Advanced Problem Solving for ITIL® 4

ROOT CAUSE ANALYSES FOR IT INCIDENTS

ITpreneurs™
Effective Learning Solutions

Advanced Problem Solving for ITIL® 4

Root Cause Analyses for IT Incidents

Cover design by pikisuperstar - www.freepik.com

ITpreneurs Nederland B.V.
Weena 242
3012 NJ Rotterdam
The Netherlands
info@itpreneurs.com
https://www.itpreneurs.com/

Contents

Every incident has at least two answers.

Introduction

In today's ITIL world there is still much confusion about the concept of Root Cause Analysis (RCA). The two terms IT and Root Cause just don't seem to fit together, because Root Cause emanates from years back and is mostly applied in the Manufacturing Industry. There are different levels of confusion about the following, which when understood and embraced could make a whole lot of difference to staff productivity in an IT environment. The difficulties are the following:

- Root Cause is seen as the ultimate objective when it is seen as the last component of three outcomes i.e. Incident Restoration, Technical Cause and only then Root Cause
- Root Cause is perceived as a single dimension impact while when executed properly it could have a multi-dimensional and exponential impact on recurring incidents, incident rate and avoiding other related incidents.
- Root Cause is perceived to be for certain people only and not the responsibility of every IT professional. Root Cause is dependent on the effective deep dive into data analysis and not intended for everyone.

Every incident has at least two answers. One is the Technical Cause (the change or event that triggered the incident) and the second one called Root Cause (the company condition that is the underlying reason for the incident and 'WHY' it happened) that needs to be identified and removed. This second reason is commonly known as the root of the incident or the Root Cause of the incident.

In this white paper we will look at rectifying the general lack of understanding of the role of Root Cause Analysis by investigating the following topics:

- Typical problems and misconceptions in Root Cause Analysis
- Paradox of data, investigating the difference between Data Analysis and Problem-Solving
- The underlying and brilliant concept of Divergent and Convergent Thinking
- The Challenges facing the adoption of Root Cause Analysis
- The Process and Thinking Approach of Root Cause Analysis
- Specific desirable Benefits of Root Cause Analysis

Typical problems and misconceptions in RCA

IT incidents, in general, are too complex for a simple Root Cause Analysis approach. The reason why this is happening is that problems and incidents are presenting themselves at a higher level and then it does sound very complex. For instance, real-life examples such as; "servers not communicating" or "Internet banking slow" all involve the customer and there is pressure to remedy the situation immediately. The aim would be to have a robust and proven way of how to "frame" the incident in such a way to drive specificity and define a very specific OBJECT and the FAULT associated with the incident.

"We never have enough data to solve an incident quickly and accurately." There is either too much or too little data available and when the team cannot find the answer, they tend to blame the data. There is a third component and that is the relevance of the data. Normally seeking more data would lead to gathering irrelevant data and hence confuse the team. The aim should be to use a process framework that would indicate the kind of data needed and help the team to know which questions to ask and who to ask them.

Data that we need normally lies in another domain and is difficult to obtain. When this happens the team seems to think they are not allowed to use their initiative to find the data they need. We talk a lot about cross-silo collaboration, but many teams still have a problem with this concept. We are simply not "walking the talk" in this regard. It would be helpful to have a framework with common templates and an embedded structure of questions that could help with this.

We do not have time to "investigate" an incident in a time-consuming method. A formal process however simple is normally frowned upon as being too time-consuming. Problem-solving teams incorrectly associate a problem-solving process using a factor analysis approach with that of lengthy data investigations. You cannot blame them because they do not understand the protocol being used in factor analysis, which is using the available data and working with that to arrive at an answer.

If your investigation is not pitched correctly, you could end up trying to solve the effects rather than the underlying reasons. When something goes wrong it normally manifests itself as a consequence or said in another way, an effect. The problem-solver tends to latch onto this effect and then without realizing their mistake tries to find the cause of that effect which is near impossible. The secret is to take the effect and investigate the underlying reason (what has happened) and identify the correct fault to work on.

The paradox with data

We are dealing with a "contradiction in terms" when it comes to a problem-solving approach. We've noticed on numerous occasions that in the mind of the IT professional, problem-solving represents a "deep dive" into the analysis of the incident situation. This is a real contradiction because in the mind of the IT professional they are sure they are analyzing the problem, which is true, but it is not problem-solving. Let us explain…

Problem-solving is about finding an irregularity in the data that might explain why we are experiencing a particular incident. The IT professional thinks that taking a deep dive into the data would identify that irregularity, which would be fine if they were addressing the correct fault and the correct unique aspect of the fault. In the mind of the problem-solver, they are following the "Factor Analysis" approach made famous by Rudyard Kipling many years ago.

For the factor analysis problem-solver, it is about finding the correct fault for the starting point during the Divergent "Fact Gathering" phase and then narrowing down the problem with the Convergent "Thinking Approach" (this concept is explained in next section).

Through many years of experience in working through 300 Root Cause Analysis exercises with about 50 clients, we can testify that in 96% of all cases we helped the client to identify the true fault as the starting point. Up to that point, and the sole reason for not finding the Root Cause, the client was working on a general description of the fault and thus not the best starting point. Therefore, the data analyst would have a better chance of success, if only they had the means of starting with the correct fault.

