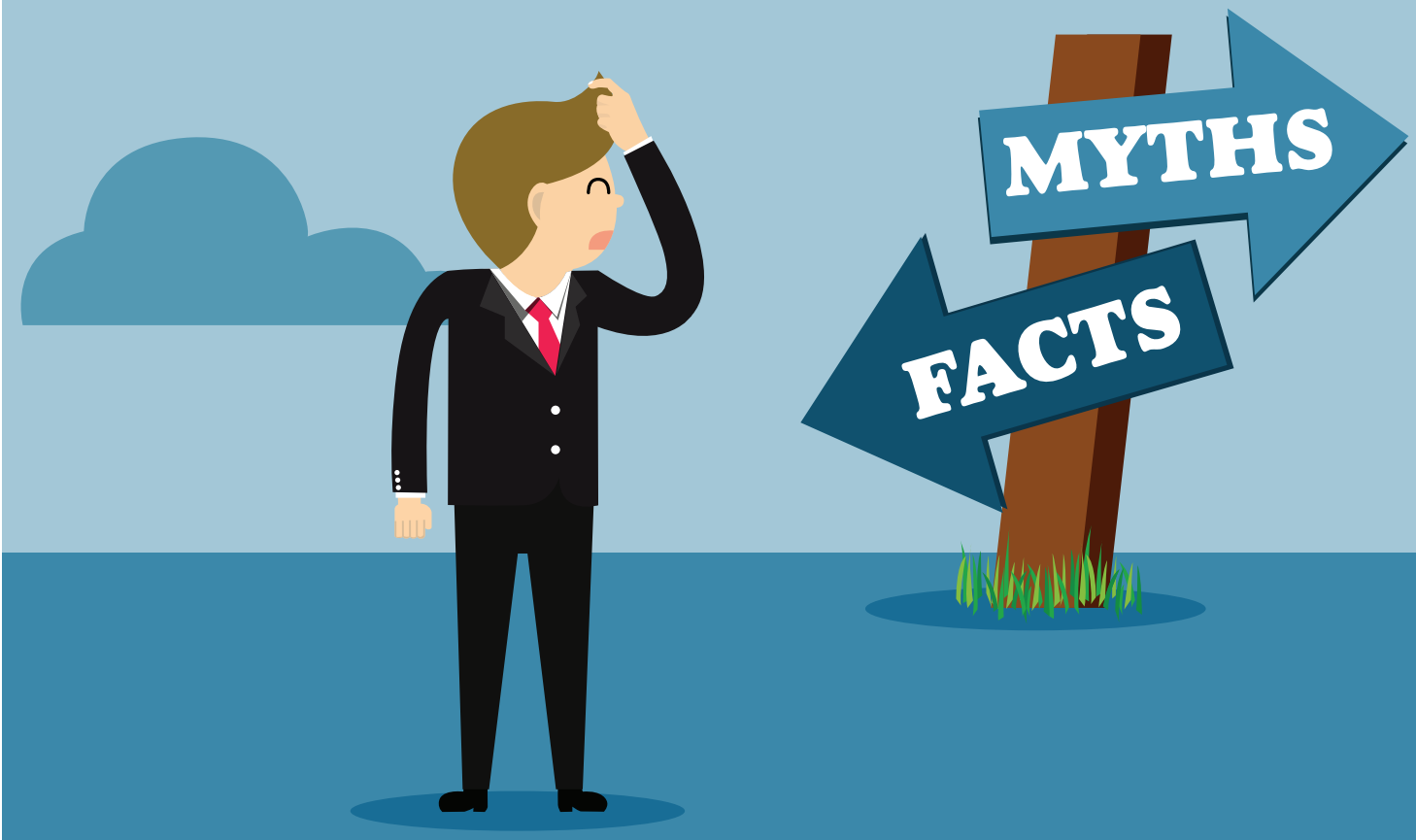
ITSM.tools

10 Common ITIL Myths... Busted!



Vawns Murphy

Introduction

So, here's the thing. ITIL isn't the easiest thing to sell at the best of times, but recent world events mean that economic forecasts are rough and it's getting harder to get funding for projects. Think about it – everyone has a horror story about an ITIL project that's gone awry, a tool that cost thousands but delivered nothing, or a process that was meant to drive efficiencies but ended up causing nothing but endless headaches and red tape. There's also a lot of misinformation and myths out there that can make ITIL a harder sell.

