Benchmarking of Root cause analysis for Software

Dr. T. SuganthaLakshmi[1], Dr. M. Saravanakumar[2]

[1]Assistant professor, Department of Management Sciences, Anna University, Regional Centre - Coimbatore. Coimbatore 641047, Tamil Nadu, India. Sugi1971@rediffmail.com
[2]Associate Professor, Department of Management Sciences, Anna University, Regional Centre - Coimbatore. Coimbatore 641047, Tamil Nadu, India. Skumar_rvs@rediffmail.com

Abstract: Root Cause Analysis is a method that is used to address a problem or non-conformance, in order to get to the "root cause" of the problem. It is used to correct or eliminate the cause, and prevent the problem from reoccurring. The visible problem or symptom is considered to be the cause and the hidden problem is ignored. Hence root cause analysis is necessary to fix a problem or non-conformance forever. When root cause analysis is done in an unstructured and undisciplined approach, proves to be useless as the time and cost involved in this process are high. Experts are employed to make a proper root cause analysis, which means the cost involved is high and the time consumed should be used efficiently. Manufacturing industries for decades follow some structured and disciplined approach for root cause analysis and have proved for number of times the analysis have a positive outcome. One of the methods in root cause is dominos phenomenon. In normal problem solving methods, what we see is the final piece of domino that hits the ground. But the real cause for the domino to hit the ground was the first domino which started the triggering factor. Hence the real root cause of the problem is the first domino, where as in normal problem solving method we capture only the physical failure the domino hitting the ground and try solve it, which will never fix the problem. In this paper Root cause analysis are studied the factors that help to form a structured and disciplined approach of root cause analysis for Software, which to be benchmarked for manufacturing industries and to adapting it suitably for software.
[T. SuganthaLakshmi, M. Saravanakumar. **Benchmarking of Root cause analysis for Software**. *Life Sci J* 2013;10(3s):492-500] (ISSN:1097-8135). http://www.lifesciencesite.com. 76

Keywords: Root Cause, Benchmark, Software, analysis

1. INTRODUCTION:

Root Cause Analysis is a method that is used to address a problem or non-conformance, in order to get to the "root cause" of the problem. It is used so we can correct or eliminate the cause, and prevent the problem from reoccurring. Often Effect, the visible problem, or symptom is considered to be the cause and the big problem is missed out. Hence root cause analysis is necessary to fix a problem or non-conformance forever.

But when this root cause analysis when done in an unstructured and undisciplined approach, proves to be useless as the time and cost involved in this process high. Experts are employed to make a proper root cause analysis, which means the cost involved is high and the time consumed should be used to do something which is useful.

To make the root cause analysis useful and fruitful, we need a structured and disciplined approach to get things right for the first time. Manufacturing industries which are in existence for decades and centuries do follow some structured and disciplined approach for root cause analysis and have proved for number of times the analysis made to be useful and fruitful.

To have a better understanding on what a root cause is, dominos phenomenon comes to rescue. In normal problem solving methods, what we see is the final piece of domino that hits the ground. But the real cause for the domino to hit the ground was the first domino which started the avalanche. Hence the real root cause of the problem is the first domino, where as in normal problem solving method we capture only the physical failure the domino hitting the ground and try solve it, which will never fix the problem.

OBJECTIVE OF THE PROJECT

Primary objective

To formulate a structured way of root cause analysis for software, from existing methods.

Secondary objective

Root cause analysis a cost effective one..

To improvise the root causes analysis.

2. RESEARCH METHODOLOGY

Research can also be defined as a scientific and systematic search for pertinent information on a specific topic. It is an art of scientific investigation. According to Clifford Woody research comprises defining and redefining problems, formulating suggested solution; collecting organizing and evaluating data. Secondary data was the main source for this study. Information are collected from internet, data available on root cause analysis for software from various sources including word of mouth. The research design used for this study is Descriptive.

This project is a desk research and all the data used here in this project are completely collected from the data available already.

