing company reputation.
- Warranty Costs – If a product fails to perform as it should within the warranty period, the replacement or repair are very high and definitely will negatively affect profits, as well as gain a negative attention from customer.

- **Business Gain** – The way of working towards improved quality and functionality by continuous testing efficiency improvement shows to customers that the manufacturer is serious about its product and committed to customer satisfaction. This attitude has a positive impact on current and future customer business.

This research is done on one of the most difficult high runner product from testing point of view that has the lowest yield among the similar product family. There might be several root causes that can be under consideration as the physical repair procedure was completely correct. The most common of them are:

- Repair decision, e.g. component exchange caused another failure to appear;
- Repair action or failure definition was wrongly determined by a trouble-shooter;
- A test software or hardware instability or quality issue with component exchanged.

1.4. Objective of applied research

This study analysed the current AS-IS troubleshooting process and process related data. Based on analysis result, a framework for business process improvement is developed and a new process flow TO-BE is proposed that will help to improve the quality of used data, improve the productivity and quality of decision making process of troubleshooters and reduce the waste of materials, resulted from incorrect corrective actions data provided by troubleshooters application. The main task was to develop a solution that supports new process flow. For this purpose a relationships between failures and corrective actions based on statistical analysis of test results was defined, to develop the decision making mechanism for repair database (DB) that enables to define the most efficient corrective actions. Main activities of business processes with responsible personnel for elaboration of improved troubleshooting process are introduced in a framework which can be seen Fig.1.

Fig.1. Main business process activities for elaboration of improved solution

The main result of the research was speeding up and enhancing the decision making of troubleshooting process known as improved solution. In order to achieve this goal, business process improvement approach was adapted, it is presented in fig.1 in the form of a value added chain. It starts with business process analysis followed by business process improvement then development of IT application (data cleaner) and proposed business process implementation. Process and IT engineers were the responsible to carries out this approach to get an improved solution.

2. Business process analysis

Nowadays management of an enterprise deems the root cause of most of the problems can be found in the process [7]. Business processes of a company describe its operations and identify how the company deliver products or services moreover, business processes facilitate firms to react towards change more quickly [8]. In today's world companies hold the idea of "process driven company" as they desire to develop, extent, deliver consistently and become less dependent on individuals, processes represent the way to reach there [9]. Moreover, the triangle of "Quality, Cost and Time" is not only applicable on product but also on processes, thus it is possible to manufacture customer oriented products at low cost and fast only by capable efficient processes [10]. This section consists of process mapping of one of the product that was selected for process analysis. The main focus will be on the testing and troubleshooting process which was the key and an important area of improvement during the case study. Section starts with a brief description of process mapping and process flow of existing activities from sub assembly to final assembly and inspection, data collection points, formats and sources of data will also be identified in the process flow of product. Followed by AS-IS process model of testing and troubleshooting, furthermore statistical analysis of testing data will be described at the end of this section.

2.1 Process mapping

A process is "a series of actions or steps taken in order to achieve a particular end" [11], but most important: a process is an input-output system. Where there is no input, there is no output, meaning all steps performed will not produce the desired product, if the right raw-material (components) feeding is not assured. Also it is necessary to define the process

at the start of analysis for better perception and awareness; process mapping is a tool that facilitates in this direction. Process maps are also used to record various types of supplementary information that are relevant to the project. It can be done on a very high level; even there are no limits for going deeper into sub-processes of processes [12]. Moreover, even the initial step of risk assessment of a system like a machine tool starts with mapping of manufacturing business process [13].

Fig. 2. Process flow of selected product

The process map of selected product is depicted in fig. 2. The process starts with the fetching of components from the component area. The main activities along with data collection points (highlighted green) are defined and can be seen in fig. 2, it ends with packing of finished product. As mentioned before the focal area of concern during this research was testing and troubleshooting & repair activities.