Here's the truth of it – ITIL, and the set of best practices that go with it, is actually a common sense mechanism for IT service delivery. Even if you've just got some of the basics in place – such as a service desk or change enablement – then you're already making a start. And the opportunities for, and benefits of, ITIL only increase in uncertain times. Need further proof? Try these:

- By staying on top of incidents, you stop things from spiralling out of control.
- By carrying out root-cause analysis, you can fix things once and for all as well as nixing so called "boring "or repeat incidents.
- By managing planned work, you can understand the impact and plan changes accordingly.
- By having a configuration management database (CMDB), not only can you truly understand the impact of planned work, you can fix incidents faster because you can clearly see what services are affected as well as the impact to the business.

ITIL adds value to the organization by aligning and combining people, processes, knowledge, and tools – driving greater productivity, efficiency, and transparency, without compromising value or service quality. As organizations focus on their competitive differentiators, not only to survive but also to improve in order to remain competitive, now more than ever, your IT department needs to be seen to be delivering value to your business.

However, with so many ITIL myths and scare stories out there, it can be hard to see the forest through the trees – but don't worry, we've got you covered. With everything from dealing with major incidents to running your change advisory board (CAB) effectively, here are my top ten ITIL myths as well as tips on how to bust them once and for all.

MYTH #

01 No one cares what the incident form looks like

Short answer: In the immortal words of Shania Twain, "you must be joking, right?" Your incident form is your way of capturing all the relevant information to what's gone wrong and how to fix it. Your form must be well designed and applied consistently so that no matter who deals with the ticket or what's being logged, the incident is treated in the same way with nothing lost, ignored, or forgotten about.

Incident management is one of the first things your end user will see. Everyone in the business will need to log an incident at some stage. This means that incident forms need to be well designed and user friendly given that everyone in the organization likely sees them. It's a way of setting out your stool and acting as your shop window. It's also how you capture all the relevant information about the issue and how to fix it.

Your incident form also needs to be applied consistently. It doesn't matter how your end users are logging an incident – phone, online, self-service, or email – the look-and-feel should be consistent, and the same questions, prompts, and mandatory fields need to be there. That way it doesn't matter what time of day it's logged, or who on the service desk is dealing with it – whether it's Bob who has been there forever or Dave whose been there two days – exactly the same information will be captured and the incident will be routed in precisely the same way. This means that nothing can get lost, ignored, or forgotten about in the process.

Keep in mind that a good form will depend on the organization. Some highly regulated organizations, like those in the financial, healthcare, and public sectors, will probably need more details than others. In contrast, tech start-ups tend to be more flexible and need less detail. From an incident form point of view, it's vital that you find the right balance because it's something that has the potential to turn your incident management process from good to great.

Some useful examples of key items for any kind of incident form include:

- **Title** – Aka what's the problem? Ideally you want your end users to provide a short snippet that succinctly explains the performance issues they're experiencing or what's down.
- **Description** – This field allows end users to provide you with more detailed information on what's going on, including what the person was trying to do, and if there was anything different than normal occurring, etc.
- **What's the impact?** – This is where priority matrixes come into their own because, let's face it, every end user thinks their incident is the end of the world. If you have a queue of incidents and they're all priority "DEFCON 1," which do you do first? A priority matrix also helps avoid the "if in doubt, tick 'medium'" selections.
- **What service is affected?** – I know this sounds really obvious but sometimes when the service desk is under pressure and has a growing number of calls in the queue, we forget to ask the obvious questions. What is it that's causing pain? What's down or unavailable? What exactly is experiencing performance issues?
- **How many people are affected?** – If it's just one person who can't access their email, that's one thing. But if it's a department of 500 people, that's something else entirely. That's when you scramble to the Batmobile, start making calls, and start getting people out of bed.
- **Is a workaround available?** – It's important to know if it's a "hard down" or if there's a way to work around the issue. You need to know if there's anything you can do to mitigate the incident while you fix it permanently.

One final piece of advice: it's especially important in a self-service environment to make the form as easy to navigate and to use as possible. One of the golden rules of ITIL is to always make it easy for people to use your service and to follow the process in the right way.