The bottom line is that the process thinking approach comes before the deep dive into the data. The problem-solving approach, when handled correctly is simple, easy to follow and could provide an answer within 6 questions. The problem-solver would ask questions that highlight the 6 factors in factor analysis, they are;

- **WHAT** happened
- **WHO** is impacted
- **WHERE** is it happening
- **WHEN** is it happening
- **HOW** did it happen
- **WHY** did it happen

Many times a team will already have the answer by just asking these questions factually. In fact, a team would rarely answer all 6 questions, because they seldom have all that data available at the outset of the incident.

The summary is that Problem Solving thinking comes first and only then does the deep dive data analysis follow. It is required because the team might already have the answer. Once you understand this paradox you will understand how the concept of Divergent and Convergent Thinking can further leverage the successful efforts of the problem-solving team.

Divergent & Convergent Thinking

We all have the ability to do this because our natural thinking style follows the pattern of Divergent and Convergent Thinking. Imagine that we say to you that we are experiencing a 'Client Billing Problem' and want you to help us to resolve this issue. However, we do not volunteer any further information about this situation. You will eventually ask us to give you more information to be able to help us. This is a very natural response and so is the procedure and process of Divergent Thinking. Let us explain a typical situation to prove to you that you are already using the appropriate thinking skills and all we want to do is to get you to use the same approach in the work environment. You get to your car one morning and as you turn the ignition key it gives you that sound of 'click' and nothing happens. The ignition does not want to turn and start the engine. *So, what are you thinking now?*

Maybe it is the battery that has lost its charge – that is your potential answer to what and why it is happening. However, what do you ask yourself at this stage before jumping to any action and ripping out the existing battery and getting a new one? You need to gather more information and you need to make sure it is the battery. The only way to make sure about this is to put on the lights of the car (gathering factual information about the problem situation). You switch on the lights and they are okay! That is a new fact that just entered into the knowledge base of your situation.

You make the following argument – if it is not the battery, what could explain the fact that the lights are okay, but the ignition does not want to turn? Your conclusion is that it must be something to do with the ignition itself, possibly a loose wire or poor connection point at the starter motor? (Analyzing information for fit – Convergent Thinking). You reach the conclusion that it must be a loose wire because it is a fairly new car and you check, and this proves to be the answer.

Divergent and Convergent Thinking is a natural thinking pattern used by most people, correctly used it can be very helpful in prompting you on how to approach an incident or problem. The key to success is to learn and use the appropriate questions that would go with these two phases of critical thinking. The challenge to most IT professionals is to follow the four steps explained below in the investigation and resolution process. To make it easier we have developed customized questions recorded on a question sheet that you need to follow. It is that simple, really!

The process is simple and easy to follow. You need to follow four steps and each step has a specific tool or method that will help the problem-solver to ask the right questions from the right people and then arrive at the right answers. The steps are the following:

1. **State the Situation** – This consists of identifying the type of incident situation you are facing. Is it an incident or a problem situation or is it a situation that needs a solution? This step might seem to be fairly insignificant, but it is the step that will guide you through the rest of the analysis. It is important to get it right because not all problems are the same.

2. **Gather the Information** – This step is all about getting the information relevant to the incident situation. In the case of an incident, it would be the information surrounding the incident (factor analysis). In the case of making a decision, it would be about finding the appropriate requirements from all the applicable stakeholders. However, the tendency is to gather any information, which could include irrelevant information that would confuse the problem-solver. Every problem is unique and calls for the information most applicable and relevant to the incident or decision situation.

3. **Analyze the Information** – This is the first step in the Convergent Thinking phase and also the first step in the analysis of the information gathered. If we had to ask any individual

how they are managing this, they would not be very confident in their response. They might say something like dealing with a 'process of elimination'. That would be correct but again we would ask 'how?' and in most cases, the problem-solver would not be able to tell us. It is normally a mental process of randomly accessing bits of information and discarding those bits of information that do not seem to fit. This should be the basis of how it is done correctly, but we would suggest it needs to be a more organized, systematic and structured method (more on that method later).

4. **Reach Conclusion** – This is the step where the problem-solving team comes to a mutually agreed realization of what is causing the incident or what would be the best solution for the problem situation. It is normally a logical conclusion based on the information analyzed and narrowed down to provide the most logical answer.

Technical Cause versus Root Cause

The misinterpretation of the true definition of "Root Cause" is another obstacle standing in the way to become a good problem-solver in IT incidents. Root Cause Analysis needs to be practiced in relation to how the IT professional is supposed to approach any incident. Initially, they have to restore an incident virtually at all costs, especially if it is a Priority 1 incident affecting Business or Customers. Only once the service has been restored will they have the opportunity to identify the Root Cause.

The IT Root Cause Analysis approach is conveniently attached to three very simple concepts namely;