2.2 AS-IS process model

Fig. 3 presents the AS-IS process model of testing and troubleshooting of a product. The process gets started if a unit (product) final test fails. To prevent the possibility of test software failure the unit will be sent to repeat testing. If the retest showed the same result then unit will be disassembled and inspected by a troubleshooter. Troubleshooter is using the troubleshooting application to see what the failure code was and to track corrective actions. After the possible fault cause was found the unit is assembled and sent again to the testing. This procedure will be repeated until the test is passed. The measurements from test stations as well as corrective actions taken by troubleshooters are transferred to and stored in the troubleshooting database.

Fig. 3. AS-IS Process model

2.3 Statistical analysis for need of retest

In the following the application of statistical methods and historical data for the clarification of the need for retest is discussed. The key question was which symptoms (problems) are likely to cause the product to fail the retest without repair. In case of these symptoms it may be useful to omit the retest and send the product directly to repair. This may contribute to the reduction of the testing time. The logic and questions relevant for clarification of the need for retest are the following:

- Based on historical data, which symptoms have been identified in cases where the first test showed the problem but the retest did not (the product actually passed the retest)?
- Based on historical data, which symptoms have been identified in cases where the first test as well as the retest showed the problem?
- Do the symptoms and their frequency of occurrence differ statistically significantly in these two groups?
- What is the probability in case of a symptom found during the first test that: (I) the retest shows no symptom; (II) the retest still shows symptom?

The aim was to clarify which symptoms have statistically significantly higher frequency of occurrence in the group where both the first test as well as the retest showed the problem. This question can be clarified by means of chi-square test [14]. To start the analysis, the frequencies of occurrence of the symptoms were identified and compared in the two groups. For each symptom the total frequency of occurrence has to be calculated by summing the frequency in the group where the first test showed the problem but the retest did not and the frequency in the group where both the first test as well as the retest showed the problem. Based on the sizes of the groups the “expected frequencies of occurrence” can be calculated. Then the symptoms were filtered out in case of which the expected frequencies of occurrence were smaller than the actual ones for the group where both the first test as well as retest showed the problem. These symptoms were sorted in the increasing order according to percentage of passed retests whereas the symptoms in case of which there were too few observations to analyse were eliminated. The statistical significance was validated by means of chi-square test. Based on the analysis, the list of symptoms can be elaborated in case of which the recommendation is to omit the retest.

The possible benefits of applying the recommendation of not carrying out retests for particular symptoms include:

- It enables to solve the symptom more quickly;
- It enables to test other products earlier.

2.4 Analysis of troubleshooter needs

In order to start the analysis troubleshooter needs a special troubleshooting database. Then, there can be seen a whole unit testing and repair history. The main problem of the process flow was the presence of risk of bad data appearing in the troubleshooting database. The incorrect data is any discrepancy between actual actions of troubleshooter and logged ones or actual failure caused and the one that was tracked. There were found four possible scenarios that lead to the incorrect data appearing in the database, all of them are human factor based: *Incorrect fault group, test instability, actual fault undetected and incorrect component replaced*.

Currently on the phase of failure logging to the DB, human factor based errors were not handled. From that point the incorrect information stored into the database was available for access by other troubleshooters as a result of possible solution for given failure. Most unfortunate situation was that when information misleads the less experienced troubleshooter; that may result into the material, components, time and finally the money waste.

The main points that should be considered to develop the existing database and decrease the time of troubleshooting:

- Cost of the components replaced during the repair. The decision should be taken based on the component cost (minimum first) and repair success probability that leads to most beneficial combination.
- The proper selection of failure mode description based on the failure code and corrective action taken. The selection should be done not from the whole list but only from the possible root causes applicable for particular failure.
- Probability calculation for each troubleshooting action: more visual friendly and makes possible to choose the right repairing combination to be performed first.

3. Business process improvement

The improved process model for testing and troubleshooting will be presented in this section. The business process management (BPM) approach is used for the creation of TO-BE process model of troubleshooting. BPM identifies the significances of which business processes to enhance first. It helps to determine either management processes or operational processes or perhaps both to be working on first that would bring most benefits to the company [15].