Limitations of the study

- Data collections are limited to some specific areas – only secondary data are used.
- Problem kind may vary from person to person based on their perception.
- Limited time period is available for this study.
- Data collected from primary sources may be biased based on the way the question was put forth.

3. PARARMETERS CONSIDERED FOR THE BENCHMARKING PROCESS

- Problem Measurability
- Definability of the problem
- Classification of the problem
- Selectability of the problem solving tool
- Differentiability between a Cause a Symptom
- RCA Complexity based on the technical complexity

3.1 Problem Measurability:

Problem measurability is one of the major factors in deciding whether to go for root cause analysis or not. An organization can decide to go for root cause analysis and fix the problem, only if it is going to gain something out of it. Huge investment with zero returns is something that no company likes to have it. If the root cause analysis is going to teach some valuable lessons to avoid such errors or problems in the future then it is worth full to make the root cause analysis.

Or if the organization is going to gain some good will on their customers and because of that they are going to get some business in the future, then RCA can be said to be useful. Unless and until the Organization or the company or the boss is not going to gain anything at the end of the day, then the chances for RCA diminishes.

Inference: Problem that are really useful to the organization should be given more importance in case the root cause analysis is going to consume more time and effort.

3.2 Definability of the problem:

"Problem" is defined as a gap between "the way it should be" and "the way it is". The problem statement is a clear and concise statement that describes the problem to be addressed in measurable terms.

To accurately define a problem statement, there are some simple questions to be answered:

- What is the problem,
- When and where did it happen,
- What is the scope of the problem?
- How often does the problem occur?
- What is affected by the problem?
- When the problem is considered really a problem?
- Is the environment when the problem occurred noted and described?(For problems that are not reproducible often)

3.2.1 Rephrase the Problem.

When an executive asked employees to brainstorm "ways to increase their productivity", all he got back were blank stares. When he rephrased his request as "ways to make their jobs easier", he could barely keep up with the amount of suggestions. Words carry strong implicit meaning and, as such, play a major role in how we perceive a problem. In the example above, 'be productive' might seem like a sacrifice people are doing for the company, while 'make your job easier' may be more like something people are doing for their own benefit, but from which the company also benefits. In the end, the problem is still the same, but the feelings — and the points of view — associated with each of them are vastly different.

3.2.2 Expose and Challenge Assumptions.

Every problem — no matter how apparently simple it may be — comes with a long list of assumptions attached. Many of these assumptions may be inaccurate and could make the problem statement inadequate or even misguided.

- The first step to get rid of bad assumptions is to make them explicit. Write a list and expose as many assumptions as possible — especially those that may seem the most obvious and 'untouchable'. That, in itself, brings more clarity to the problem at hand.
- But proceeding further and testing each assumption for validity: thinking in ways that they might not be valid and their consequences. What we will find may surprise us: that many of those bad assumptions are self-imposed — with just a bit of scrutiny those could be safely dropped.

3.2.3. Find Multiple Perspectives.

Before rushing to solve a problem, always make sure that the problem is looked from different perspectives. Looking at it with different eyes is a great way to have instant insight on new, overlooked directions.

- For example, the owner of a business, who is trying to 'increase sales', should try to view the problem from the point of view of, say, a customer. For example, from the customer's viewpoint, it may be a matter of adding features to the product that one would be willing to pay more for.

- Rewrite the problem statement many times, each time using one of these different perspectives. How would the competitor see this problem? The employees? A house wife?
- Also, how the people in various roles would frame the problem should be thought upon. How would a politician see it? A college professor? A nun? Try to find the differences and similarities on how the different roles would deal with your problem.

3.2.4. Use Effective Language Constructs.

There is not a one-size-fits-all formula for properly crafting the perfect problem statement, but there are some language constructs that always help making it more effective:

- **Make it positive.** Negative sentences require a lot more cognitive power to process and may slow down — or even derail the train of thought. Positive statements also help to find the real goal behind the problem and, as such, are much more motivating. For example: instead of finding ways to 'quit smoking', ways to 'increase energy', 'living longer' and others are much more worthwhile goals.
- Frame the problem in the form of a question. Human brain loves questions. If the question is powerful and engaging, brains will do everything within their reach to answer it. Brains will start working on the problem immediately and keep working in the background, even when people are not aware of their brain activities.