What happened, which is supporting Restoration efforts,

How it happened, which is about finding the Technical Cause and lastly

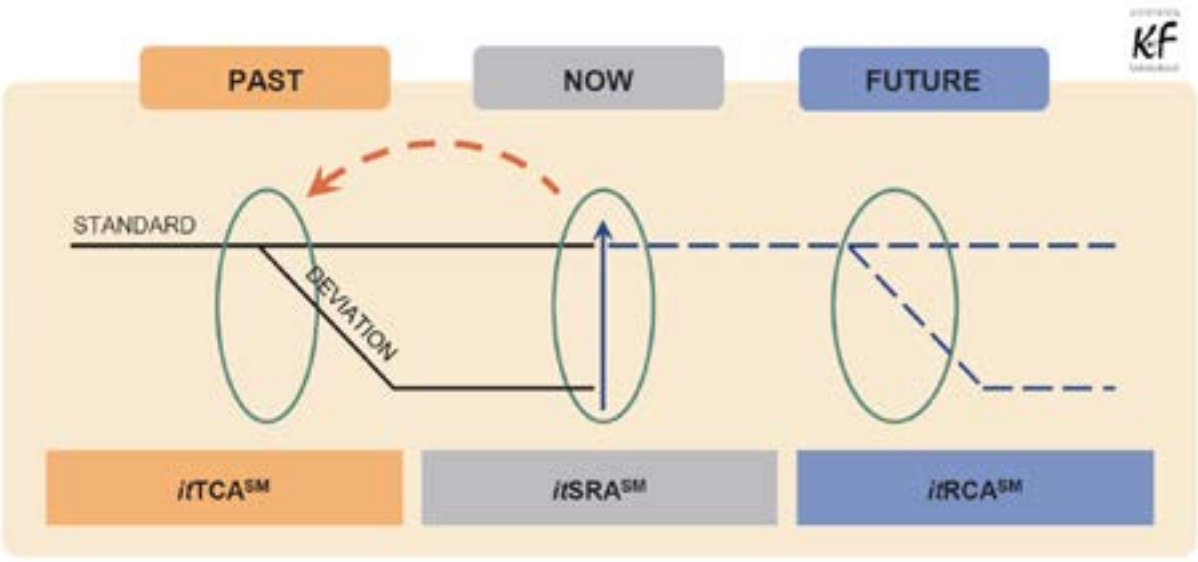
Why it happened, which would indicate the Root Cause itself.

A more detailed explanation is covered in the "Three Investigation Skills" diagram and the accompanying description.

Service Restoration Analysis (***it***SRA™) – This is a set of tools to help the team or incident investigator contain the impact of an incident on Business and Technology. This is about WHAT happened and is intended to get the problem-solver to understand the most accurate OBJECT with the most specific FAULT. The aim is to find a corrective or adaptive action that would either remove the fault or at least provide a "workaround" for the fault.

Technical Cause Analysis (***it***TCA™) – This is the set of tools that would be applied to identify HOW the incident occurred and would normally point towards an event or change that took place that "broke the camel's back". Something technical occurred that the system could not handle!

Root Cause Analysis (***it***RCA™) – This set of tools refers to the process of finding the underlying reasons, WHY a Technical Cause happened in the first place. This is normally described as a "condition that exists". It's been like that for some time and would most probably be like that for a considerable time. Unless removed this condition would create further repeat incidents over time.



Unfortunately, the search for technical and Root Causes is not simple and yet it should be. We believe with the following guidelines any IT professional will be able to improve their chances markedly if followed. Here are a few pointers, which in our experience make a major impact on the success of incident investigation;

Challenge	***Guidelines***
Team cannot make progress	– Do a systematic questioning drill to identify the correct fault. – Ensure you collect your data from the most appropriate information sources closest to the incident situation – be as specific as possible.
Incident seems way to complex	– Reduce the complexity by framing the incident with one OBJECT and one FAULT only. – Ensure you get your background information from the most appropriate Subject Matter Expert (SME).
Too much data to work through	– Stop collecting data and just answer the questions that would give information for the What, Who, Where and When factors – be as specific as possible. – Speak to the most appropriate SMEs and not the most senior ones.
Not enough data to work with	– Take the data you can get for the moment and frame it into a factual snapshot – even verify the facts if you need to. Be as specific as possible. – Always determine what you don't know and plan on how to get the information.
The quality of the data seems suspect	– Speak to the right SMEs and ensure they can verify the data. – Drive specificity by asking probing interrogative questions.

You would notice that there are a few themes running through these guidelines. This is not a coincidence at all! These themes have personally "saved the bacon" of many of our consultants on assignment with a client.

The process of Root Cause

The aim of the CauseWise process is to assist the problem-solver to determine the best route for the restoration of service and once the service is restored to also help in identifying both the Technical Cause and the Root Cause of the incident. Typically, it would be a situation where there is a technical incident such as; 'website dropping the connection' or 'users cannot log on to their online banking account', etc.

CauseWise is a process that utilizes the Divergent Thinking information/data gathering approach to establish an incident snapshot with factual data. It then utilizes the Convergent Thinking information analysis intuitively to arrive at a Consensus Restoration, Technical Cause and Root Cause for the incident.

The four steps in the process are:

1. State the Problem – The incident investigation team needs to identify the most correct and accurate object (thing) and most correct and verified fault (defect) in the incident.
2. List Problem Detail – The incident investigation team would gather factual information about the incident in the applicable appropriate dimensions of WHAT, WHO, WHERE and WHEN. We do this to create a factual snapshot of the incident and to frame the incident accurately.
3. Analyze the Information – The investigation team, with the help of SME inputs, will look at the information gathered and hypothesize specific theories on what they feel could have caused the incident.
4. Confirm Technical Cause and Root Cause – The team now uses logic and gut feel to test the SME theories against the factual snapshot information gathered. Once the team is agreed on the Most Probable Cause(s), they then devise a plan of how to verify which cause(s) is actually true. This is normally done by designing a replication to mimic the fault.

Let's examine the detail of this process by using an example and working through each step with the objective for you to gain a good understanding of how these steps are normally executed.

