SAMPLE

IT – Standard Operating Procedures for Major Incidents and Problem Management

Document History

Revision	Issued	Effective
Original	July 2007	July 2007
1.1		

Approval / Sign off sheet				
Position	**Designate**	**Approved**	**Rejected**	**Comments**
Infrastructure Manager				
CTO				
CIO				
Service Level Manager				

Purpose

The purpose of this document is to define the IT operational procedures associated with problem management, task scheduling, change freeze, and major incident notification and reporting. It is intended to provide a clear, concise description of these management processes as established, along with the associated responsibilities. A brief overview of the various IT Operational Areas is also provided including respective functions, inter-relationships, and contact information.

Scope

The principal goal of this document is to ensure effective and efficient communications, coordination, and decision-making between the IT Operational Areas, Central Services and the business units, particularly regarding problem management, task scheduling, change freezes, and major incident notifications and reporting. Clear communication is essential to provide quality Information Technology services to IT's services and business unit customers.

The nature of IT's business now requires an increased ability to work across areas, generating requirements that, by their nature, are best met at the corporate level. The shift in how information and knowledge are generated, used and managed when coupled with the competition for limited budgets dictates a more strategic

approach to providing information infrastructure services. There are several specific drivers for approaching IT systems more strategically. These include:

- Improving IT infrastructure to meet the overall business Vision and Strategic Plan
- Positioning the IT infrastructure to support corporate applications such as Enterprise Risk Management and Integrated Messaging
- Ensuring availability of integrated services across areas
- Supporting a robust collaborative program and management environment
- Achieving reduced cost of services to IT customers
- Improving security
- Delivering consistent, quality services to IT customers

Fundamental concepts involved in providing and receiving quality IT services include classes of service with varying levels of service delivery and restoration priorities. Additionally, IT services are supported by technical architectures with varying degrees of redundancy and survivability, depending on the class of service. IT operates a service desk, with primary responsibility for classes of service. IT also operates multiple availability management areas and coordinates the escalation and support activities of numerous service providers, carriers and manufacturers. The processes governing these areas are critical to successful operation and management of IT. The procedures and responsibilities defined in this document are applicable to IT Operational Areas and IT customers and refer to all types and classes of IT services.

Definitions

- 8 X 5 - Time period that extends for a typical 8-hour work day Monday through Friday.
- 24 X 7 - Time period that extends for 24 hours each day of the week.
- Change - Refers to any planned operational, maintenance or upgrade action associated with a IT service that has the potential to produce a temporary interruption of service.
- Back-Out Plan - Defines the action required to abort an activity and return to original condition.
- Best Effort - Scheduling of a change at the most appropriate time period so that it has the least impact to services.
- G-24 Hours - A time period of 24 hours prior to Go Live of a major project.
- G-4 Hours - A time period of 4 hours prior to Go Live of a major project
- No Comment Acceptance - If a report has been distributed and the affected IT areas do not respond with questions, comments, or concerns within a five working day period, the report will be considered acceptable.
- Cause (Causal Factor) - An event or condition that results in an effect. Anything that shapes or influences the outcome.
- Proximate Cause(s) - The event(s) that occurred, including any condition(s) that existed immediately before the undesired outcome, directly resulted in its occurrence and, if eliminated or modified, would have prevented the undesired outcome. Also known as the direct cause(s).
- Root Cause(s) - One of multiple factors (events, conditions or organizational factors) that contributed to or created the proximate cause and subsequent undesired outcome and, if eliminated, or modified would have prevented the undesired outcome. Typically, multiple root causes contribute to an undesired outcome.
- Root Cause Analysis (RCA) - A structured evaluation method that identifies the root causes for an undesired outcome and the actions adequate to prevent recurrence. Root cause analysis should continue until organizational factors have been identified, or until data are exhausted.
- Event - A real-time occurrence describing one discrete action, typically an error, failure, or malfunction. Examples: pipe broke, power lost, lightning struck, person opened valve, etc…
- Condition - Any as-found state, whether resulting from an event, that may have safety, health, quality, security, operational, or environmental implications.
- Organizational Factors - Any operational or management structural entity that exerts control over the system at any stage in its life cycle, including but not limited to the system's concept development, design, fabrication, test, maintenance, operation, and disposal. Examples: resource management (budget, staff, training); policy (content, implementation, verification); and management decisions.
- Contributing Factor - An event or condition that may have contributed to the occurrence of an undesired outcome but, if eliminated or modified, would not by itself have prevented the occurrence.
- Barrier - A physical device or an administrative control used to reduce risk of the undesired outcome to an acceptable level. Barriers can provide physical intervention (e.g., a guardrail) or procedural separation in time and space (e.g., lock-out-tag-out procedure).

- Problem - Undiagnosed underlying cause of one or more incidents or potential incidents
- Known Error - Exists after discovering the root cause of a Problem.
- Workaround - Way of preventing or resolving an Incident or Problem. Workarounds can be used to temporarily resolve an issue or steer the user toward another solution. Situations that require expedited action be taken in order to effect restoration of impacted services, or to mitigate a potential service impacting condition.
- TOP - A prearranged gathering of specialist technical support staff from within the IT support organization brought together to focus on specific aspects of IT Availability. Its purpose being to monitor events, real-time as they occur, with the specific aim of identifying improvement opportunities or bottlenecks which exist within the current IT Infrastructure.
- Work request: A defined set of work that has been requested either via standard operating procedures/changes or via a project task or action.

References

IT Service Catalogue
IT Standard Operating Procedures for Communications
IT Standard Operating Procedures for Integrated Messaging
IT Standard Operating Procedures for Capacity Management
IT Standard Operating Procedures for Availability Management
IT Change windows

Quality records

- Major Incident Draft
- Major Incident Notification
- Major Incident Report
- Problem Report
- Infra Incident
- Infra Work Request
- Infra Problem
- Service Management self assessments
- Measurement Metrics

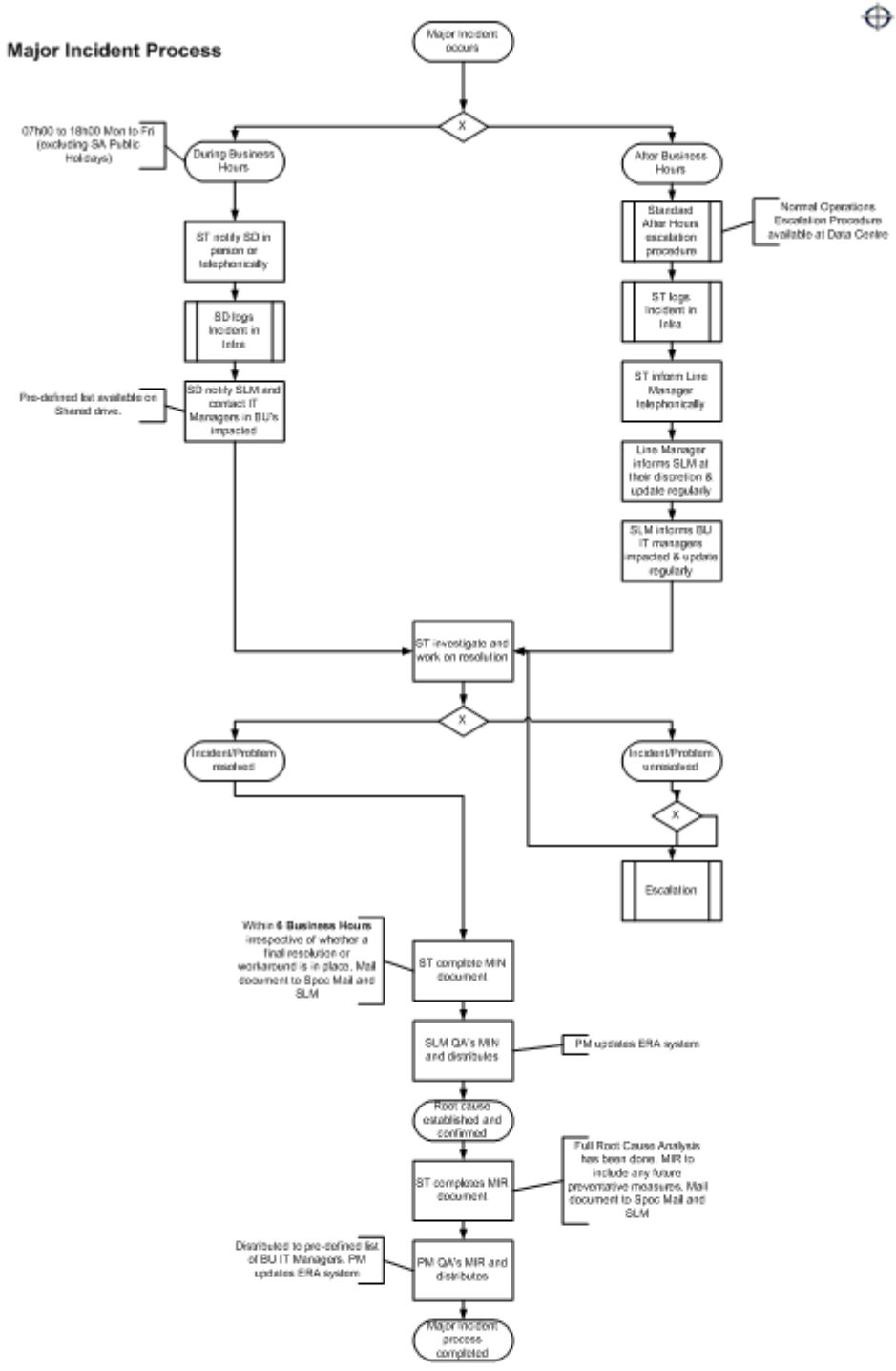
What constitutes a Major incident?

An incident is any event that is not part of the standard operation of a service and that causes, or may cause, an interruption to, or a reduction in, the quality of that service. Incidents are recorded at the service desk and are represented as a database records which are used for documenting and tracking outages and disruptions. A major Incident is and unplanned or temporary interruption of service with severe negative consequences. Examples are outages involving core infrastructure equipment/services that affect a significant customer base, such as isolation of a company site, which is considered a Major Incident. Any equipment or service outage that does not meet criteria necessary to qualify as a Major Incident is by default a Minor Incident.