3.2. 5. Make It Engaging.

In addition to using effective language constructs, it's important to come up with a problem statement that truly gives an excitement so people are in the best frame of mind for creatively tackling the problem. If the problem looks too dull invest the time adding vigor to it while still keeping it genuine. It should be made more enticing.

3.2.6. Reverse the Problem.

One trick that usually helps when people are stuck with a problem is turning it on its head. If one want to win, find out what would make the one lose. To find the ways of increasing the sales, set the target to find the ways to decrease the sales. Then, everything that is needed is to reverse the answers.

- 'Make more sales calls' may seem an evident way of increasing sales, but sometimes we only see these 'obvious' answers when we look at the problem from an opposite direction.
- This seemingly convoluted method may not seem intuitive at first, but turning a problem on its head can uncover rather obvious solutions to the original problem.

3.2.7. Gather Facts.

Investigate causes and circumstances of the problem. Details such as the origins and causes about the problem should be probed thoroughly. Especially if there is a problem that's too vague, investigating facts is usually more productive than trying to solve it right away.

- If, for example, the problem stated by a house wife is "You never listen to me", the solution is not obvious. However, if the statement is "You don't make enough eye contact when I'm talking to you," then the solution is obvious and one can skip brainstorming altogether. (one will still need to work on the implementation, though!)
- When framing a problem statement, several questions about it should be asked. What is not known about it? When did it last work correctly? Is it possible to draw a diagram of the problem? What are the problem boundaries? Asking questions and gathering facts is important. It is said that a well-defined problem is halfway to being solved: Correction can be made for it, saying a problem statement that is defined perfectly is not at all a problem anymore!!!

3.3 Classification of the Problem:

Problems can be broadly and further down more precisely classified.

3.3.1 Common cause problem:

Problems that are occurring to everyone in a team should have to be considered as a common problem. These problems do have a common root cause, which when removed will have the problem solved.

3.3.2 Special cause problem:

Problems of this kind are occurring at some instances due to insufficient design or incomplete evaluation of the impact in case of Change Request (CR) etc.

Apart from the above mentioned kind of problems, problems can further be classified in the order of the number of causes causing the problem. It can be classified as

1. Single cause problem
2. Multiple cause problem.

Single cause problem: Problems that are caused by single problem, which when removed removes the problem completely out of the system.

Multiple cause problem: Problems that have more than a single cause causing the problem. For a problem of this kind, it is necessary to remove all the causes to remove the problem. Hence the way of analysis for problems of this kind should be in a different way.

Furthermore the problems can be classified based on the reproducibility too.

Therefore it is clear that, proper classification of the problem is necessary for the proper root cause analysis without which may the root cause analysis may take up long time to resolve the issue. Sometimes in case of multiple cause problems, chances are there not all the issues resolved or not all the causes are found out and removed. In that cases the issue persists in the system and may come later in future.

Inference: Each problem has needs to be analyzed carefully and found out to which category the problem belongs to, and then the further steps to be carried should be planned accordingly.

3.4 Selection of a tool

Tools selection plays a vital role in the root cause analysis. Selection of a tool is based on the previous parameter " Classification of the problem". Incorrect tool selection may end up either in wrong root cause which might be a symptom or in increase of the cost involved for root cause analysis.

3.4.1 Tools and Techniques to determine root cause:

Technique Name: 5 Whys

Features: The Five Whys represent more of a conceptual, as opposed to a systematic, fact-based approach, to root cause analysis. It was adopted from the Japanese approach to management, most notably practiced by Mr. Shingo, who would use the five whys on the production floor when he would tour manufacturing sites. In essence, Mr. Shingo would continue to ask why five or more times to get to the true cause of a problem, with his questions being structured to help lead the employees he was talking to towards the problem's source.