"We were struggling with the 'server communication' problem for weeks looking at theories from A to Z, but only when we were coached to a different statement of 'ABZ Dell server not receiving data packets' did we start looking at the relevant information and we made progress immediately. In fact, we had both the Technical and Root Causes identified and verified within two hours"

- John Hill
Infra-Structure VP for a large Retail Store

Step One – State the problem

The problem statement is the most important step in this process because it frames the accurate starting point for the incident investigation. The success of the incident investigation will rest squarely on the success of establishing the Incident Statement correctly. Did you get that? Unless you get this right, you will not be a highly effective incident investigator.

We suggest a very specific questioning drill to identify the object and fault. Basically you start with the object and clarify it to identify the initial focus.

1. Ask the question **'What is the most specific thing you are having a problem with?'**
2. Secondly, identify the most specific fault to the point where there is a good understanding of what the fault is.
3. Ask the question 'What is wrong with the object/thing?'
4. You might want to clarify it even further by asking 'What do you mean by fault "X"?' Let the owner of the incident explain in detail what they mean in the fault description and that explanation might lead you and your team to a new Object and Fault.

The diagram is a list of examples of how to simplify and improve each statement before we could work on the specific incident. In our experience, we would say that in more than 95% of cases we had to help the client to modify their original Incident Statement.

How do we do this in a real-life situation? It is a very well-rehearsed questioning drill. Look carefully at each of the statements, you would notice a few observations such as:

- The revised statement is much more specific and detailed – this is one of the critical requirements for formulating an incident statement
- The revised statement has a single OBJECT and also a single FAULT
- The revised statement does not have any information about users, location, timing, size or even the pattern of the incident situation

This questioning drill is displayed in the following example;

Note how the Incident Statement changed during the questioning. It went from 'Servers not communicating' to that of a specific 'Dell server ABC not receiving defined data packet'.

This subtle change in the wording of the statement changed the nature of the OBJECT and also made the FAULT much more specific. Look at a video clip of how this is done in a real-life situation. Click on www.thinkingdimensionsglobal.com/training-clips and click on the "Incident Statement" tab.

Step Two – List problem/incident detail

In this step, we want to collect the available factual data in all the appropriate dimensions of the incident situation. We would like to create an accurate snapshot of what the incident looks like in various dimensions such as the '**What**', '**Who**', '**Where**, '**When**', and anything that might be '**Unique**' about it.

It is also important to stress the fact that we need to deal with verified factual data and that makes it important to gather information from the right people or right sources.

Another important point to remember is that we do not always have the factual data to answer all the questions, which is okay for the initial purposes of creating an initial factual snapshot of the incident.

Many investigators have the notion that senior people would be the best to get inputs from, when it is actually just the opposite that is true. We need to talk to the people 'closest' to the incident situation to form an accurate snapshot of what has occurred.

We suggest you do this to have the best chance to ensure 100% accuracy of the facts surrounding the incident. Look at the incident situation and basically decide the following;

- What do you know about the incident?
- What don't you know about the incident and who will be the best suitable resource that would be able to provide you with the most accurate and appropriate inputs?

Information dimension	*Yes*	*No*	*Who to consult*
We know the exact object we have a problem with		x	Database operator
We have an accurate description of the fault		x	Network specialist
We know which users are effected	x		
We know which geographic areas are effected	x		
We know when the incident started		x	Branch IT networks
We know the time pattern of the incident		x	Infrastructure supervisor
We know the phase of operation this incident is happeening in	x		

Look at the following matrix on Selecting Information Sources, this should help you to decide who to collaborate with to ensure you collect specific factual data about the incident situation.

This matrix is based upon the dimensions represented in the "Factor Analysis" problem-solving approach.

When you ask the questions some names would pop-up immediately but in other cases, you will have to enquire who to consult with to ensure accurate data/information.

Arrange a time when it would be convenient for all investigators to meet and if possible, you could make use of a facilitator to manage the process of information gathering.

Look at the video clip and go to www.thinkingdimensionsglobal.com/training-clips and click on the "Incident Detail" tab. (The questions are in the diagram below).

Is	***But not***	***Why not***
What is the OBJECT you are having problems with?	What other similar objects could have the same fault, but do not?	Why do we have a problem with this object and not with the others?
What is wrong with the object? (Fault)	What other similar faults could be observed, but is not?	Why do we have this specific fault and not the other similar faults?
Where is the UNIQUE impact of the fault? Is it the USERS, LOCATION, TIMING or PATTERN or a combination of some factors? (Ask the following questions for each uniqueness identified)		
Who are the USERS or which LOCATIONS are experiencing the fault?	Which USERS and/or LOCATIONS could have experienced this fault, but do not?	Why are these specific users and/or locations experiencing the fault and the others do not?
What is the most specific TIME this incident started?	When could it have started, but it did not?	What is UNIQUE about this timing when compared to the others?
What is the PATTERN of occurrences?	What could the Pattern have been, but it is not?	Why this particular Pattern and no other pattern?

This should create contrasting information that would motivate you to ask 'WHY' this object only, or 'WHY NOT" the 'BUT NOT' objects? E.g. You could ask the "WHY NOT" question in context as follows: "Why do we have a problem with the Dell Data Servers and not the Dell XYZ servers?"

This is a natural way of thinking and we try to capitalize on this method by creating a contrast that would get the investigator to start looking at 'what makes sense' and what 'does not make

sense'. These 'WHY NOT' contributions would later become the springboard for generating and theorizing possible causes.

There are many other reasons why we prefer a method such as the one above. The following are just a few pointers;

Looking for a "BUT NOT" contribution for each of the "IS" information pieces allows the system to create a contrast for that dimension, which you may not have thought of before. This normally leads to new insights into the incident situation.