3.1 TO-BE process model

TO-BE process model, an improved process is shown on the fig. 4. The starting and ending points remain the same while the events between are rearranged comparing to the AS IS model. Firstly the sending unit to the testing is now depending on the probability the test will pass, which can be counted up basing on the historical information regarding every particular failed unit. This number will be taken from the advising application (tool) designated as “Data Cleaner” on the model, developed in the scope of this project. If the percentage of the successful retests is not high enough, the unit must be disassembled and sent directly to the troubleshooter for the diagnosis. Initially it has to be checked by

the less experienced troubleshooter, who can make a decision based on the probability that is available from the advising tool. If it does not give any determined answer, then the case should be sent to the more experienced troubleshooter. After the correct corrective action found and the unit pass, successful corrective action will appear in the database and from there it will reach to advising tool. When another unit with the same fault code reaches the troubleshooting point, the less experienced troubleshooter will be able to see, what previously done to fix it and will already know how to react. The new scheme allows less experienced troubleshooter to be guided by the software and in the opposite way lets the more experienced worked to teach it. The purpose of new process is to reduce the number of unnecessary tests in order to win the time, to speed up the efficient corrective action finding and to distribute the tasks, so the more experienced troubleshooters could only concentrate on serious failures and not to waste their valuable time on the “simple cases”. This model requires the usage of advising application, which would give the rates of success for possible corrective actions assigned to the failure code. The tool (Data Cleaner) will be taught by experienced troubleshooters and will be used by newcomers as a dictionary of working solutions for the failure codes of the unit. The comparison of AS-IS and TO-BE process can be seen in the table 1.

Fig. 4. TO-BE Process model

AS IS	TO BE
Every defected unit is tested twice.	The defected unit for which failure code has low probability to be caused by test equipment malfunction will not be tested the second time in a row.
Troubleshooters' experience does not matter in selection of failed unit to be assigned to him/her.	The units that can be diagnosed by every troubleshooter will not be assigned to the most experienced ones.
Troubleshooters are only using their experience and unfiltered historical data for decision making.	Troubleshooters are using not only their experience but filtered clean historical data and component price category table for decision making.

Table 1. AS IS and TO BE Models comparison

4. IT solution development

For improvement of troubleshooting efficiency and decrease repair cost, it was necessary to analyse product final testing results and to find out the relationships between failures and causes.

Fig. 5. Process engineers' activities required for IT solution development

The actions suggested in this section will help to decrease the troubleshooting time and incorrect reporting by modifying the current test database in more convenient, detailed and user friendly way. This will make possible for troubleshooters instantly to decide what action (exchanging the component, repair etc.) should be taken to execute the process in cost-effective manner. Main problem stated by the experts was the data quality entered by testing and troubleshooting personnel. Data accuracy impacts on quality analysis and improvement efficiency. The current section

consists of analysis of troubleshooter needs, classifier for faults development, cost index and data cleaning, in the last but not the least troubleshooter application will be demonstrated. The scheme of main steps is depicted in fig. 5.

4.1. Classifier for faults development

“Classifications group and organize information meaningfully and systematically into a standard format are useful for determining the similarity of ideas, events, objects or persons. The preparation of a classification means the creation of an exhaustive and structured set of mutually exclusive and well-described categories” [16].

All test failure codes can be grouped according to their nature and the process step these are coming from. It is important to take into consideration that some of the failure codes can be related to different process steps, so the troubleshooter can put correct failure group definition once the failure is successfully eliminated. For understanding the relation between failures and causes the classifier structure has been created. As a base for the elaborated classifier, the standard DOE-NE-STD-1004-92 was used [17].

4.2 Cost index implementation

All components have their own part number (P/N) and price. For every component the special cost indexes was assigned. There were three visually different types of indexes implemented: colour index, price bar and price index. Price index was selected as most suitable and simple one to appear in new troubleshooter application from lowest “A” to highest “C”.

4.3 Data cleaning

All data charts were carefully checked to exclude the mistakes in data entering or reporting that was done during the period taken for analysis. Potentially incorrect reported corrective actions were described and analysed according to component part numbers. Some of them were completely deleted from “cleaned” data list when it was not possible to define the right corrective actions. Others were transferred to “read as” correct actions not to lose important data for statistics. After all the corrections were done and the failures were excluded, the probability can be calculated for new troubleshooting application.