Major Incident Process

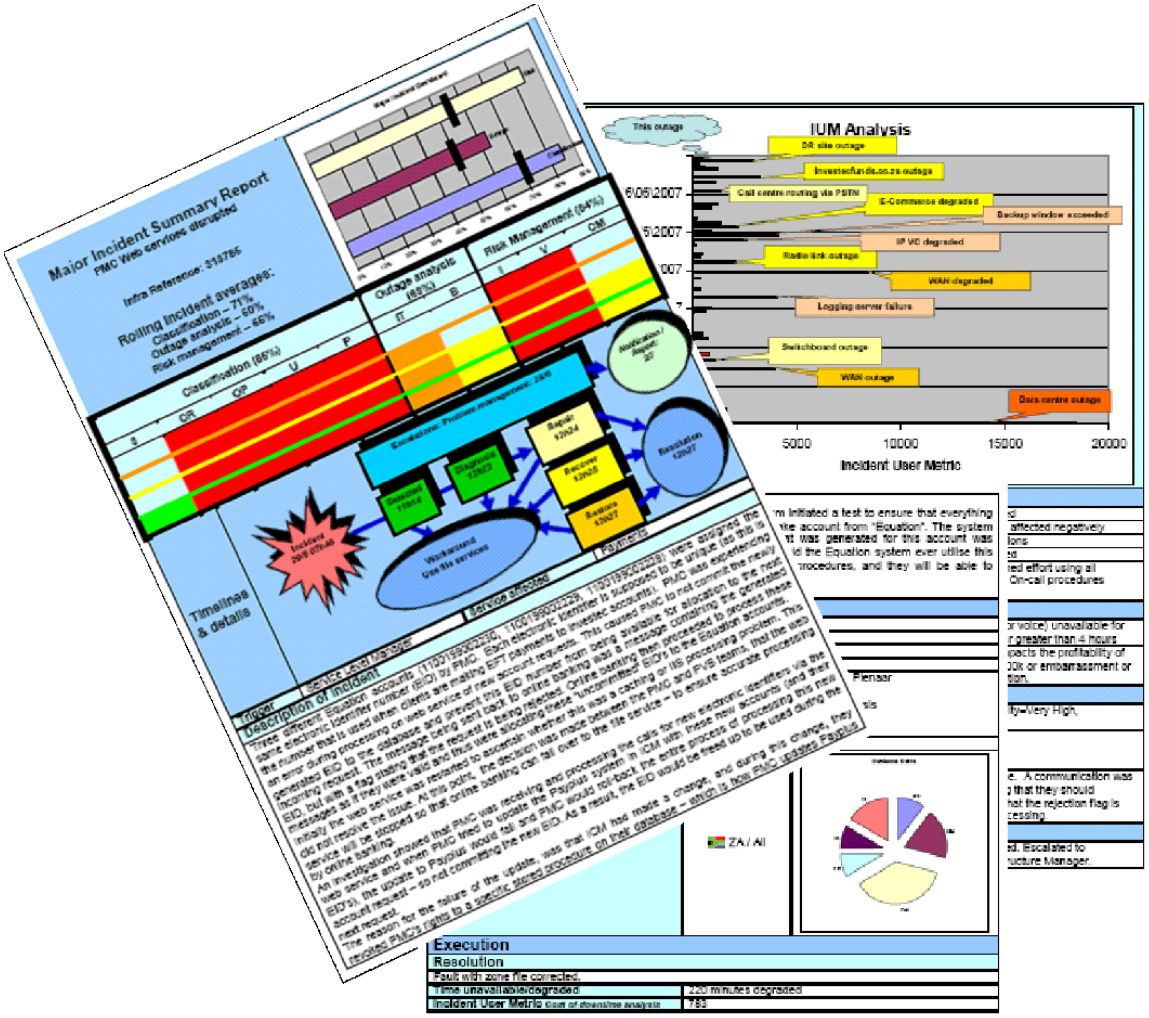
Major Incident Process

Investec
Central IT



Annotations	
ST	Support Technician
SD	Service Desk
SLM	Service Level Manager
BU	Business Unit
QA	Quality Assurer
PM	Problem Manager
MIN	Major Incident Notification
MIR	Major Incident Report

Major Incident Report Summary (MISR)



Recording the details of the incident

The infrastructure engineer records the details by clearly identifying the incident or outage and then describing the business impact and conditions in existence. In the MISR these details are combined into a single details item that also includes the resolution recorded.

Trigger (who requested the report/notification)	
Service affected	
Identification (please clearly describe the incident and its symptoms)	
<Description of the incident or outage and including the symptoms displayed or experienced>	
Business impact (please describe clearly the undesired outcome)	
<Describe how the business was impacted by stating the undesired outcome>	

Conditions (please describe the environment – business or IT - conditions that caused or were present during the incident)
<The business and IT conditions present when the incident or outage occurred>

The infrastructure engineer records the resolution as follows:

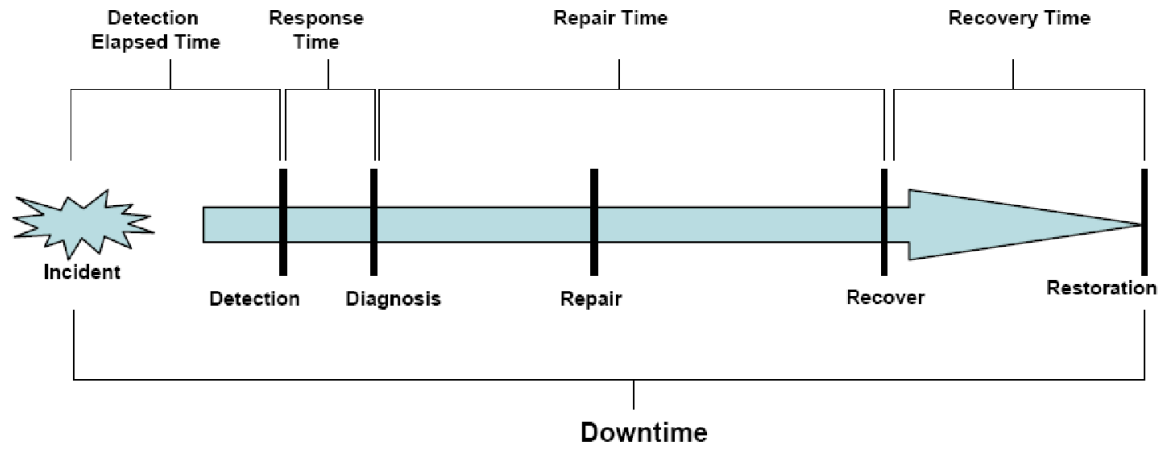
Resolution
Initial (describe the workaround)
<Initial actions and any possible workaround>
Final (describe the steps taken to resume normal operations)
<Steps undertaken to resume normal operations. Include any root cause analysis>

Reporting and notification deadlines

- Notification - issued within 6 working hours of trigger. This includes an email to all affected IT customers and a follow up phone call to the business unit IT representatives.
- Draft - issued within 6 working hours of workaround. This is an interim based upon if a workaround was possible.
- Final - issued within 6 working hours of normal business operations resuming. This includes the MISR being emailed to the business unit IT representatives.

Major Incident Lifecycle

Incident Life Cycle:

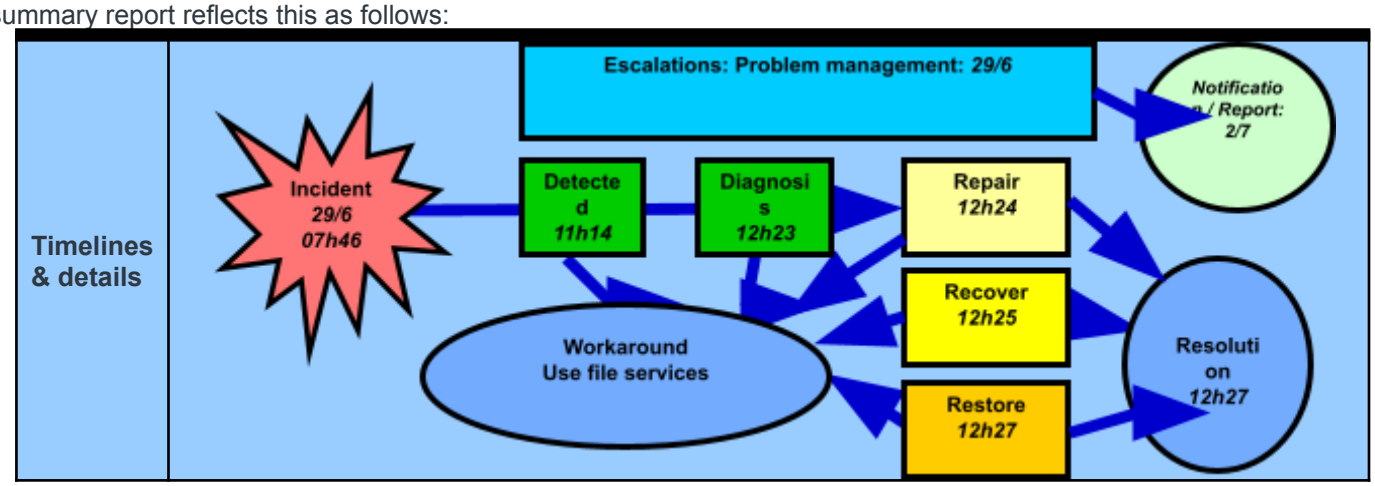


- Occurrence – Something happens to a configuration item (CI).
- Detection – The incident is detected either by monitoring tools, IT personnel or, worse case, the user.
- Diagnosis – the next step is to determine what has happened.
- Repair – Then the CI needs to be corrected. This may be a true solution or a temporary work-around aimed at getting the user back to some degree of productive work.
- Recover – The CI is then put back into production.
- Restore – Finally the service is put back into production.