It is interesting to note that ITIL's hierarchy of DATA is utilized strongly in this approach. The "IS" data is simply just data, the "BUT NOT" data adds that extra dimension to turn data into INFORMATION and then the "WHY NOT" step would take that raw information and turns it into KNOWLEDGE; utilizing the expertise and knowledge of the Subject Matter Experts.

Is	***But not***	***Why not***
Dell ABC Server	Dell XYZ Servers	– New Servers installed over the weekend – Operator error
Not receiving Data Packets	– Data generation issue – Slow performance	– Router issue – Cache restrictions – Port configuration issue
Where is the UNIQUE impact of the fault? Is it the USERS, LOCATION, TIMING or PATTERN or a combination of some factors? (Ask the following questions for each uniqueness identified)		
Smaller outlets	Larger outlets known as BIG Z outlets	– Smaller outlets on ADSL and do own upgrades – BIG Z outlets on LAN and upgrades by techie
Monday October 10th, first thing in the morning	Any time before the 10th Later during the day	– Picked up a bug during weekend upgrades – New Excel parameters for sales reports – Normal data back-ups over weekend

What is the benefit of recording the factual data about an incident in this way? The investigation teams I've been involved with have found when they work through these worked questions systematically, they invariably find a question they have not asked before. This

'oversight' normally leads to discovering the 'hidden' information not considered up to this point. This normally leads the investigator to the actual cause.

- Discovering the 'BUT NOT' information also forces you and your team to be much more specific about the incident characteristics. This leads to clarifying the information and also forces you to be clearer about what is factual information and what is not.
- The aim is to be as specific as possible in every area of the problem detail where you are providing an answer in the system. Words such as 'Random' do not fly with this system. We see words such as random, failing, broken, not working, incorrect, out of order, blue screen, and something is dead. We regard these as banned descriptions and not to be used in incident investigations.

You could also use this information to elicit contributions from other stakeholders for their ideas of what could have caused a situation with this 'snapshot' of factual symptoms.

Step Three – Evaluate possible causes

Step three aims to help the investigator to generate and then to evaluate the causes to see whether the team managed to develop a Most Probable Cause (MPC). Once again it is important to have access to the appropriate SME information sources to generate these theories. This step involves Convergent Thinking, so we are trying to narrow down what is causing the incident situation.

How do we suggest you do this? At this stage, you have a verified factual snapshot of all the relevant information to form the basis for an effective screening framework of possible causes. In the Divergent Thinking phase, we concentrated on being factual and specific whereas in this step we will use the intuition, gut feel and logic of the SME team to generate the causes.

We look at the 'WHY NOT' information in the Incident Detail section and ask the following questions;

- Looking at all the 'WHY NOT' information, what do you think caused the incident? or
- What do you think is the 'event' or 'change' that could have caused the incident? or
- Do you have any theory why you think this incident occurred?

The second and third questions are based on the principle that as you learned more about the incident; you most probably started to develop a stronger idea of what could have caused it.

Let's look at our incident situation of the 'Dell Server not receiving data packets' example. We

have a few 'WHY NOTS' to look at. As we worked through this incident our team started to feel strongly about the listed four 'WHY NOTS';

1. The 'upgrade sent remotely' which could have been 'botched' due to lack of skills
2. The 'new turnover formula' that caused an issue with smaller outlets
3. An 'upgrade bug' that might have been introduced during the weekend upgrades
4. The introduction of 'new Excel spreadsheet' parameters, which would have needed individual upgrades from the smaller outlets.

The investigation SME team had to develop a hypothesis for each of these pieces of information. In other words, they had to describe exactly how each of the 'WHY NOT' elements could have caused the 'data packets not being received by the ABC server'.

The fully phrased Possible Causes in the Diagram were produced.

Which cause do you think is the correct one? At this point, you might have a few plausible theories of what could have caused the incident, but there is no guarantee that your cause would be the correct one. You might not even know why your possible cause could be the correct one, let alone explaining to your colleagues why you are thinking so.

Why not	***Possible causes***
Upgrade sent remotely	The ABC Server upgrade for ADSL users sent remotely and the operators did not know how to do this upgrade.
New turnover formula	The new way the turnover figures are determined is causing a conflict and does not transmit.
Upgrade bug in code	There is a coding mistake in the upgrade code causing transmission reception problems.
Now using Excel spreadsheet	The new Excel sheet parameters not loaded by smaller ADSL users, causing a sync conflict, affecting receiving.

This dilemma will bring us to the last step in the process, which is how to confirm the correct Technical Cause for the incident. There are two sub-steps in this step. The first is to test our theories on paper and the second to isolate the most probable causes to be verified in the workplace.

Step Four – Confirm Technical Cause

We are in the last phases of the Convergent Thinking for the CauseWise process. The aim is to analyze the information we have and so arrive at the Most Probable Cause (MPC) or possibly even the cause of why the incident occurred.