MYTH #
02

There's no need to have a separate major incident process

Short answer: Major incidents are, by their very nature, serious stuff. They're issues that cause severe business impact which may extend to financial, reputational, and even regulator or legal ramifications. In other words, a serious incident deserves a serious, appropriate process.

I know there's a school of thought that says: "they're all incidents and they all need to be treated exactly the same way." But major incidents are different. They're the big, clunky, serious stuff. The stuff where when things start to escalate, senior management gets involved and the business starts escalating. Ultimately, major incidents are the ones that cause severe impact and pain and that might extend beyond your organization and result in financial, reputational, or legal ramifications.

- **Financial** – I used to work for a trading company and there were very clear financial risks associated with major incidents. We had one particular application where for every 15 minutes that it was down the trading floor lost a million dollars. That's a scary amount of money!
- **Reputation** – Say you wanted to download a book or a song from Amazon, but it wasn't working, you'd go elsewhere to iTunes or Barnes & Noble instead, right? Additionally, you'd think to yourself that Amazon clearly can't be relied upon. Amazon is all about being an easy way to shop online 24/7 and a major incident could completely ruin that reputation.
- **Legal/Regulator** – It's not been uncommon of late for there to be banking-related downtime in various areas of the world. That's where regulators start to get involved and make sure everything is as it should be. Additional audits, fines, sanctions, and even legal action can be the result of a poorly handled major incident; so make sure you're not the example everyone uses as the cautionary tale of what not to do.

TIPS FOR CREATING A MAJOR INCIDENTS PROCESS

Whenever you put in any major incident process you should ensure that you create a checklist that details step-by-step what needs to be done. For example, at five minutes you recognize that it's a major incident, you've logged it as such, and you send out a communication to the appropriate escalation teams. Within 10 minutes, communications have gone out to the business with an initial acknowledgement that you're aware of the incident and the team is working on it. In 30-60 minutes, another update is sent out. And so on, and so forth until the incident is resolved.

MYTH #

03 | Aren't problems just incidents that we haven't managed to fix yet?

Short answer: Incidents and problems are two completely different things. Incident management deals with coordinating the incident, managing communications with both technical support teams and business customers, and ensuring that the issue is fixed as soon as possible. Problem management focuses on root cause investigation, trending, finding a fix, and ensuring that any lessons learned are documented and acted on.

If you're struggling with the difference between incident management and problem management, think of it in metaphorical terms. Incident management is Batman (the superhero), and problem management is Columbo (from the crime drama series).

Incident managers are the superheroes of ITIL and their motto is "fix it quick." The goal is to swoop in, save the day, and get everything running again – just like Batman. Incident management also deals with coordinating the incident – managing communications with both tech teams and the business customers – to ensure that it's fixed as soon as possible.

In contrast, problem managers are like Columbo. They're the diligent detectives who come in after the event, ask all the right questions, and figure out what happened, why, and how to fix it. With problem management we're looking at the root cause so trending is extremely important. Has this issue popped up before? Is this a recurring thing? What's causing it? Next is finding a fix (both a temporary workaround and a permanent solution), working with change enablement to get that fix in safely, and then making sure that any lessons learned are captured, documented, acted upon, and built into continual service improvement (CSI).

DEFINITION RECAP

- **Incidents** – Incidents are interruptions to an IT service, a reduction in the quality of that service, or a failure of a configuration item (CI) that has not yet affected service. They're instances where something's gone wrong. Incidents don't become problems, they're blips or downtime, and they're unplanned.
- **Problems** – Problems are the why. They're the underlying root causes of one or more incidents. For example, an incident could be that five people can't access their email while the problem record could be that the server is experiencing performance issues because it's not patched to the optimal level. The problem record is there to look for the root cause and to figure out a fix.

While we're talking about definitions, we might as well throw "known errors" into the mix, as it's common for people to confuse these with problems. A known error is a type of problem record where we've figured out the root cause and we have a workaround. We don't have a permanent fix yet but we do know what the root cause is and we've identified a temporary solution. We log these in a known error database (KEDB) in our problem management tool and we ensure they're tracked, updated, and worked on. Known errors can be raised proactively by support teams or by suppliers, vendors, or third parties.