4.4 Troubleshooter Application

A prototype was made that consist of series of tasks allowing designer and user to carry out each task independently and systemically. When application opens the only enabled button is “Data Import” from the same name section; user tacitly offered to click it. After data import the next three buttons get active: “SAP Import”, “Logical” and “Statistical”.

Fig. 6. Troubleshooter application

It is not necessary to import the SAP (ERP) data to proceed with cleaning the raw data based on logical rules. The troubleshooter's report can be also generated without ERP data, but the price category will be missing then. So when the “Logical” button presses the process of data cleaning starts and it lasts few seconds. When it's finished the information message will appear making short report of how many rows were inspected and how much the clean useful data amount has been found. The same output will be after selecting the button “Statistical”, except the fact that the different process will be activated, and the data will be cleaned based on the statistical analysis. Report section also has combo box for choosing the fault code as shown in fig. 6.

After successful completion of application a report is generated with the following captions: FAULT_CODE, CAUSE_PRODUCT_NUMBER, Probability (%), Cost Index, and Corrective Action as illustrated in fig. 7. Some captions remain in title as it is in the database because troubleshooters are used to operate with those names and the other applications also used them. Cost Index column is multi-coloured; colours are distributed depends on the Cost Index itself.

Time period						
Product number + revision				Manual Input		TS Report
Serial number						
FAULT_CODE	CAUSE_PRODUCT_NUMB	Probability (%)	Cost Inde	Corrective Action 1	Corrective Action 2	Corrective action 3
Fault Code 1	Product 02	38%	C	Material Functional Fault	OTHER Unknown cause of fault	Material Mechanical Damaged
	Product 04	45%		Test Hardware Fixture fault		
	Product 01	9%	A	FA Missing	Material Functional Fault	Material Mechanical Damaged
	Product 05	2%	C	OTHER Unknown cause of fault		
	Product 03	6%	B	REP Solder Damaged	Material Functional Fault	FA Solder Lack of solder

Fig. 7. Troubleshooter report through application

5. Solution implementation

As a business process improvement framework along with data cleaner application was developed to solve the research questions. It enables to filter out the incorrect data and store correct product failure with corresponding corrective actions. The application would get the inputs from statistical analysis of test results for decision making improvement. Moreover, business process management helps to mapped a structure of information sources for data and figured out a responsible person of each activities performed during the process. Under table 2 compares estimated financial and time spending for each solution. The relative convenience ratio of each solution is graded on a scale from “0” to “+++”. From the table, the cheapest and the fastest solution is using of the prototype. However it is the less convenient from the three presented options, due to the fact that it increases the level of disunity. It can be avoided by choosing the third option, the most expensive and longest to implement, but the most convenient to use. So as the compromise the while the second or third option is in the development phase, the first (using of the ready prototype) could be implemented.

Fig. 8. Common framework of activities for troubleshooting work improvement

During the prototype implementation process the application showed the successful results. All the features considered as required to improve the performance of troubleshooters were implemented into the prototype and tested by troubleshooters. The feedback was positive and troubleshooters agreed with the layout the solution have; it makes no obstacles to work with and can become a handy tool in their routine working process. Troubleshooter and team leader were satisfied with the functionality of the Data Cleaner. In fifteen cases from eighteen the application suggested the efficient corrective actions, and rest of the cases were analysed.

The case study is sum-up in the fig. 8 where we can see the common framework of activities for troubleshooting work improvement. This framework can be used for the improvement of other similar diagnostic processes.