Advantages : If a person knows how to ask good, successive 'why' questions, and is able to ask them of the right people, he or she will find at least one root cause for a given problem. This approach takes little time to perform – as few as five minutes can be used to perform a five why analysis – and does not require the use of special software, flip chart paper, or reading materials. If it is performed repeatedly with the same group of people in a sound manner, its use can lead to a new way of thinking amongst those people that have been exposed to the tool's use.

Disadvantages: The 5 Why approach normally leads to the identification of just one root cause for the problem in question. One needs to go through the '5 Why' process several times for a given problem in order to ensure that all root causes are identified, and being able to do so effectively requires even more skill of the part of the question asker. It also does not necessarily point the problem solver towards the generic causes of similar problems.

This approach requires significant skill in order to learn how to ask the right why questions – the five why technique is not as simple as asking 'why?' alone five times. While the use of this tool will lead to the definition of a root cause that is also a change that is needed (a corrective action), it does not often result in a corrective action that is well developed and defined. Most people fail to gain much success when using this tool simply because they cannot develop the ability to ask good 'why' questions in succession, even though Mr. Shingo was quite skilled at doing so. Also,

- Tendency for investigators to stop at symptoms rather than going on to lower level root causes.
- Inability to go beyond the investigator's current knowledge - can't find causes that they don't already know
- Lack of support to help the investigator to ask the right "why" questions.
- Results aren't repeatable - different people using 5 Whys come up with different causes for the same problem.
- The tendency to isolate a single root cause, whereas each question could elicit many different root causes

3.4.2 Technique Name: Fault Tree Analysis

Features : My perspective of fault trees is that they encourage the user to (1) ask the five whys multiple times for a given type of problem and (2) evaluate several possible problem causes on one diagram (similar to the manpower, methods, materials, and machines boxes on a fishbone diagram). Like the other common root cause analysis approaches, fault trees tend to be a predominantly opinion-based tool, in that there are no predetermined questions that are used to help you create the branches of a given tree.

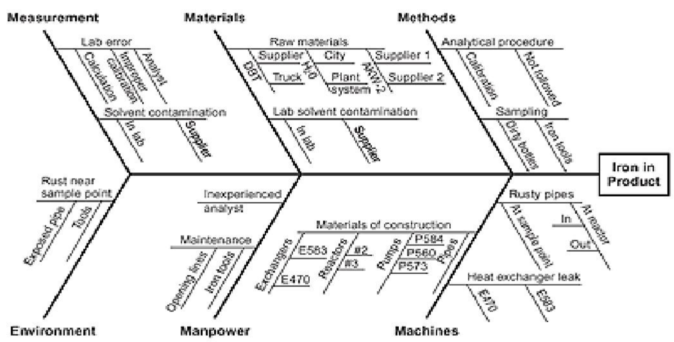
Advantages : From my perspective, I prefer fault trees over fishbone diagrams because their design allows four to five levels of 'why' to be identified for a given problem, if the users are willing to exercise a high level of discipline as they draw their charts. I find them to really be useful for troubleshooting reoccurring problems, such as quality defects, because such problems tend to have a common set of causes and sub-causes. ***Disadvantages:*** Fault trees typically fail because (1) people do not use them in a disciplined manner to develop multiple problem causes at each level, (2) multiple levels of potential causes exist to be sorted through for each problem type, and (3) they are opinion driven. They often tend to be a blend of a cause effect diagram and a flow chart, but in such cases, the user can easily get lost and not arrive at any particular root cause. Also, a well-developed fault tree often leads the user to discover that the same management systems (such as poor training, employee turnover, weak communications, and poor procedure design) are at the root of their problems. The main disadvantage lying inside this analysis is, it is time consuming as well as needs deep knowledge about the system to perform the process correctly. Deviation at one point may end up to Africa.