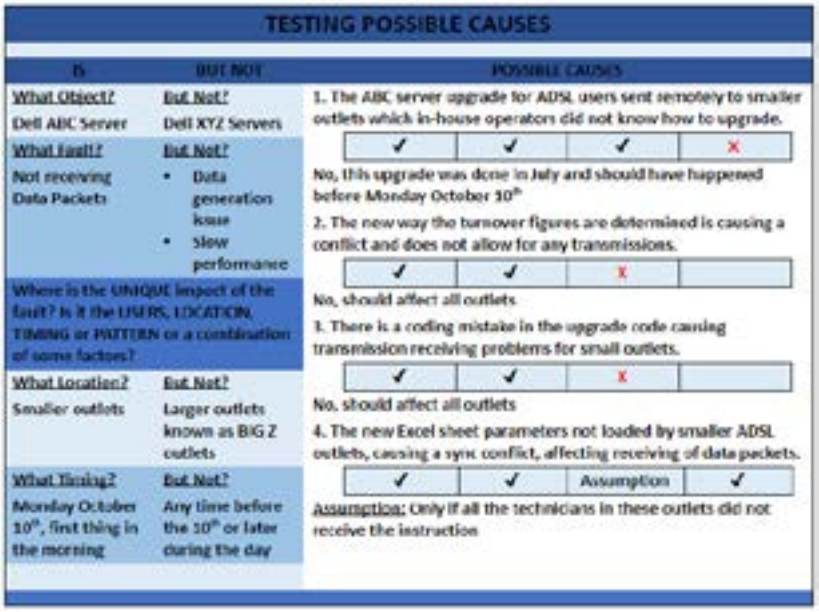
Testing the suggested causes

In this phase we look at each cause in isolation and test it (screen it) through the 'IS' and the 'BUT NOT' incident information to see which cause can explain the data. We will test each cause against each piece of 'IS' and 'BUT NOT' information until we get to one of two points. Either it cannot explain a specific piece of factual information, in which case we will eliminate this possible cause, or the cause will explain all the information. In this case we will isolate this cause to be tested and verified further in the work situation.

We will ask an important and robust question for each set of information in each of the appropriate dimensions of the incident detail. The question is; **'If "X" (the first listed cause) is the true cause, then how does it explain we have a problem with the 'IS' object and not with the 'BUT NOT' object?'**

In our DELL Server issue we tested all four Technical Causes and only the last one passed the testing phase with one assumption. See the diagram below.

If you look at the first cause 'The ABC server upgrade for ADSL users sent remotely and the operators did not know how to do this upgrade', it does very well in explaining the information for both the 'IS' and the 'BUT NOT' up to the point when it comes to the timing. This server upgrade was done in July and should have caused problems earlier if this was the technical reason for the incident. So it does not explain why the incident only started on October 10th.



This is the same argument for the second cause about the 'turnover figures calculation' that does well by satisfying all the information sources but does not explain why the smaller outlets are experiencing the incident and the bigger ones do not.

When you look at the fourth technical reason you will notice this cause satisfies all the information in each one of the FOUR dimensions. For dimension number three (Smaller Outlets) we had to make an assumption that all the technicians in the smaller outlets were not aware of this change with the Excel Spread Sheet and therefore the incident occurred with them only.

Assumptions are allowed and are an important element in the thinking at this stage of the CauseWise process. The assumption needs to be plausible and is generally made at this stage either because of lack of understanding or lack of information. The assumption will be one of the first activities to be tackled in our next stage of the analysis, which is the 'verification' stage.

Verify the most probable cause

This is an important stage in the process because this is the difference between theory and reality. Up to this point we've done well by using a structured process to arrive at what we believe to be the most probable cause of the incident. However, this is on paper and we need to verify this thinking with real life, on the job verification. Only when the assumptions and the stated Technical Cause have been verified to be true can we state that we've found the Technical Cause of why the incident occurred.

The team needs to look at the most probable cause(s) identified and set an action plan on how to go about verifying the assumptions and ultimately the Most Probable Cause. The team needs to ask a combination of the following questions;

What would be the cheapest, surest, safest, fastest and least disruptive way to verify each assumption? This exact same question is also repeated for the actual cause itself.

What will be the verification action and who will do it by when?

When doing the verification of a specific assumption and it appears that the assumption is 'not holding water', then that assumption receives an "X" and that particular cause is then eliminated. The converse is also true and that is when the assumption is verified as being true, then we progress to continue verifying the actual Technical Cause itself.

With the right information sources and gathering correct and accurate information, this meeting should last about 20 minutes. We have recorded cases where it only took about 5 minutes to determine the technical reason for an outage and a phone call was all that was needed.

This CauseWise methodology also lends itself to a much quicker and shorter format, which many Major Incident Managers are now using to narrow down and identify the most probable causes for a major incident. The shortened version would normally only look at the "IS" information for the 'Object', 'Fault' and what is 'unique' about that particular incident. In such cases we are working with incomplete information and should be careful not to jump to a conclusion. One way to overcome this is to make absolutely sure that we have the Incident Statement correct.

Identifying Root Cause

Okay, you have solved the incident and fixed it. You feel very good about the effort that produced the answer and as the euphoria declines, you get a call that the exact same incident has occurred again. How is this possible, because you were 100% sure you 'solved' the incident for good! This is because up to this point, we've identified the Technical Cause (how it happened) and not the Root Cause (why the incident occurred).

Have you ever heard the term 'recurring incidents'?

The above is an example of that and we are sure you were on the receiving end of some of these types of incidents, which is annoying, frustrating and time-consuming. The most probable reason for experiencing recurring incidents is because we have not found the Root Cause of that incident yet.

Sometimes the root of an incident is another technical reason. We found this to be true in less than 20% of incidents. Where you have a suspicion that the root of the situation is another technical reason, then we need to do another CauseWise to identify the next technical reason. You need to continue with this until you get to the level where you have reached the root that is embedded in some kind of 'soft issue' or 'company condition that exists' situation. Strictly it is wrong to say that a second technical reason is the root of the situation. The root of any incident eventually lies in some component of a systemic or people component of the incident.