Solution Option	Convenience Level	Development Cost*	Implementation Time*
Prototype	+	0	1 month
Data Warehouse changes	++	4000 €	6 months
External development	+++	6000 €	12 months

Table 2. Implementation Options Comparison

6. Conclusion

In this research a framework for business process development is established to discover an improved solution of stated problem, the framework consists of business process analysis, business process improvement, IT solution development and proposed business process implementation. The AS-IS troubleshooting process of a case company is analysed and improved through business process management approach. The problem in troubleshooting process was inaccuracies in troubleshooting data that consequently leads to wrong corrective actions of certain faults. Such inaccuracy

causes additional expenses and time delays, which has a harmful effect on production outcome. A data cleaner application is developed in order to perform data filtering procedure according to the new troubleshooting process flow, it helps to enhance the decision making and performance of troubleshooters while speed up the production and reduce the waste of materials. Moreover, the proposed process model is suggested to divide troubleshooters in groups based on their working experience. Also the new application prototype supports the process of teaching of beginners and less experienced troubleshooters. The results of validation process were found to be positive during the implementation of solution. One of the future plan could be to extend this framework of business process improvement which may include risk analysis and explore the approach into service industry.

7. Acknowledgement

This research was supported by Estonian Ministry of Education and Research for targeted financing scheme B41. In addition, the UNIDEMI author would like to appreciate the funding from Fundação para a Ciência e Tecnologia, Project: UID/EMS/00667/2013.

8. References

- [1] Karjust K., Pohlak M., Majak J., Technology route planning of large composite parts, *International Journal of Material Forming*, 2010, Vol.3 (1): 631– 634.
- [2] Sahno J., Shevtshenko E., Karaulova T., Tahera K., Framework for continuous improvement of production processes, *Inzinerine Ekonomika- Engineering Economics*, 2015, 26(2), 169–180
- [3] Lee N.C., *Reflow soldering processes and troubleshooting: SMT, BGA, CSP and Flip Chip Technologies*, Newnes, 2002.
- [4] Yin R.K., *Case Study Research – Design and Methods*, Sage publications, Thousand Oaks, CA, 1994.
- [5] Mandeep Walia., *Functional testing – Challenges and best practices*, Infosys Technologies Ltd., 2013.
- [6] Sahno J., Shevtshenko E., Quality improvement methodologies for continuous improvement of production processes and product quality and their evolution., 9th International DAAAM Baltic Conference “Industrial Engineering”, 2014, 181-186.
- [7] Madison D., *Process Mapping, Process Improvement and Process Management: A practical guide for enhancing work and information flow*, Paton Press, 2005.
- [8] Kangilaski T., Poljantchikov I., Shevtshenko E., Partner network and its process management., *ICINCO*, 2013, vol2, 519-527.
- [9] Kangilaski T., Shevtshenko E., Dynamics of Partner Network, *IEEE 23rd International Symposium on Industrial Electronics*, 2014, 105-110.
- [10] Weckenmann A., Akkasoglu G., Werner T., Quality management – history and trends, *The TQM Journal* , 2015, Vol.27 (3), 281 – 293.
- [11] Brook Q., *Lean six sigma and minitab*, third Ed., OPEX Resources Ltd., 2010.
- [12] Abdulmalek, F.A. Rajgopal, J., Analyzing the benefits of lean manufacturing and value stream mapping via simulation: A process sector case study. *Int. J. of PE*, 2007, 107, 223-236.
- [13] Mahmood K., Shevtshenko E., Analysis of Machine Production Processes by Risk Assessment Approach, *Journal of Machine Engineering* , 2015, vol.15, 112 – 124.
- [14] Curtis, K., Youngquist, S. T., Part 21: Categorical Analysis: Pearson Chi-Square Test. – *Air Medical Journal*, 2013, vol. 32, no. 4, pp. 179-180.
- [15] Jeston J., Nelis J., *Business Process Management: Practical guidelines to successful implementations*, third Ed., Routledge, New York, 2014.
- [16] Eivind Hoffmann., *Standard Statistical Classifications: Basic Principles*. Bureau of Statistics, International Labour Office and Mary Chamie, United Nations Statistics Division, 1999.
- [17] DOE-NE-STD-1004-92, *Root Cause Analysis Guidance Document US*, (1992). Downloaded from http://www.everyspec.com/DOE/DOE+PUBS/DOE_NE_STD_1004_92_262, accessed on: 2011-03-13.