The DRAFT requires the timelines to be completed as follows:

Timelines (date and times) the expanded incident lifecycle		
Time when incident started (actual - something has happened to a CI)	<dd/mm/yy>	<hh:mm>
Time when incident was detected (incident is detected either by monitoring tools, IT personnel or, worse case, the user)	<dd/mm/yy>	<hh:mm>
Time of diagnosis (underlying cause – we know what happened?)	<dd/mm/yy>	<hh:mm>
Time of repair (failure fixed)	<dd/mm/yy>	<hh:mm>
Time of recovery (component recovered – the CI is back in production)	<dd/mm/yy>	<hh:mm>
Time of restoration (normal operations resume – the service is back in production)	<dd/mm/yy>	<hh:mm>
Time of escalation to problem management team	<dd/mm/yy>	<hh:mm>
Time period service was unavailable (SLA measure)		<minutes>
Time period service was degraded (SLA measure)		<minutes>

The summary report reflects this as follows:



Incident User Metric (IUM)

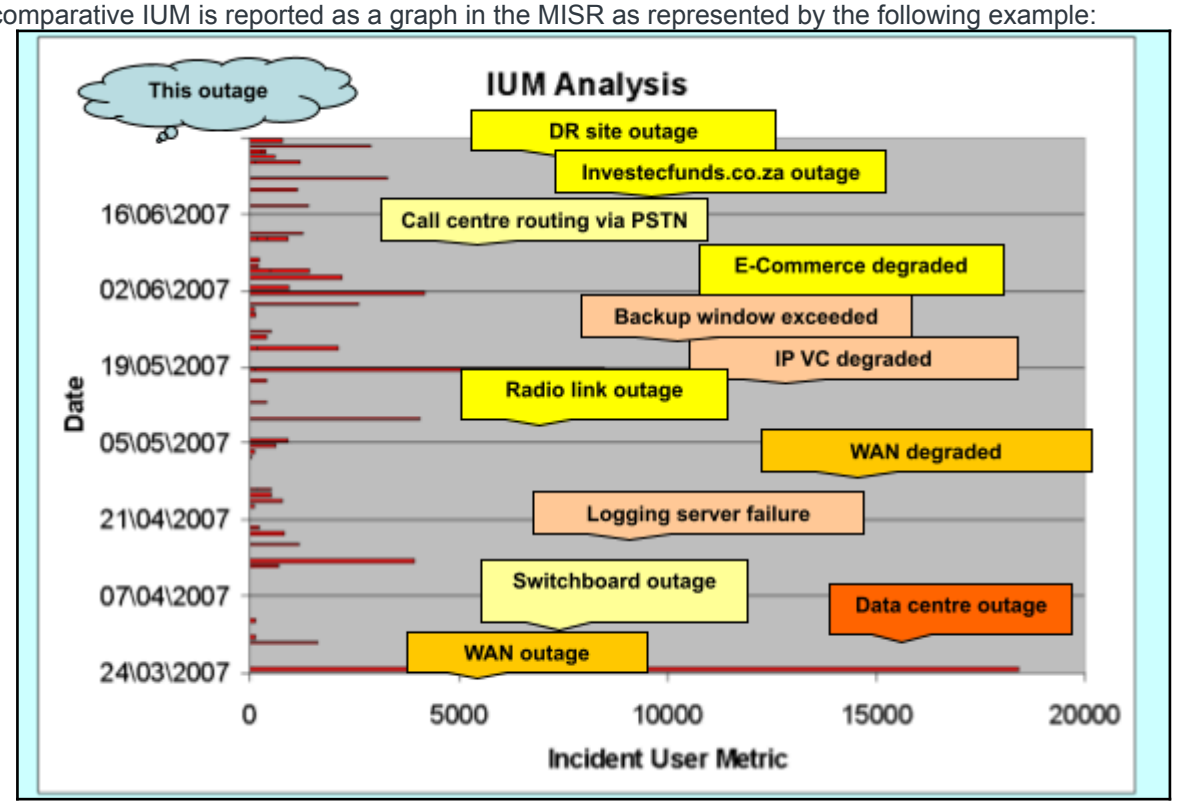
The IUM measurement is used for a relative comparison for the cost of downtime. The infrastructure engineer calculates the Incident User Metric as follows:

Type (mark with a X)						
Outage	**Usage**	**Multiplier**				
Scrutiny by management	Profit	49	Yes		No*	
Effect on productivity*	Staff	48	Yes*		No	
Impact on bank's image	Share price	94	Yes		No*	
Direct financial impact	Assets	91	Yes		No*	
Liability or vulnerability	Nominal	1	Yes		No*	
Limited to internal Central IT process	Budget	2	Yes		No*	
Incident User Metric Cost of Downtime Analysis			X			
Metric is based on Outage type multiplier * outage time in minutes * percentage of users effected. Calculation is also weighted based on degradation at 60% and non-business hours at 50%. Default outage type is "Effect on productivity."						

The multipliers are determined in the following manner:

- Scrutiny by management has profit as a usage for its multiplier. The input is the operating profit per year and is calculated as (388767000 * 14.45) / (12*20*8*60) which equals R48,764.61 per minute. This is rounded to the nearest '000 at 49.
- Effect on productivity has staff as a usage for its multiplier. The input is the staff compensation % of Operating Income per year and is calculated as (40.1% * R(964555000 * 14.45)) / (12*20*8*60) which equals R48,516.19 per minute. This is rounded to the nearest '000 as 48.
- Impact on bank's image has share price as a usage for its multiplier. The input is the share price and is calculated as Average (Price * volume) per day over past year which equals R94,616.54 per minute. This is rounded to the nearest '000 as 94.
- Direct financial impact has assets as a usage for its multiplier. The input is the operating income per year and is calculated as R(964555000 * 14.45) / (16*20*8*60) which equals R90,741.01 per minute. This is rounded to the nearest '000 as 91.
- Liability and vulnerability has a nominal amount as a usage for its multiplier. This is determined to be 1.
- Central IT process has budget as a usage for its multipler. The input is the Central IT Budget per year and is calculated as R160000000 / (12*20*8*60) which equals R1388.88 per minute. This is rounded to the nearest '000 as 2.

The comparative IUM is reported as a graph in the MISR as represented by the following example:



Classification

The infrastructure engineer captures the classification information as follows:

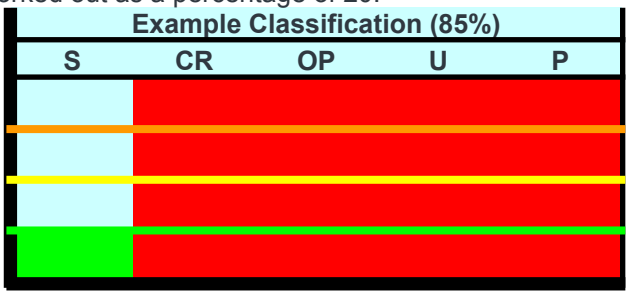
Scope (Mark with a X) Dashboard designation = S	
More than 50% of customers affected	
More than 25% of customers affected	
Less than 25% of customers affected*	
< 1 % of users affected	
Credibility (Mark with a X) Dashboard designation = CR	
Areas outside the company will be affected negatively	
Company affected negatively	

Business unit affected negatively	
No credibility issue*	
Operations (Mark with a X) Dashboard designation = OP	
Interferes with core business functions	
Interferes with business activities*	
Interferes with normal completion of work	
No work interference	
Urgency (Mark with a X) Dashboard designation = U	
Underway and could not be stopped	
Possible to be rescheduled	
Possible to postpone	
Completion time not important*	
Prioritization (Mark with a X) Dashboard designation = P	
Reviewing the scope , credibility, operations and urgency please classify the priority of the incident	
Critical - An immediate and sustained effort using all available resources until resolved. On-call procedures activated, vendor support invoked.	
High - Technicians respond immediately, assess the situation, and may interrupt other staff working low or medium priority jobs for assistance.	
Medium - Respond using standard procedures and operating within normal supervisory management structures.	
Low – Respond using standard operating procedures as time allows. *	

The classifications ratings are then aggregated into a dashboard that is used in the MISR. The dash board calculation is performed by assigning the ratings a value:

- Red = 4
- Orange = 3
- Yellow = 2
- Green = 1

The maximum score is 20 so the classification dash board value is the combined aggregated ratings score worked out as a percentage of 20.



Outage

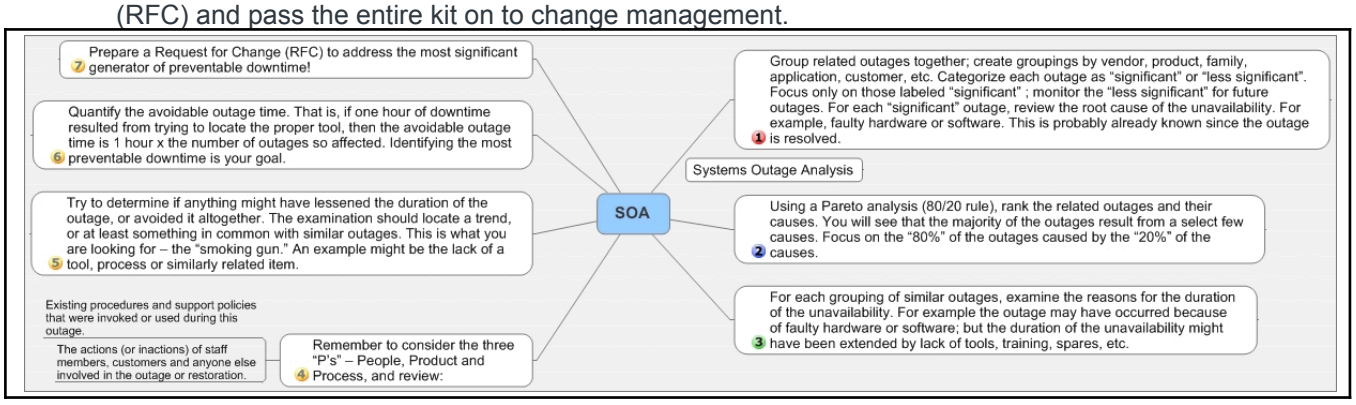
Systems outage analysis (SOA)

SOA is a method to improve availability. Presented as an availability management process tool or technique, SOA is a powerful management tool to improve quality. The reason to use SOA is to identify the causes of outages and thus reduce the frequency and duration of outages. SOA aims to improve mean-time-to-repair (MTTR). The result of a SOA is clear understanding of what happened to cause an outage, and exposes the risk of future outages due to the same cause or causes.

Performing a SOA is straight forward. Working with problem management and IT customers, you examine past outages to identify configuration items (CI) (products, people, or process) related to an outage. In effect, you simply review the impact to the organization and infrastructure as reflected by how the organization responded

to an outage. This is different from proactive problem management since availability management has a scope that includes the organization (people, process, training, staffing, etc.)
To get going, collect outage data in the form of incidents, any related closed problems, or known errors. Gather together a team familiar with the outages, the infrastructure, processes, procedures, people, and so on. Be sure to include a customer representative and perhaps some users on the team as well (their input will be critical in guiding the team through the SOA process). SOA involves the following steps:

- Group related outages together by vendor, product, family, application, customer, etc. Then, using customer and user input as appropriate, categorize each outage as “significant” or “less significant." Focus only on those labelled “significant,” and monitor the “less significant” for future outages. For each outage tagged as “significant” review the root cause of the unavailability (this requires closed incidents and problems.) For example, faulty hardware or software. This is probably already known since the outage is resolved.
- Perform a simple Pareto analysis to break the significant issues into a smaller group. The Using the Pareto 80/20 rule you can rank the related outages and their causes. You will find that the majority (80%) of the outages result from a select few causes (20% of the organization or infrastructure.) Of course, you want to focus on the 80% of the outages caused by the 20% of the causes.
- For each grouping of similar outages, examine the reasons for the duration of the unavailability. For example, the outage may have occurred because of faulty hardware or software, but the duration of the unavailability might have been extended by lack of tools, little or no training, unavailable spares, etc. Remember to consider the “3 P’s” – people, product and process. Then review:
    - All existing procedures and policies used during the outage.
    - The actions and inactions of staff members, customers and anyone else involved in the outage or its restoration.
    - The management directives given to all involved during the before and during the outage.
- You must determine if anything might have lessened the duration of the outage, or better yet, avoided it altogether. Your examination of the “3 P’s” should locate a trend, a related cause, or at something in common with similar outages. This is the smoking gun. For example, a common cause might extend an outage may be a hierarchical escalation requirement that does not allow staff to proceed without management approval or a special tool is required and could not be found.
- The next step is to quantify the avoidable outage time. That is, if one hour of downtime resulted from trying to locate the proper tool, then the avoidable outage time is one hour times the number of outages so affected. Identifying the most preventable downtime is your goal. This is then the most significant generator of preventable downtime.
- End the SOA by creating a report summarizing the number of outages analyzed timeframe, avoidable outage time, and the suggestions for improving or avoiding the outage. Prepare a request for change (RFC) and pass the entire kit on to change management.