3.4.3 Technique Name: Fishbone or Ishikawa Diagram:

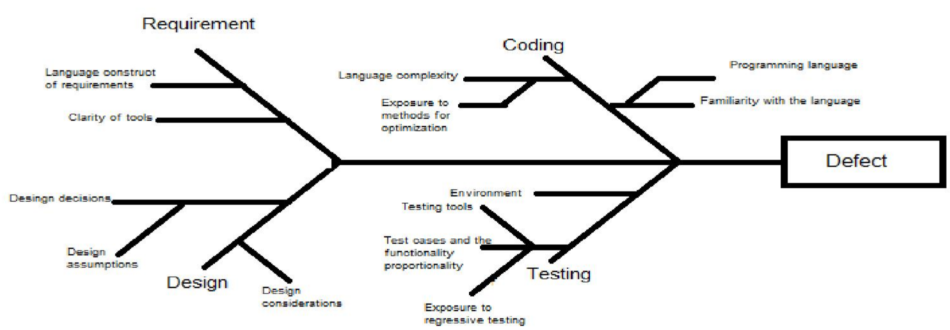
Features: This tool is perhaps the oldest, and most well-known, tool for conducting a root cause analysis. In its most common form of use, the user attempts to define multiple possible causes for a given problem in the four areas of manpower, methods, materials, and machines. The five why technique is often used with this tool to construct the bones of the chart, with the answer to each why resulting in a new branch being created off of the previous one that the question originated from.

Advantages: This tool is better than nothing, and serves as a useful tool for getting individual opinions onto a sheet of paper so that everyone involved can talk about them and suggest additional possible causes. In a lot ways, it is similar to identifying the conditions for a snap chart, but that is where the comparison ends.

Disadvantages: This is an opinion-based tool, and its design limits the user's ability to visually define multiple levels of 'why' answers unless the paper that is being used is really large. Worse yet, opinion (voting of some form) is normally used to select the most likely causes from those listed on the diagram. Teams are then encouraged to test different countermeasures for the selected causes to see if the problem goes away, which can be both time consuming and costly. The tool also does not focus on finding and eliminating.



Dispersion Fish bone



Process Fish bone

3.4.4 Technique Name: Pareto Chart

Also called: Pareto diagram, Pareto analysis

Variations: weighted Pareto chart, comparative Pareto charts

Description:

A Pareto chart is a bar graph. The lengths of the bars represent frequency or cost (time or money), and are arranged with longest bars on the left and the shortest to the right. In this way the chart visually depicts which situations are more significant.

- When analyzing data about the frequency of problems or causes in a process.
- When there are many problems or causes and you want to focus on the most significant.
- When analyzing broad causes by looking at their specific components.

- When communicating with others about your data.

1. Decide what categories you will use to group items.
2. Decide what measurement is appropriate. Common measurements are frequency, quantity, cost and time.
3. Decide what period of time the chart will cover: One work cycle? One full day? A week?
4. Collect the data, recording the category each time. (Or assemble data that already exist.)
5. Subtotal the measurements for each category.
6. Determine the appropriate scale for the measurements you have collected. The maximum value will be the largest subtotal from step 5. (If you will do optional steps 8 and 9 below, the maximum value will be the sum of all subtotals from step 5.) Mark the scale on the left side of the chart.
7. Construct and label bars for each category. Place the tallest at the far left, then the next tallest to its right and so on. If there are many categories with small measurements, they can be grouped as "other."
   Steps 8 and 9 are optional but are useful for analysis and communication.
8. Calculate the percentage for each category: the subtotal for that category divided by the total for all categories. Draw a right vertical axis and label it with percentages. Be sure the two scales match: For example, the left measurement that corresponds to one-half should be exactly opposite 50% on the right scale.
9. Calculate and draw cumulative sums: Add the subtotals for the first and second categories, and place a dot above the second bar indicating that sum. To that sum add the subtotal for the third category, and place a dot above the third bar for that new sum. Continue the process for all the bars. Connect the dots, starting at the top of the first bar. The last dot should reach 100 percent on the right scale.