Once the technical reason is identified, we need to do a Root Cause analysis thinking exercise. This exercise is not as rigid as the technical investigation, but it is an investigation in its own right. If you suspect there is another technical reason that caused our first Technical Cause you need to do a stair stepping exercise to check why the first Technical Cause occurred. The following is an example of such an exercise involving a "5 WHY's" questioning drill;

Let's look at the stair stepping method, which is basically utilizing the principle of the "5 WHY's" questioning method. (See diagram below) You start with the technical reason identified and put that on the top of our stairs and then ask; Why did this happen and what was that caused by?

You continue with this until you reached the area where you do not know the answer. Eventually you will end at a spot where you are not sure anymore and that spot, in most cases, will be a possible systemic or people reason for the root of the situation.

A Root Cause is normally some kind of "company condition that exists" and unless removed it will cause continuous future incidents. The following are some examples of past "Root Causes" identified by clients;

1. Documentation – A typical example would be out of date specifications of hardware and software.
2. Policies, Procedures and Processes – A typical example would be inadequate testing procedures (SOP's).
3. Training & Education – Not having the skills to perform a certain task. Not keeping up to date with new developments.
4. Systemic Deficiencies – Typical example would be a developer not aware of coding that could create synchronization issues.
5. Communications/Instructions – Vague and sometimes non-existent communications coupled with confusing instructions.
6. Staff Decisions – Decisions about upgrades, patches, and vendors that are good for one section might not be that good for other sections in the company.
7. Vendor actions and materials – In many situations Vendors do not provide on-site support services.

Referring to our example of the DELL Server issue: we were sure that the person responsible for the upgrade (technician) was under the impression that the upgrade instruction for the system (LAN) did not reach the ADSL users. Something needs to be corrected regarding this situation, otherwise it will happen again.

Company & individual benefits

Any RCA system that would provide a structure that is repeatable and provides guidance on the flow of the thinking approach would lead to benefits all around, providing:

Framework -- The holistic glue that puts all the templates and tools together for RCA in general.

Process -- Having a process for each type of problem indicating where and how to start the investigation.

Template -- Having a template that indicates where to put information so that it makes the most sense.

Structured questions -- Providing the questions that would ensure specific quality data is entered.

Technique -- Indicating the most appropriate information sources and stakeholders to ensure asking the right question of the right person to get the right answer.

Benefits to the organization	*Benefits to the individual*
– More effective problem solving meetings enabling improved productivity – Seamless handover between IM, PM and Change Control ensuring the integrity of data throughout the process – Reduce "open tickets" by at least 80% – Reduced MTR from days to hours and even minutes – Reduction in the level if incidents by at least 63% – Eliminate roll backs	– Information resources only engaged when they've been identified as critical, improving the productivity of staff – Improved levels of self-confidence to solve an incident "first time every time" – Knowing where and how to start by following the template flow providing credibility for investigators

The ultimate exponential benefit

The biggest benefit of all is hidden behind a 'blind spot' for most senior managers. The obvious focus is about the present and the impact an incident has on current operations and the satisfaction of clients and their businesses. However, because of this urgent and serious focus the biggest and most rewarding benefit is overlooked.

We've learned that we need to restore the service interrupted by this incident as quickly and accurately as possible. So, we set out with our analysis and eventually restored the service.

All good so far and we are now ready to investigate the Technical Cause to determine how this happened. Let's say that during our analysis we found that a specific LAN rule was not updated during a specific hardware upgrade and that caused the incident. We are satisfied and we correct the situation without determining why this happened.

This is the exact point where we have overlooked the potential exponential benefits of not taking our analysis a few steps further: the Root Cause of the incident situation. As per our reasoning the Root Cause is the underlying reason or differently stated the company condition that exists that triggered the Technical Cause. Without this Root Cause this incident would never have occurred, right? Right!

As per our definition the Technical Cause in this example was identified as "a LAN rule that was not updated during a hardware upgrade" or so it was thought. When we ask WHY someone did not remember to update the rule, we got the answer that the person responsible was not working that day and there is no other trigger to remind others to update the rule. The lack of a trigger was the Root Cause for the incident that occurred.