The MSIR employs a “lite” version of SOA. The infrastructure engineer captures the outage information as follows:

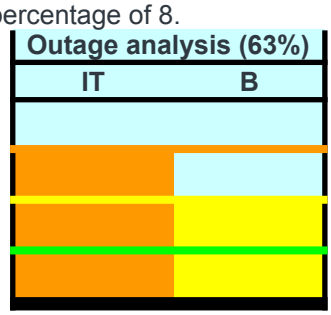
IT Service outage classification (Mark with a X) Dashboard designation = IT	
Critical - App, server, link (network or voice) unavailable for greater than 4 hours or degraded for greater than 1 day	
Major - App, server, link (network or voice) unavailable for greater than 1 hour or degraded for greater than 4 hours	

Moderate - App, server, link (network or voice) unavailable for greater than 30 minutes or degraded for greater than 1 hour	
Minor - App, server, link (network or voice) unavailable greater than 5 minutes or degraded for greater than 30 minutes	
Low* - App, server, link (network or voice) unavailable for less than 5 minutes or degraded for less than 30 minutes	
Business Service outage classification (Mark with a X) Dashboard designation = B	
Critical - Financial loss, which puts a business unit in a critical position - greater than R50m or substantial loss of credibility or litigation or prosecution or fatality or disability.	
Major - Financial loss which severely impacts the profitability of a business unit - greater than R5m or serious loss of credibility or sanction or impairment	
Moderate - Financial loss which impacts the profitability of the business unit, greater than R500k or embarrassment or reported to regulator or hospitalization.	
Minor -Financial loss with a visible impact on profitability but no real effect, greater than R50k or some embarrassment or rule or process breaches or medical treatment	
Low* - Financial loss with no real effect, less than R50k or irritating or no legal or regulatory issue or no medical treatment.	

The outage ratings are then aggregated into a dashboard that is used in the MISR. The dash board calculation is performed by assigning the ratings a value:

- Red = 4
- Orange = 3
- Yellow = 2
- Green = 1

The maximum score is 8 so the outage dash board value is the combined aggregated ratings score worked out as a percentage of 8.



Risk management

CRAMM

ITIL promotes the *CCTA Risk Analysis and Management Method* (CRAMM) for risk assessment. CRAMM is simply a process template for analyzing risks (*threats* an *asset* faces due to *vulnerabilities*) and then managing those risks through countermeasures. CRAMM provides a framework to calculate risk from asset values and vulnerabilities, referred to as Risk Analysis. The framework also helps you avoid, reduce, or choose to accept these risks, referred to as Risk Management. By analyzing assets one can realize the potential damage caused by a failure in Confidentiality (unauthorized disclosure), Integrity (unauthorized modification or misuse) or Availability (destruction or loss). CRAMM assumes that it is cost prohibitive to eliminate risk; but that you can cost effectively mitigate risk by structured analysis of assets.

CRAMM uses the following format:

- Uses meetings, interviews, and questionnaires for data collection.
- Identifies and categorizes IT assets into one of three categories: 1) data, 2) application/software 3) physical assets (equipment, buildings, staff, etc.)

- Requires the assessor to consider the Impact of the loss of *Confidentiality, Integrity and Availability* (CIA) of the asset.
- Expresses Vulnerability (the likelihood that a threat may occur) as: very high, high, medium, low or very low.
- Expresses Risk (the likelihood that a threat could exploit the Vulnerability) as: high, medium or low.

The three stages of CRAMM are:

1. Asset identification and valuation. The goal here is to identify and value assets.
2. Threat and vulnerability assessment. The goal here is to assess the CIA risks to assets.
3. Countermeasure selection and recommendation. The goal here is to identify the changes required to manage the CIA risks identified.

Asset Owner: CRAMM Spreadsheet Example
Asset: Credit Card Data

	CONFIDENTIALITY public (0), restricted (1-5), confidential (6-9), secure (10)			**INTEGRITY** low (1-3), moderate (4-7), high (8-9), very high (10)		**AVAILABILITY** low (1-3), moderate (4-6), high (7-8), very high (9), mandatory (10)	
Impact Requirement (1-10)	10 / secure			10 / very high		8 / high	
Threats *list all that apply*	**Disclosure**	**Theft**	**Loss**	**Hacking**	**input errors**	**Drive failure**	**Power Failure**
Vulnerability (1-10) *none (0), low (1-4), moderate (5-7), high (8-9), very high (10)*	10	3	1	8	2	5	2
Threat (1 to 100) *Impact X Vulnerability*	100	30	10	80	20	40	16
Risk Level *Low (1-33), Medium (34-67), High (68-100)*	High	Low	Low	High	Low	Medium	Low
Countermeasures *list all that apply*	Password protection			Firewall	Data input forms Data validation		

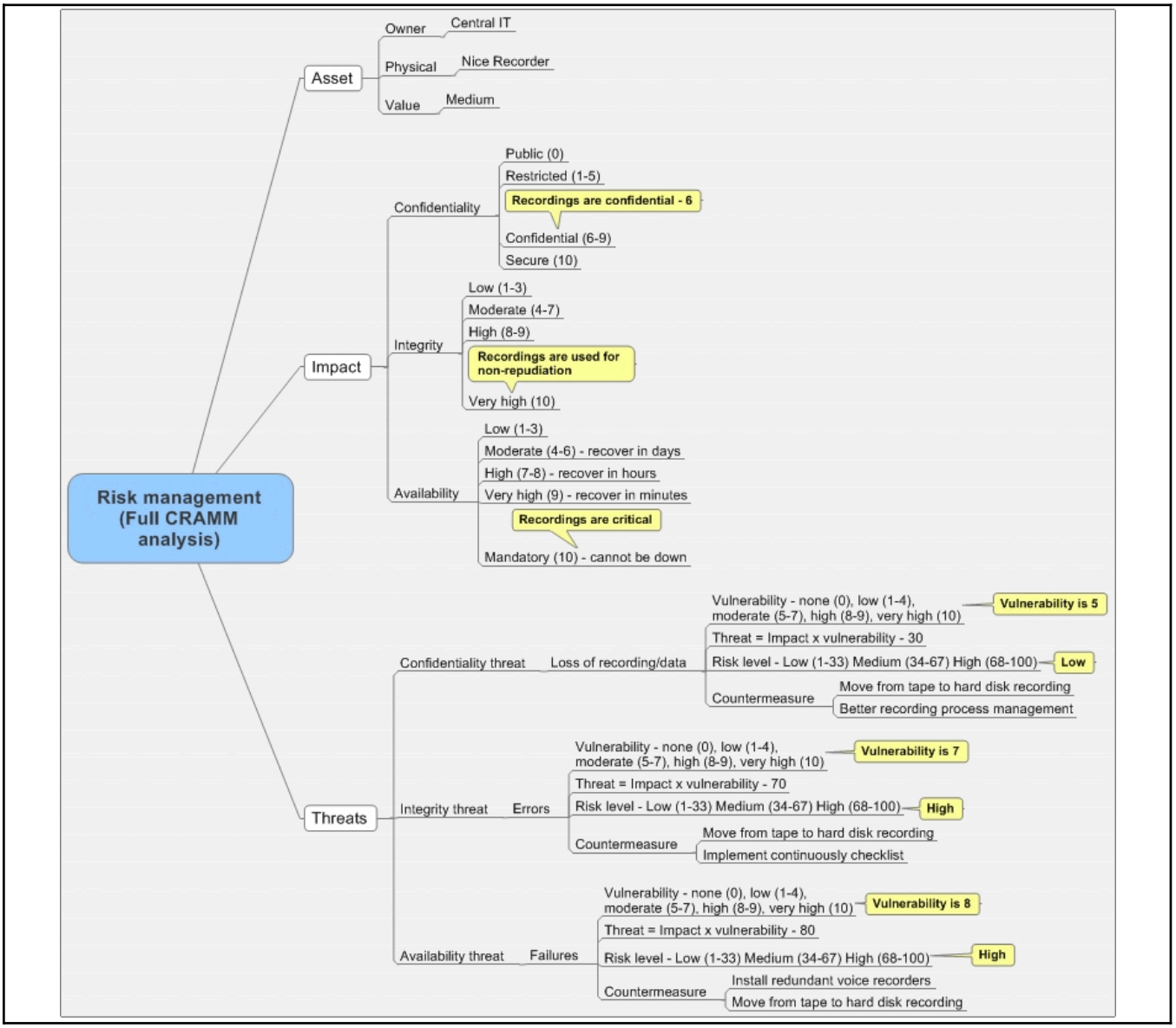
10-step plan for CRAMM

1. Gain (or grant) authority for the security review to proceed.
2. Define the scope of the review (IT service, location, application, etc.,) assign a team, and identify sources of information. Draft, review, and agree a control document.
3. Begin stage 1 -- identify assets within the scope of the review (data, application/software and physical assets.) A *Configuration Management Database* (CMDB) is valuable here, if you do not have a CMDB then ask around about important data, software or physical assets. You can leverage previous *Business Impact Analysis* (BIA) work that you have done here as well.
4. Prepare a grid or table. A spreadsheet works well for this. (See above.) For each asset, list the asset and the asset "owner" -- the owner is the person who knows best, the usage and value of this asset. This process also raises the level of acceptance of your review findings and proposals.
5. Interview the asset owner; have the asset owner value data and software assets by the impact/cost resulting from loss of Confidentiality, Integrity, or Availability; and physical assets by replacement cost. Have data owners consider the impact of the following attributes of their asset:
    a. Confidentiality -- impact of or sensitivity to disclosure of the asset to non-authorized parties, e.g., "employees," "contractors," etc. Use confidentiality requirement categories of **public** (0), **restricted** (1-5), **confidential** (6-9), and **secure** (10).
    b. Integrity -- impact of unknown or unauthorized modification e.g. "data input errors," etc. Use integrity requirement categories of **low** (1-3), **moderate** (4-7), **high** (8-9), and **very high** (10).
    c. Availability -- impact of the asset being unavailable for various time frames, e.g., "less than 15 minutes", "1 hour", "1 day", etc. Use availability requirement categories of **low** (1-3), **moderate** [OK to recover in days] (4-6), **high** [hours] (7-8), **very high** [minutes] (9), and **mandatory** [cannot be down] (10).