Types of customer complaints – Common cause problem

Documents	Product quality	Packaging	Delivery	Poor durability	Less attenuation to noise
40	35	10	5	3	2

Pareto analysis Data

Figure 1 shows how many customer complaints were received in each of five categories.

Pareto analysis – Data interpreted

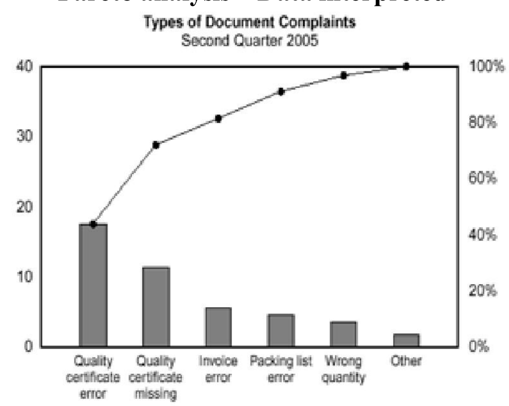


Figure 2 takes the largest category, "documents," from Figure 1, breaks it down into six categories of document-related complaints, and shows cumulative values.

If all complaints cause equal distress to the customer, working on eliminating document-related complaints would have the most impact, and of those, working on quality certificates should be most fruitful.

3.4.5 Technique Name: Scatter Diagram

Also called: scatter plot, X–Y graph

Description:

The scatter diagram graphs pairs of numerical data, with one variable on each axis, to look for a relationship between them. If the variables are correlated, the points will fall along a line or curve. The better the correlation, the tighter the points will hug the line.

- When you have paired numerical data.
- When your dependent variable may have multiple values for each value of your independent variable.
- When trying to determine whether the two variables are related, such as when trying to identify potential root causes of problems.

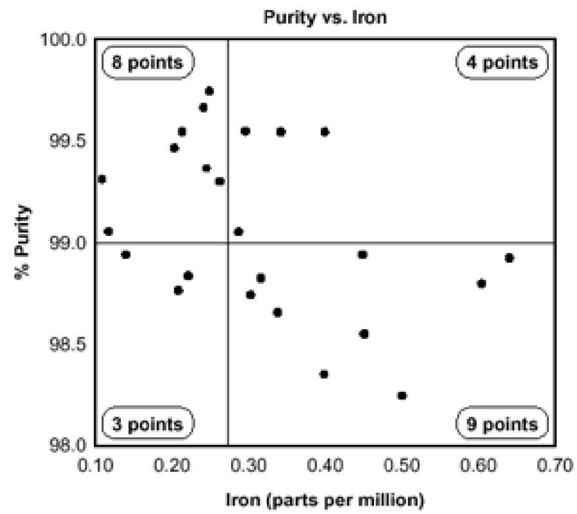
- After brainstorming causes and effects using a fishbone diagram, to determine objectively whether a particular cause and effect are related.
- When determining whether two effects that appear to be related both occur with the same cause.
- When testing for autocorrelation before constructing a control chart.

Procedure:

1. Collect pairs of data where a relationship is suspected.
2. Draw a graph with the independent variable on the horizontal axis and the dependent variable on the vertical axis. For each pair of data, put a dot or a symbol where the x-axis value intersects the y-axis value. (If two dots fall together, put them side by side, touching, so that you can see both.)
3. Look at the pattern of points to see if a relationship is obvious. If the data clearly form a line or a curve, you may stop. The variables are correlated. You may wish to use regression or correlation analysis now. Otherwise, complete steps 4 through 7.
4. Divide points on the graph into four quadrants. If there are X points on the graph,
5. Count X/2 points from top to bottom and draw a horizontal line.
6. Count X/2 points from left to right and draw a vertical line.
7. If number of points is odd, draw the line through the middle point.
8. Count the points in each quadrant. Do not count points on a line.
9. Add the diagonally opposite quadrants. Find the smaller sum and the total of points in all quadrants.
10. A = points in upper left + points in lower right
11. B = points in upper right + points in lower left
12. Q = the smaller of A and B
13. N = A + B
14. Look up the limit for N on the trend test table.
15. If Q is less than the limit, the two variables are related.