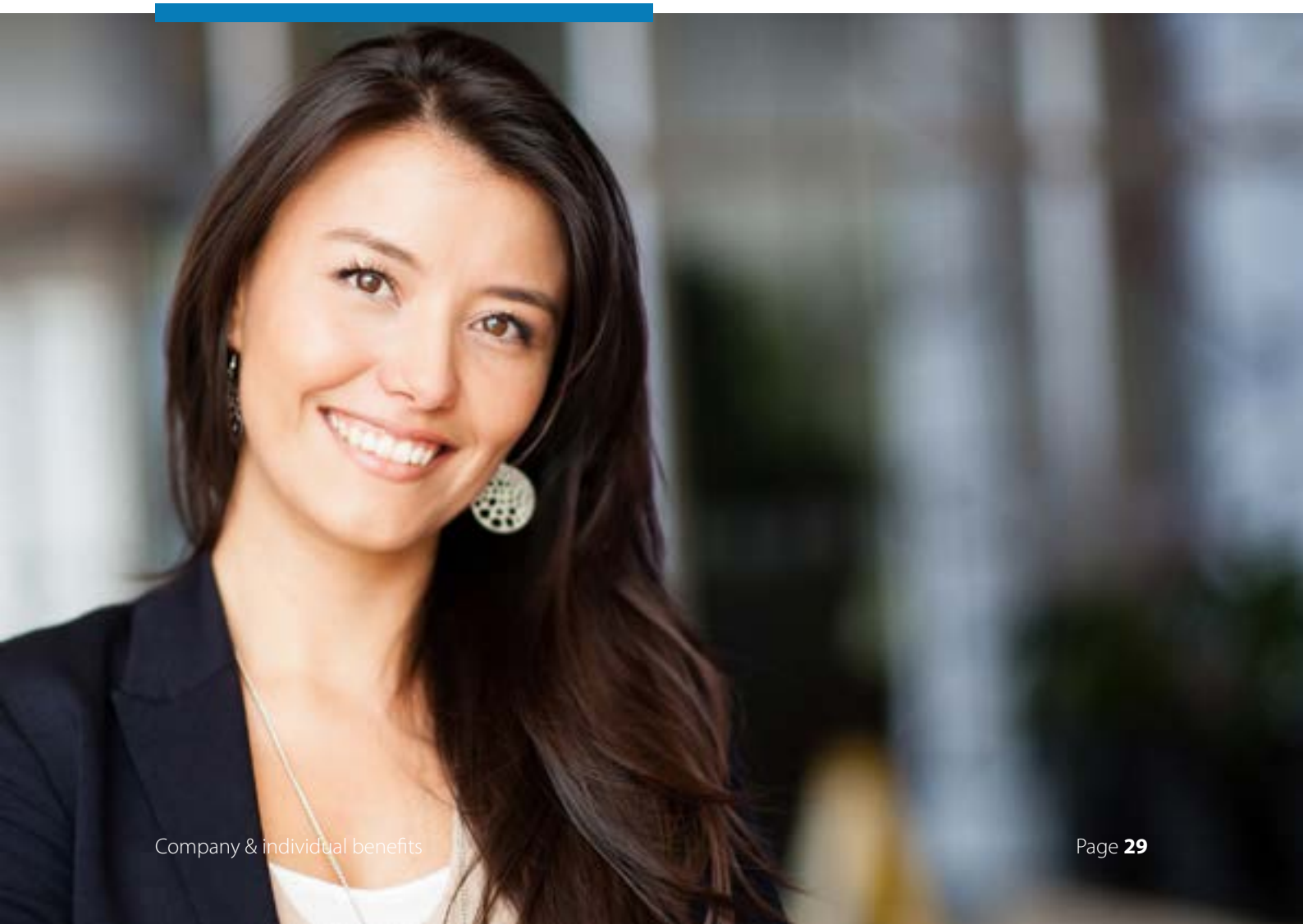
Would you agree that solving the Root Cause will ensure that incident #1 will not reoccur?

Would you agree that the Technical Cause (not updating a LAN rule) could possibly also cause additional incidents in the future?

Would you also agree that if we solved the Root Cause the first time and fixed it permanently (installed a trigger) that we now effectively done the following?

- Have an action in place that would stop a recurrence of the same incident.
- Have a single but same action in place that would effectively avoid additional future incidents.

The situation provides an exponential benefit, which is very likely the single most important and beneficial advantage of educating and training an in-house capability to handle incidents quickly (restoration), accurately (Technical Cause) and permanently (Root Cause).



Summary

To reiterate the observation that it would be highly challenging for an IT team to successfully collaborate on an incident restoration effort bearing in mind all the pressures surrounding the incident situation. Therefore, it would be advisable to provide the team with a tool that would help them to leverage their collective experience and skills to arrive at a restoration, then a Technical Cause and lastly at a Root Cause with speed and accuracy.

No one is born with the problem-solving skills that are needed in highly pressurized IT service incidents and problem situations. We know that each person is exposed to a different upbringing and background providing each person with unique problem-solving skills. However, as we've highlighted in this whitepaper, more than individual sparks of brilliance are needed to solve recurring incidents and any other type of problem-solving situation surrounding incidents.

The good news though that providing and training IT staff to use a process template with structured questions will give each person the opportunity to shine within the process and offer their unique technical know-how and associated problem solving suggestions. That is why it is needed to have a robust system with templates, techniques and structured questions to create a highly intuitive and successful flow that every IT person can follow regardless of their background and upbringing.

Imagine having a team of well-distributed trained problem solving facilitators that can be called upon at any time to facilitate and coach a problem solving team through their own problems and get the kudos from their seniors as it is rightfully deserved.

Acknowledgements

We would like to sincerely thank the lead author of this whitepaper.

Matthys J Fourie, Founder and Chairman at Thinking Dimensions Global Consulting

Matt Fourie is a Professional Problem-Solver specifically in the fields of Root Cause Analysis, Decision Making, and Project Optimization. He co-founded Thinking Dimensions with Chuck Kepner in 1997. Together they developed the KEPNERandFOURIE® thinking methodology. His primary focus is on product design and managing the international network across 20 countries. This is supported by solution design, facilitation and capability development in the areas of Root Cause Analysis, Process Improvement, ITIL Continual Service Improvement, Project Management, Lean-Sigma, and general problem-solving practices.

Matt Fourie holds a B.Mil (B.Sc) from The University of Stellenbosch in Cape Town, a B.Com (Honors) from the University of South Africa, and his Ph.D. from UCL London, UK.

He brings over 30 years of global and Fortune 1000 experience. The main area of his work experience is in addressing client issues with customized programs and transferring this expertise in-house via personal coaching strategies.