    For a, and c above, have the data owner first choose a category for each, then a value within the category. For example, for Integrity, have them choose first from *low, moderate, high* and *very high*. Then, if they chose *moderate* in this case, ask them to rank the impact on a scale of 4 to 7.

If there are existing measures already in place to control risks identify them during this stage. Update the grid and move to stage 2.

6. Begin stage 2 -- review and agree deliverables from Stage 1. Determine how likely each risk in stage 1 is by asking questions of support personnel, experts and other personnel using prepared questionnaires to try and asses the likelihood that the identified risks could actually occur. Consider Hackers (inside and outside the company), Viruses, Failures (hardware and software), Disasters (terrorism and natural), and People, process or procedure errors. Create a column for each threat. Use a categories of **none** (0), **low** (1-4), **moderate** (5-7), **high** (8-9), and **very high** (10). Update the spreadsheet for each asset.
7. Calculate the Risk entry by multiplying the Impact by the Vulnerability. Based on the risk score (impact X vulnerability) assign a label of **low** (1-33), **medium** (34-67), or **high** (68-100). Update the spreadsheet for each asset. You now have an agreed list of the most vulnerable areas! Since this list is developed with and agreed by the Business, you also have a powerful ally for justifying the need to proceed.
8. Begin stage 3 -- review and agree deliverables from Stage 2. Begin to identify and select countermeasures for those assets with the highest risk level.
9. Consider countermeasures and ways to mitigate the threats. Focus on the higher-level threats first, but don't overlook quick, easy or cheap fixes to lower level threats. In Figure 1, notice that we also chose to implement some countermeasures for low level as well as high level threats. Give precedence to those countermeasures that:
    a. protect against several threats
    b. protect high risk assets
    c. apply where there are no countermeasures already in use
    d. are less expensive to implement
    e. are more effective at preventing or mitigating threats
    f. prevent threats rather than detecting or facilitating recovery
    g. can be implemented quickly, easily and inexpensively (even for low risk)
10. Raise an RFC for the highest-level threats; use your assessments as justification for the Change. Produce a schedule and plan for implementation of agreed countermeasure recommendations.



The Major Incident Report DRAFT on risk management requires Infrastructure engineers to follow a scaled down, “lite”, version of CRAMM as follows:

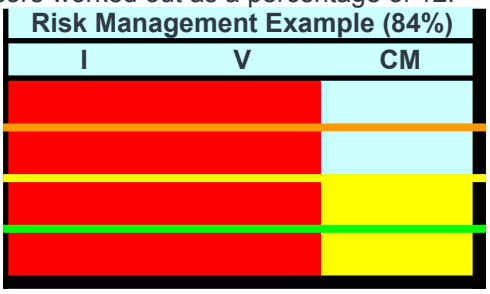
Risk impact (Mark with a X) Dashboard designation = I		
Evaluate the data and information that is directly effected by the incident		
Confidentiality (Information is accessible only to those authorized)	Secure	
	Confidential	
	Restricted*	
	Public	
Integrity (Safeguarding the accuracy and completeness of information)	Very high	
	High	
	Moderate*	
	Low	
Availability (Authorised users have access to information when required.)	Mandatory	
	Very high	
	High	
	Moderate*	

		Low		
Rating Taking into account the above please rate the Risk impact	Critical	Major	Moderate	Low
Risk vulnerability (Rate as either low, moderate, high or major) Dashboard designation = V				
Rate the vulnerability in the following categories of the information or data that is affected by the incident				
Loss		<low, moderate, high, major>		
Error		<low, moderate, high, major>		
Failure*		<low, moderate, high, major>		
Rating Taking into account the above please rate the Risk vulnerability	Critical	Major	Moderate	Low
Countermeasures Dashboard designation = CM				
What measures are in place to mitigate any risks identified with the information or data affected by the incident				
<Due diligence>				
Rating Taking into account the above please rate the Risk Countermeasures	Critical	Major	Moderate	Low

The risk ratings are then aggregated into a dashboard that is used in the MISR. The dash board calculation is performed by assigning the ratings a value:

- Critical = 4
- Major = 3
- Moderate = 2
- Low = 1

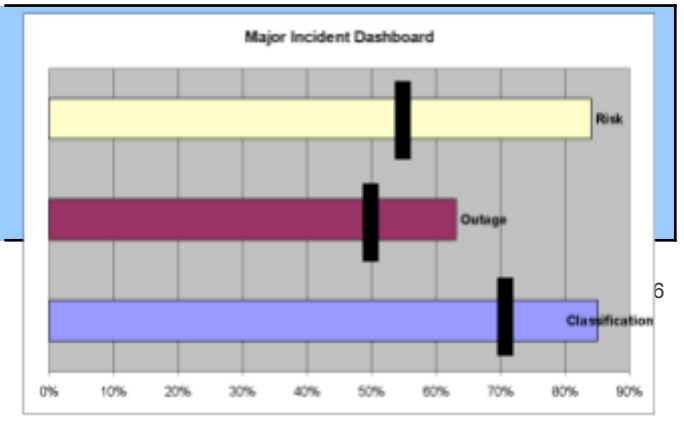
The maximum score is 12 so the risk management dash board value is that combined aggregated ratings score worked out as a percentage of 12.



The MISR Incident dashboard

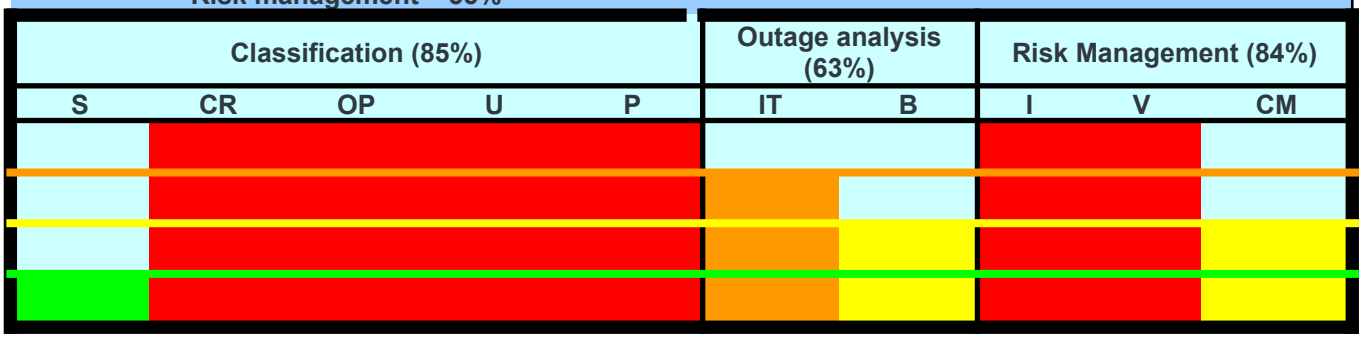
The classification, outage and risk management dashboard are combined into a single MISR dashboard. In addition the rolling incident average is calculated and represented. The following is an example incident MISR dashboard.

Major Incident Summary Report
PMC Web services disrupted



Infra Reference: 313785

Rolling Incident averages:
Classification – 71%
Outage analysis – 50%
Risk management – 55%



Problem management

An incident is an event which is not part of the standard operation of a service and which causes or may cause disruption to or a reduction in the quality of services and customer productivity. An incident might give rise to the identification and investigation of a problem, but never become a problem. Even if handed over to the Problem management process for second line incident control, it remains an incident. Problem management might, however, manage the resolution of the incident and problem in tandem, for instance if the incident can only be closed by resolution of the problem.
A problem is the unknown root cause of one or more existing or potential Incidents. Problems may sometimes be identified because of multiple incidents that exhibit common symptoms. Problems can also be identified from a single significant incident, indicative of a single error, for which the cause is unknown. Occasionally problems will be identified well before any related incidents occur.