If Q is greater than or equal to the limit, the pattern could have occurred from random chance.

Scatter diagram



Inference: From the list of tools available, it is necessary to select the tool gives out the best result in best time should have to be selected based on the type of the problem and the problem statement, as it both gives out nature and probabilities of the problem.

Symptoms and Causes

It is not uncommon for problems to be reported as symptoms. For example, the symptoms like noise, no power, not working correctly, machine down, broken tool, quality problem, worn out, not as specification, management problem, too much variation, breach of metrics, etc. are reported as problems!

The tendency is generally to treat the symptoms rather than the underlying fundamental problem that is actually responsible for the situation occurring (root cause). Even though we make an attempt to solve these symptoms, the problematic situation may likely occur again, and must be dealt with over and over again. The costs of these quick solutions can be high over time. To solve a problem, one must first recognize and should able to differentiate the problem from a symptom

Symptoms

Symptoms are the signals to indicate that our system is out of balance/ control. To be more precise, symptoms are nothing but the physical failures that are visible outside. If we are satisfied merely with making the symptoms disappear, we have lost a valuable opportunity to look more deeply into what may be incorrect.

Root Cause

A root cause is the most basic reason for an undesirable condition or problem. The root cause is the most basic casual factor or factors that, if corrected or removed, will prevent the reoccurrence of the situation. The purpose of determining the root cause is to fix the problem at its most basic source so that it doesn't occur again.

If the real cause of the problem is not identified, then one is merely addressing the symptoms and the problem will continue to exist. For this reason, identifying and eliminating root causes of problems is of utmost importance. Identifying root cause is the key to preventing similar occurrences in future and improving the company quality standards.

The below table gives the difference between the Symptom and Root cause approaches.

Symptom Approach	Root Cause Approach
Errors are often a result of worker carelessness	Errors are the result of defects in the system. People are only part of the process
We need to train and motivate workers to be more careful	We need to find out why this is happening, and implement mistake proofs so it won't happen again
We don't have the time or resources to really get to the bottom of this problem	This is critical. We need to fix it for good, or it will come back and burn us

Inference: It is very important to solve out the causes rather than removing the symptoms from the system. Removing symptoms is like putting a exhaust fan to a forest fire rather than controlling the fire. By removing the symptoms the problem may temporarily hidden from the system, but at any point of time it may come again in a different form.

3.4.6 RCA Complexity based on the technical complexity:

Complexity of root cause analysis tends to increase along with the complexity of the problem. When the problem is too much complex then the tools, that has to be used find root cause analysis needs to combine. That is one after the other; more than one tool has to be used to identify the real root cause. Hence the root cause analysis process too tends to be too complex to arrive at a solution and to fix it.

Inference: Root cause analysis methods and techniques gets complicated as the problem becomes more and more intermittent. It is necessary to reproduce the problem at least once before going on to solve the problem and document the same for future reference incase similar kind of problem or issue detected somewhere. For the problems that occur intermittently, best method of root cause analysis to be used is **Fault tree analysis**.

Structured Approach:

Structured Approach is the mantra that is going to help us end the race to find the real root cause. The concept behind structured approach is nothing but "the art of putting things in order". It is very essential to put the things that were analyzed in the day to be in order to get some

1. Correct evaluation of the problem.
2. Clear and precise problem description using a standard questionnaire.
3. Exact classification of the problem.
4. Tool for the analysis which should be chosen from the above steps.
5. Ensure that all causes arrived are really causes and not symptoms.

Last point should have to be ensured by calculating the cyclomatic complexity of the software. All the possible flow of control should have to be taken into account and all the control paths should be error free, so that we can say that the problem is completely removed from the system.

When all the above mentioned steps are followed in exact order, then the root cause analysis becomes an easier task to handle and reduces the time for it.