Prioritization

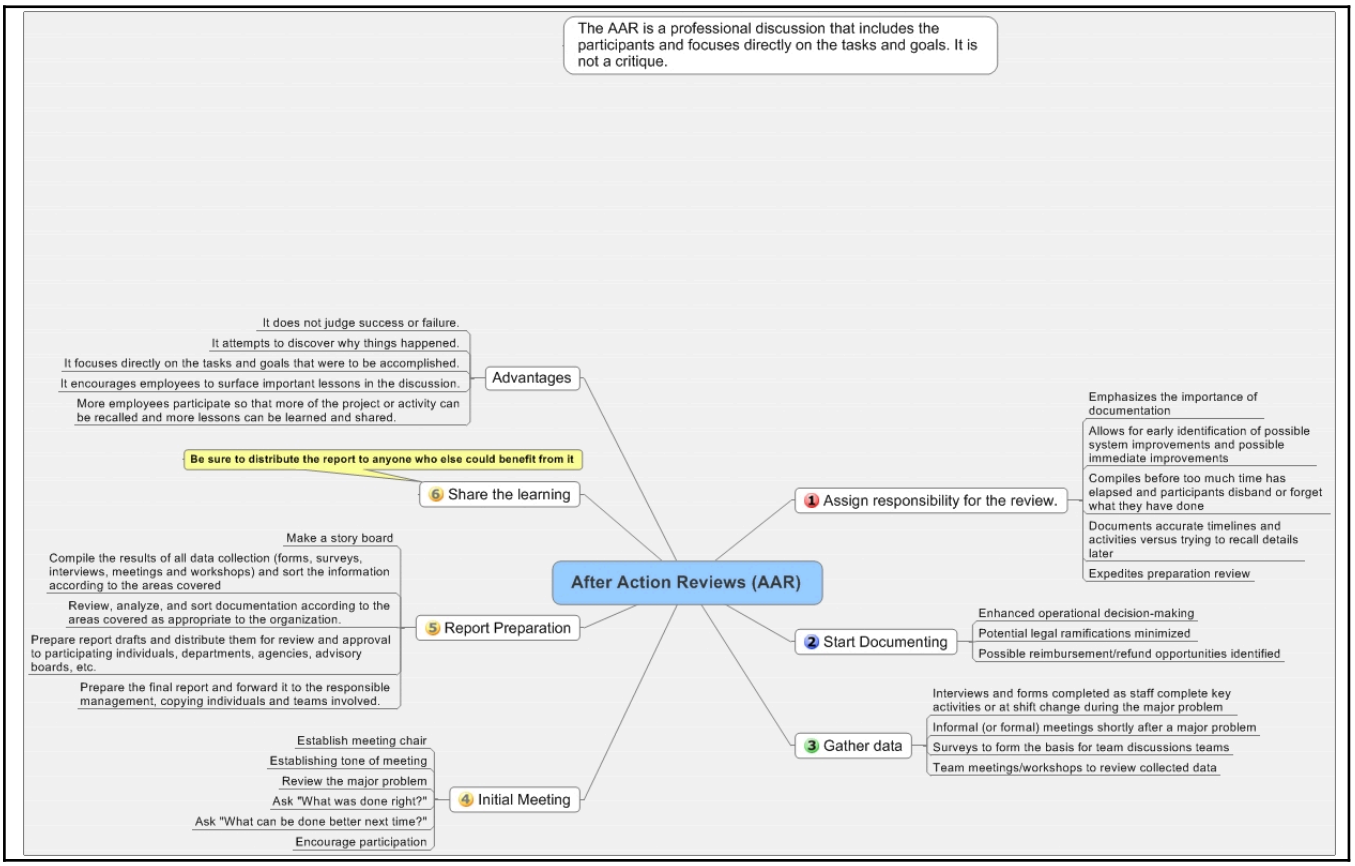
The infrastructure team implements a prioritization meeting on a weekly basis at 08:15am on a Friday as part of the problem management process.
Each Team leader fills in the form below and brings it along to the Infrastructure prioritization meeting on Friday's at 8:15am. The team leader evaluates incidents, problems and work requests by importance in his area and rates them in each category in a priority of 1 to 5. Of these 15 items, 10 are selected and rated in priority 1 to 10. These 10 items are discussed at the meeting were an overall single priority list is created which is also rated 1 to 10. This last priority list is escalated to IT management.
The Infrastructure team leader completes the form as follows:

Top 5 Incidents *This must include as a minimum all major incidents that have occurred during the past week*				
Priority	**Infra Reference**	**Description**	**Responsibility/Status**	**% completed**
1				
2				
3				
4				
5				
Top 5 Problems *These are problems that have been logged as **RED** Infra problems.*				
Priority	**Infra Reference**	**Description**	**Responsibility/Status**	**% completed**
1				
2				
3				
4				

5				
Top 5 Work requests ***These must be evaluated based upon the workload generated and the profile (management view) of the request***				
Priority	**Infra Reference**	**Description**	**Responsibility/Status**	**% completed**
1				
2				
3				
4				
5				
Consolidated Top 10 for individual Team				
Priority	**Infra Reference**	**Description**	**Responsibility/Status**	**% completed**
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

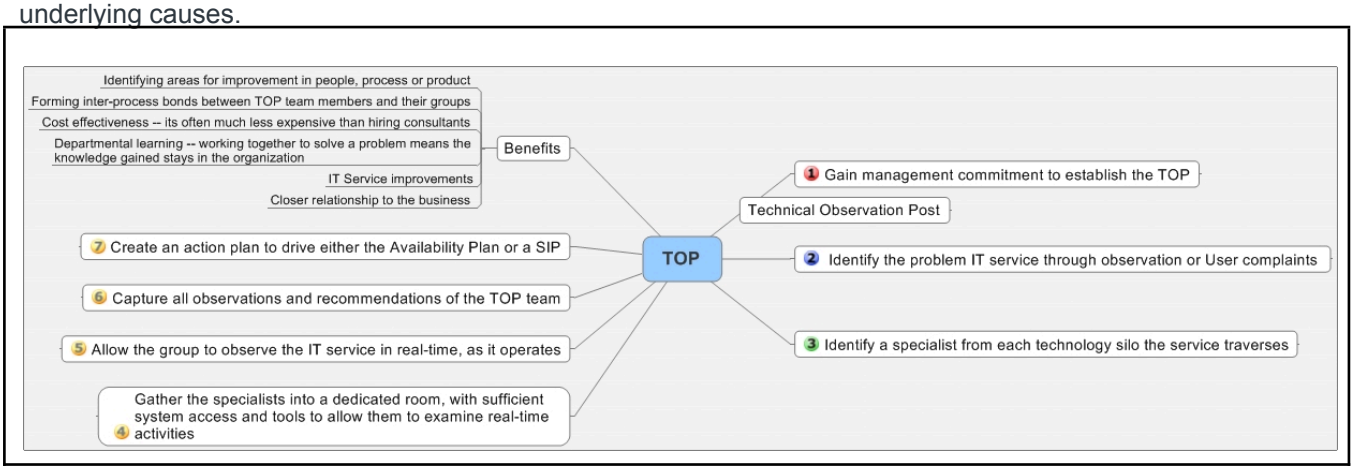
After Action Review

An After Action Review (AAR) is conducted when a change results in a Major Incident. This is also known as a Major Incident Post-mortem. The review is conducted and scheduled immediately after the change forum meeting.

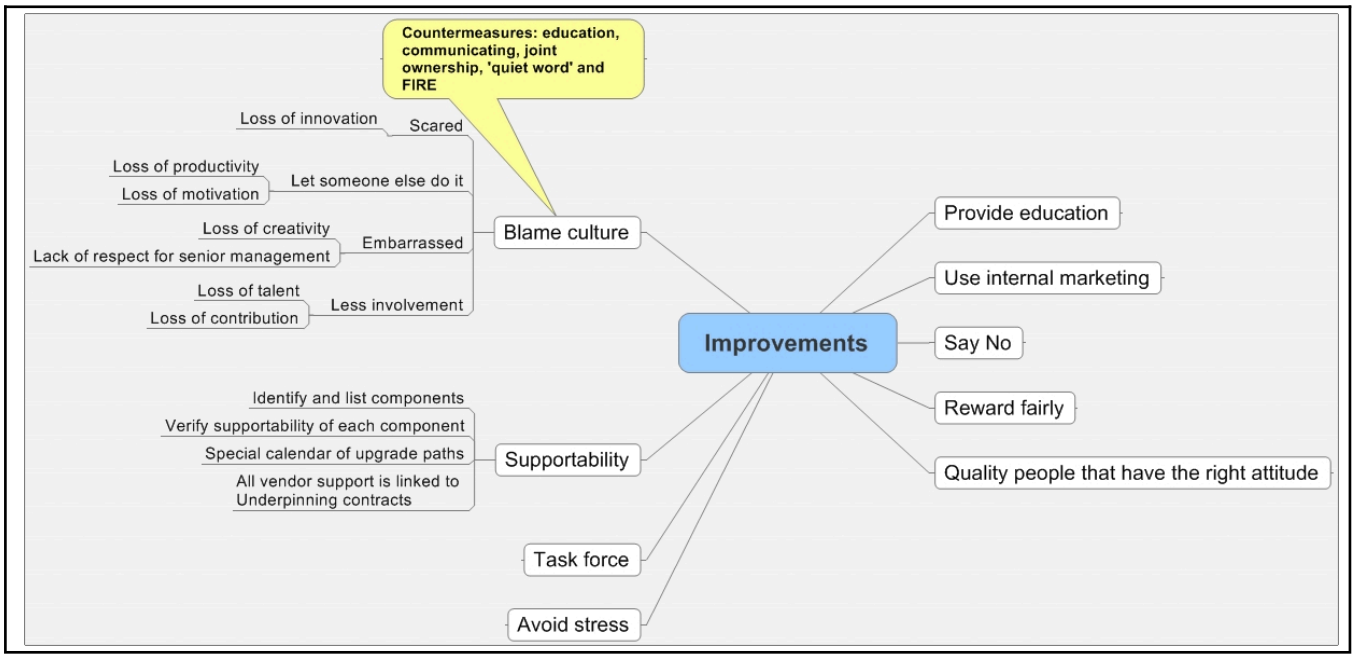


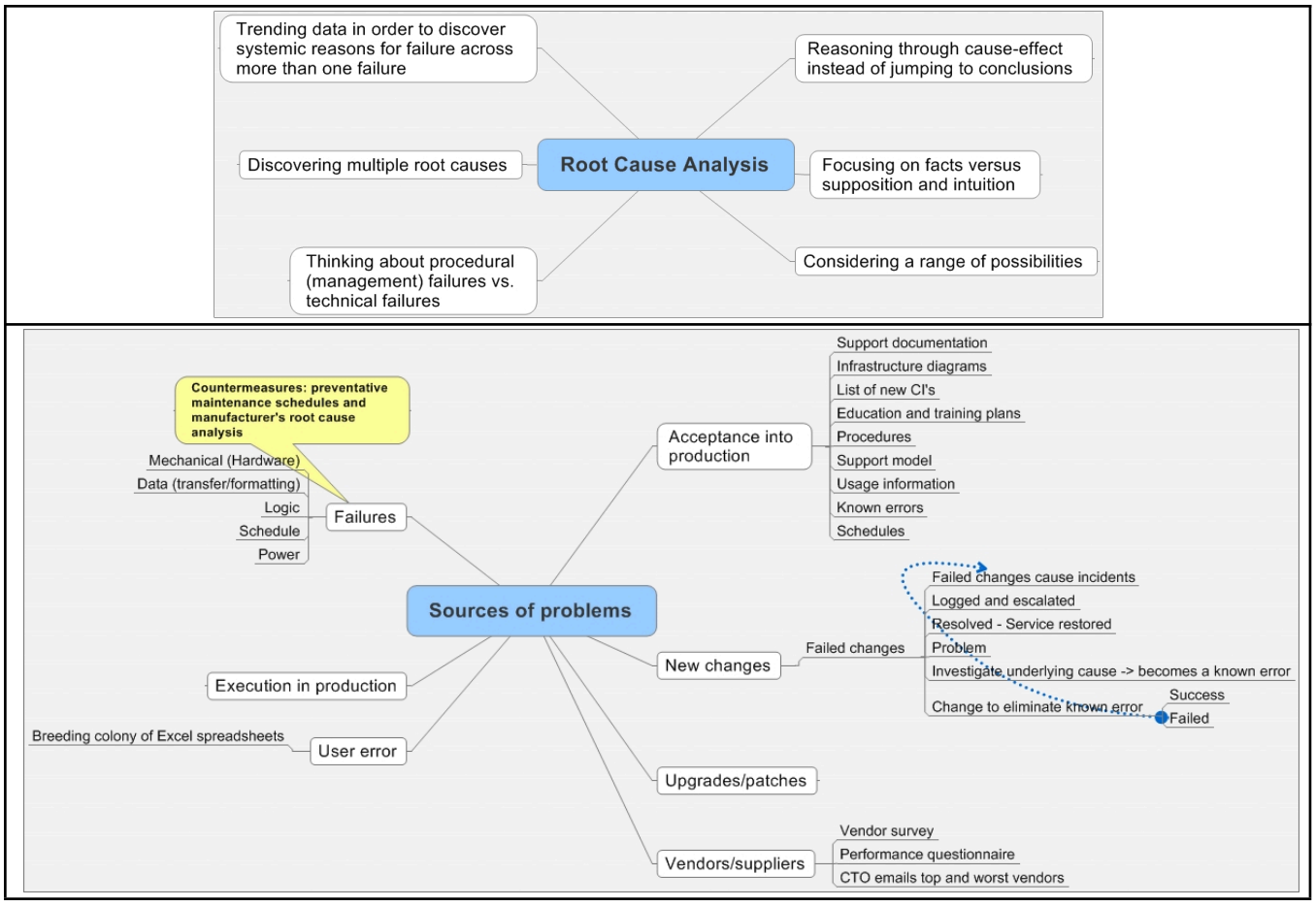
Technical Observation Post (TOP)

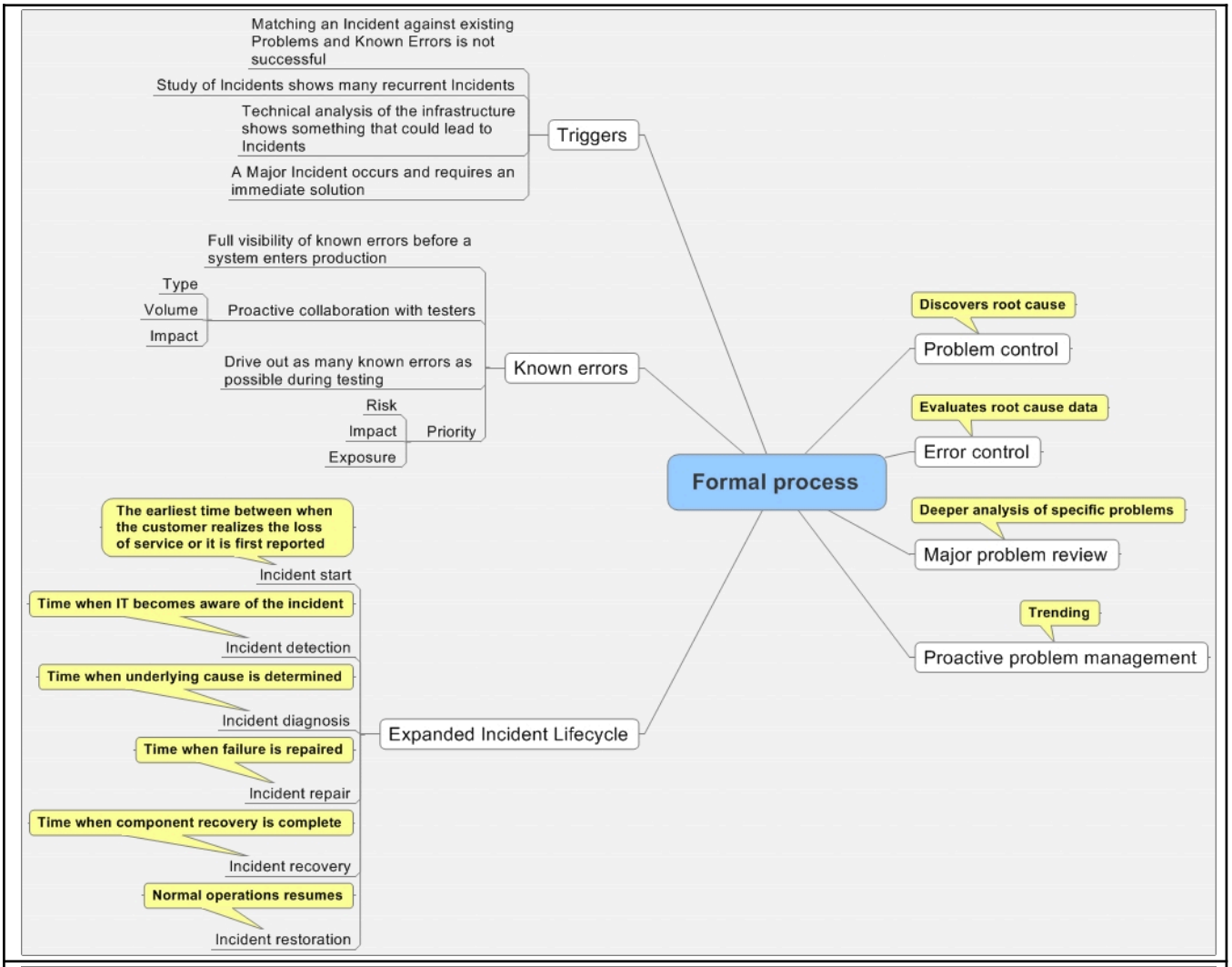
A TOP is constituted when a frequent number of Major incidents occur that are related to the same or similar underlying causes.

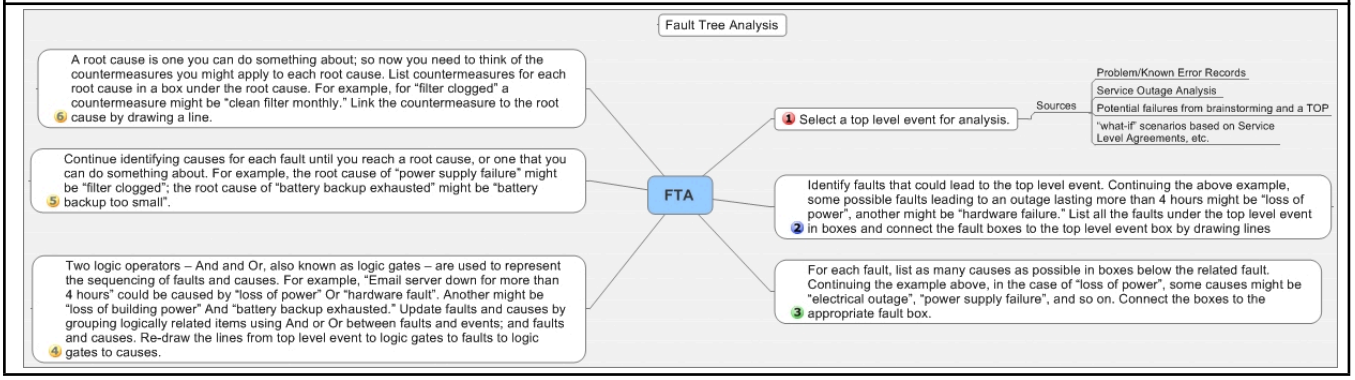


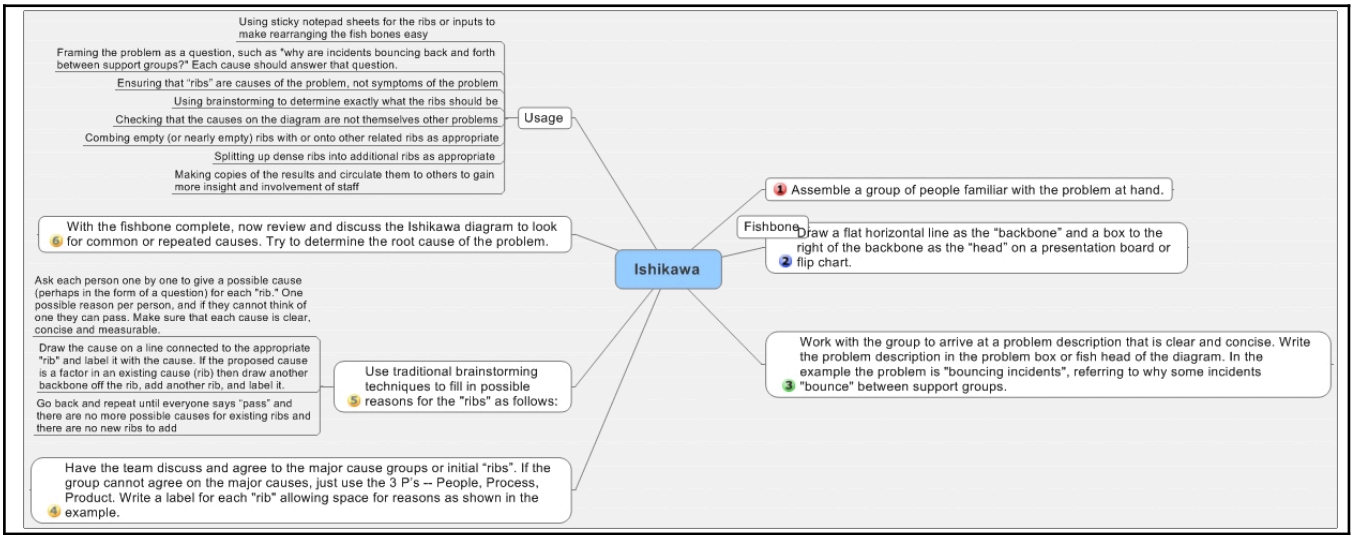
What improvements is problem management driving?