4.CONCLUSIONS

4.1 SUMMARY OF FINDINGS

- The problem statement which is an important input for the root cause analysis should be made proper with the problem expressed without any ambiguity and over details. Also the selection of tool based on the cause of the problem is plays a vital role in reaching the root cause of the problem.
- When following some unstructured method of root cause analysis, the time taken to reach the root and fix it becomes a mammoth task most of the time. The same done with structured methodology, the solution looks promising and can be done in a short span of time.
- Classification of the problem and the clear problem statement are the basic inputs for the tool selection, which when selected properly, problem solving becomes easier.
- A problem that can't be reproduced or reproduced very rarely needs specialized method of solving, which involves the fault tree analysis tool for problem solving and the entire procedure needs to be documented.

4.2 SUGGESTIONS AND RECOMMENDATIONS

- It is often better to start the root cause analysis as a team, which will prevent any individual biasing over the problem and approach to solve it.
- Problem statement formation template can be formed based on the needs and every time it can be used to formulate the problem statement, which will help the organization to

perform root cause analysis even in case of absence of experts.

- From the experience gained on the process of root cause analysis, the entire process can be made a software package, with self-explanatory questions so that the process of root cause analysis can be automated completely and can be stored in the database for easier retrieval of data and eliminating the needs to reinvent the wheel every time.

CONCLUSION

- Reports say that, any work that is executed mistakenly in the first step will cause a defect or problem which gets multiplied by 8 in the forthcoming process. Hence it is necessary to have proper problem statement for a root cause analysis. This report clearly gives out the way to write a problem statement for a root cause analysis process and helps identify a proper tool for solving the problem through the problem classification.
- This report promises to give a good start for root cause analysis by forming a proper Problem description, which

1. clearly states the problem,
2. when the problem is really a problem

and gives a clear sight to the management to carry forward with the problem. From there, exact classification of the problem, proper tool selection, takes closer towards the solution and making the root cause analysis successful.

APPENDIX 1: Automotive applications and Web applications

The difference between automotive software and web application software are mainly in terms of memory and speed of execution. For automotive applications, it is necessary to perform an operation in a particular point of time, but for web application, if an operation can't performed in this cycle, it can be performed in next cycle as it is not going to make any big difference in the execution. Automotive applications are mostly real time systems, which needs to be accurate every time. Also the memory and runtime constraints are more due to the limited resources for automotive applications. But in case of web applications, the memory available is much more higher than that of automotive applications.

1/2/2013

APPENDIX 2: Problem statement formulation template

1. What is the problem?
2. When is the problem considered to be a problem?
3. When is the problem noted?
4. Where and how was the problem occurred in the field?
5. Is the problem reproducible in laboratory conditions?
6. Is there more than one way to reproduce the problem?
7. Is the problem occurring only during the first time after power on?
8. Who involved in the problem?
9. What is the impact of the problem?

Corresponding Author:
Dr. T. Sugantha Lakshmi
Assistant professor, Department of Management Sciences, Anna University, Regional Centre - Coimbatore.
Coimbatore 641047, Tamil Nadu, India.
Sugi1971@rediffmail.com

REFERENCES:

BOOKS AND ARTICLES:

- Keith Mobley (1943) Root cause failure analysis, PP 15- 45
- Toyota Japan, Nazenaze_Mizuno_E_improved_5Why.
- US Department of Energy ,Root cause analysis guidance document, Feb 1992
- C.Robert Nelms, Root cause Analysis – Not what YOU Might Think, Failsafe Network Inc., 1992.
- Alexander (Sandy) Dunn(2004), Getting Root Cause Analysi to Work for You, , Director, Assetivity Pty Ltd
- Nathan Logan(2002), Pareto Analysis: When Quality-Control Demands Decisions.

WEBSITES

http://www.dwassoc.com/5-whys.php
http://en.wikipedia.org/wiki/5_Why
http://en.wikipedia.org/wiki/Pareto_analysis
http://en.wikipedia.org/wiki/Ishikawa_diagram
http://en.wikipedia.org/wiki/Fault_tree_analysis
http://www.slideshare.net/mssridhar/introduction-to-research-methodology-presentation
http://www.mindtools.com