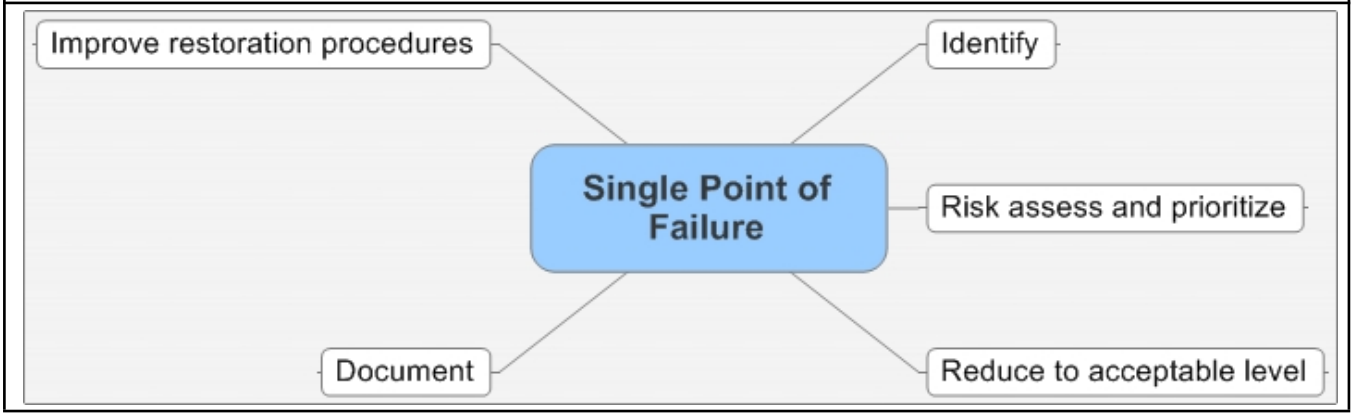




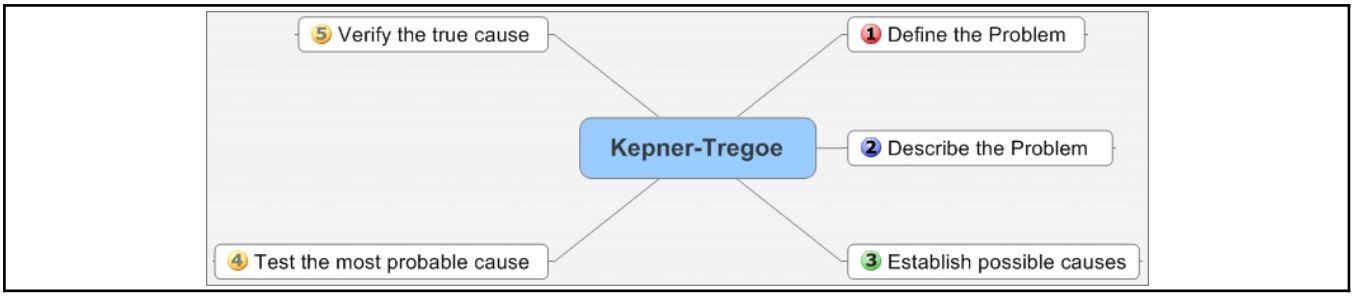

Formal process
Triggers
Matching an Incident against existing Problems and Known Errors is not successful
Study of Incidents shows many recurrent Incidents
Technical analysis of the infrastructure shows something that could lead to Incidents
A Major Incident occurs and requires an immediate solution
Known errors
Full visibility of known errors before a system enters production
Proactive collaboration with testers
Type
Volume
Impact
Drive out as many known errors as possible during testing
Priority
Risk
Impact
Exposure
Expanded Incident Lifecycle
The earliest time between when the customer realizes the loss of service or it is first reported
Incident start
Time when IT becomes aware of the incident
Incident detection
Time when underlying cause is determined
Incident diagnosis
Time when failure is repaired
Incident repair
Time when component recovery is complete
Incident recovery
Normal operations resumes
Incident restoration
Problem control
Discovers root cause
Error control
Evaluates root cause data
Major problem review
Deeper analysis of specific problems
Proactive problem management
Trending


Fault Tree Analysis
FTA
A root cause is one you can do something about; so now you need to think of the countermeasures you might apply to each root cause. List countermeasures for each root cause in a box under the root cause. For example, for "filter clogged" a countermeasure might be "clean filter monthly." Link the countermeasure to the root cause by drawing a line.
6
Continue identifying causes for each fault until you reach a root cause, or one that you can do something about. For example, the root cause of "power supply failure" might be "filter clogged"; the root cause of "battery backup exhausted" might be "battery backup too small".
5
Two logic operators – And and Or, also known as logic gates – are used to represent the sequencing of faults and causes. For example, "Email server down for more than 4 hours" could be caused by "loss of power" Or "hardware fault". Another might be "loss of building power" And "battery backup exhausted." Update faults and causes by grouping logically related items using And or Or between faults and events; and faults and causes. Re-draw the lines from top level event to logic gates to faults to logic gates to causes.
4
1 Select a top level event for analysis.
Sources
Problem/Known Error Records
Service Outage Analysis
Potential failures from brainstorming and a TOP
"what-if" scenarios based on Service Level Agreements, etc.
Identify faults that could lead to the top level event. Continuing the above example, some possible faults leading to an outage lasting more than 4 hours might be "loss of power", another might be "hardware failure." List all the faults under the top level event in boxes and connect the fault boxes to the top level event box by drawing lines
2
For each fault, list as many causes as possible in boxes below the related fault. Continuing the example above, in the case of "loss of power", some causes might be "electrical outage", "power supply failure", and so on. Connect the boxes to the appropriate fault box.
3





Kepner-Tregoe



Most infrastructure engineers do not actually follow Kepner-Tregoe. They rely instead upon preconceived ideas and often skip important steps. Then, without a plan and in desperation they fall back on the good old "when in doubt swap it out" technique. Taking the time to use Kepner-Tregoe can result in dramatic improvements in troubleshooting and deliver permanent fixes to prevent future problems as well. The full name is Kepner-Tregoe Problem Solving and Decision Making (PSDM) and this matches the ITIL *Problem Analysis* phase. Problem Analysis helps infrastructure engineers make sound decisions. It provides a process to identify and sort all the issues surrounding a decision. As a troubleshooting tool, Problem Analysis helps prevent jumping to conclusions. Immature trouble-shooters use hunches, instinct, and intuition. These individual acts of heroism may seem brilliant, but they can also result in more problems since jumping to conclusions often compounds or expands problems instead of solving them. Problem Analysis leverages the combined knowledge, experience, intuition, and judgment of a team, resulting in faster and better decisions. Using Problem Analysis to aid Problem Management not only brings the team together, but also helps identify root cause.

The Problem Analysis process divides decision-making into six steps:

1. Define the Problem
2. Describe the Problem
3. Establish possible causes
4. Test the most probable cause
5. Verify the true cause

Defining the Problem

Problem Analysis begins with defining the problem. The problem management team cannot overlook this critical step. Failure to understand exactly what the issue is often results in wasting precious time. Many immature trouble-shooters consider this step as wasted effort since they know what they are going to do – and this is the critical mistake made by many. Preconceived notions often result in increased outage duration and even outage expansion due to poor judgment. Since problem management is inherently a team exercise, it is important to have a group understanding of the problem.

Consider the following examples. A poor problem definition might appear as follows: “The server crashed.” A better problem definition should include more information. A good model for clarifying statements of all sorts is the Goal Question Metric (GQM) method. It results in a statement with a clear Object, Purpose, Focus, Environment, and Viewpoint. This results in an unambiguous and easily understood statement. A clarified problem definition might be: “The e-mail system crashed after the 3rd shift support engineer applied hot-fix XYZ to Exchange Server 123.”

When developing a problem definition always use the "5 Whys technique" to arrive at the point where there is no explanation for the problem. Using 5 Whys with Kepner-Tregoe accelerates the process.

Describing the Problem

With a clear problem definition, the next step is to describe the problem in detail. The following chart provides a nice template for this activity. You can do this using a presentation board, paper, or common office software. The worksheet describes the four aspects of any problem: what it is, where it occurs, when it occurred, and the extent to which it occurred. The IS column provides space to describe specifics about the problem -- what the problem IS. The COULD BE but IS NOT column provides space to list related but excluded specifics -- what the problem COULD BE but IS NOT. These two columns aid in eliminating "intuitive but incorrect" assumptions about the problem. With columns one and two completed, the third column provides space to detail the differences between the IS and COULD BE but IS NOT. These differences form the basis of the troubleshooting. The last column provides space to list any changes made that could account for the differences.

	IS	COULD BE but IS NOT	DIFFERENCES	CHANGES
WHAT	System failure	Similar systems/situations not failed	?	?
WHERE	Failure location	Other locations that did not fail	?	?
WHEN	Failure time	Other times where failure did not occur	?	?
EXTENT	Other failed systems	Other systems without failure	?	?

Establish Possible Causes

Anyone who has spent time troubleshooting knows to see “what has changed since it worked” and start troubleshooting by checking for changes. The problem is that many changes can occur, and that complicates things. Problem Analysis can help here by describing what the problem is and what the problem could be, but is not. For example:

Problem: “The e-mail system crashed after the 3rd shift support engineer applied hot-fix XYZ to Exchange Server 123.”

	IS	COULD BE but IS NOT	DIFFERENCES	CHANGES
WHAT	Exchange Server 123 crashed upon application of hot-fix XYZ	Other Exchange Servers getting hot-fix XYZ	Different staff (3rd shift) applied this hot-fix	New patch procedure from vendor
WHERE	3rd floor production room without vendor/ contractor support	Anywhere else with vendor/ contractor support	Normally done by vendor	New procedure, first time 3rd shift applies hot-fixes
WHEN	Last night, 1:35am	Any other time or location	None noted	
EXTENT	Any Exchange Server on 3rd floor	Other servers		

History (and best practice) says that the root cause of the problem is probably due to some recent change. With the completed worksheet, some new possible solutions become apparent. Shown above is becomes clear that the root cause is probably procedural, and due to the fact the vendor did not apply the hot-fix, but rather gave procedures for the hot-fix to the company.

Test the Most Probable Cause

With a short list of possible causes (recent changes evaluated and turned into a list), the next step is to think-through each possible problem. The following aid can help in this process. Ask the question: "If ____ is the root cause of this problem does it explain the problem IS and what the problem COULD BE but IS NOT?" If this potential solution is the root cause then the potential solution has to "map to" or "fit into" all the aspects of the Problem Analysis worksheet. Use a worksheet like that shown in figure 3 to help organize your thinking around the potential solutions.

Potential root cause:	True if:	Probable root cause?
Exchange Server 123 has something wrong with it	Only Exchange Server 123 has this problem	Maybe
Procedure incorrect	Same procedure crashes another server	Probably
Technician error	Problem did not always reoccur	Probably not

Verify the True Cause

The next step is to compare the possible root causes against the problem description. Eliminate possible solutions that cannot explain the situation, and focus on the remaining items. Before making any changes, verify that the proposed solution could be the root cause. Failure to verify the true cause invalidates the entire exercise and is no better than guessing. After verifying the true cause, you can propose the action required repair the problem. It is important here as well to think about how to prevent similar problems from occurring in the future. The Problem Manager should consider how the issue arose in the first place by asking some questions: *Where else might this problem appear? Are there other occurrences of this problem in the past? Do any procedures need to change?*

The goal is to try to eliminate future occurrences of the problem. Kepner-Tregoe is a mature process with decades of proven capabilities. Kepner-Tregoe Problem Analysis was used by NASA to troubleshoot Apollo XIII – even though the technicians did not believe the results, they followed the process and saved the mission. The rest of the story, as they say, is history...







Change Freeze

IT implements change windows to introduce operational stability over critical IT and business processing periods. There is also a due diligence requirement on changes to provide resources sufficient time to analyze and assess the associated risks. This measure reduces the number of Major Incidents occurring.