An example of a known error would be the Samsung Galaxy Note 7 – where it can overheat and become a fire hazard – which has led to safety announcements on planes stating that they're not permitted aboard the aircraft. In this instance, we know there's an issue, so we don't travel with it. This is a perfect example of a known error that was identified by Samsung, which resulted in them reaching out to their customers, and the wider public, notifying them to implement a workaround.

MYTH # 04

You don't need to be proactive

Short answer: Of course you do! It's important to look at both the proactive and reactive sides of your problem management process and find the balance between the two. If you focus only on reactive activities, you never fix the root cause or make it better; you just keep putting out the same fires. If you only focus on proactive activities, you'll lose track of the live issues causing the most pain and your service quality could spiral out of control.

Listen, I get it. We live in a world of "I want it now." We want it fixed and then we move on to the next crisis or drama. But if you focus only on the reactive side, you'll never find and fix the root cause or make it better. You'll just keep putting out the same fires time and time again.

Conversely, if you focus too much on the proactive stuff, then you'll look at all the "what ifs" and potentials and you'll get completely sucked into that rabbit hole. Then you might lose track of what's going on in your production environment and miss the most pressing issues. What's causing the most pain? Then your service quality can take a turn for the worse.

It's really important to have a balance. When starting out, your focus is naturally going to be on reactive activities but be sure to build in some time to be proactive as well. Whether that's 5-10% of your time or simply one meeting a month where you speak to people and ask what's worrying them. You can always extend it over time as your process matures and you get a tool and more resources in place. But be sure to have that little opening in the process from the beginning so that you can genuinely say you're looking at both sides, eventually you'll get that balance.

WHAT PROACTIVE ACTIONS SHOULD YOU TAKE?

Proactive actions could include working with availability and capacity management to ensure that uptime and performance concerns are addressed in new services, trend analysis to identify recurring incidents, and working with support teams to make sure that business critical business systems have the appropriate maintenance (e.g. regular patches, reboots, agreed release schedules etc.).

TREND ANALYSIS

I once worked on a client site where I had a meeting with the end user community to work on setting up end user forums. During this meeting, one of the business leads said to me, "We always have problems at month's end." I asked if he told anyone and he responded, "No. We don't tell the service desk because nothing gets done and we've given up logging it."

I did some historical trending and I noticed that for the past year, there was a spike in performance issues around month's end just as described and no one could figure out what was causing it. I ended up borrowing the technical observation concept from ITILv2 which is basically: if you have a problem and you have absolutely no idea what's causing it or how to fix it, you get someone from each area (networks, voice, Wintel, UNIX, LINUX, IT operations, applications support, etc.) to look at it. In this instance, it turned out that it was a combination of things including network contention across the local area network (LAN), batch jobs that needed to be optimized, a server that wasn't patched properly, and insufficient maintenance jobs.

It took roughly six weeks to work through the issues, but in the end, we shaved two whole hours off the overnight processing time and it completely nixed the performance issues at month's end. Had we not done that trending analysis and realized when it started and finished every month, we'd have never been able to resolve it.

TALK TO PEOPLE

This simply means talking to your technical support teams and getting their thoughts on items that haven't fallen over yet but likely will in a matter of time. Chances are that if it hasn't gone wrong so far, there isn't going to be much urgency to get it fixed, but at least it's on the radar. This way we can come up with a plan so when it eventually does fall over, we've already done the pre-work and we know the different options of what it would take to fix it.

When speaking with your service delivery managers and business relationship managers, ask them "what keeps you up at night?" For such managers nothing focuses their minds more than if they've got a monthly service review meeting with their customers the next day and things are not going well. These managers should have a holistic view of the end-to-end service, including all the things that have gone well and all the things that have gone wrong. This is valuable information. Ask them what things worry them – because it's likely they'll be different from what your executive teams will tell you.

It's also important to speak with your customers to ask what they're worried about most. Do they have any business-critical training times or major events coming up? For example, Alliance Healthcare, a subsidiary of Walgreens Boots Alliance, makes 65% of their annual profits at Christmas time. During this critical time, they have a change freeze on all transactional and financial systems to ensure that nothing destabilizes the business.

The bottom line is to just get out there and talk to people. It sounds simple and basic but the truth is that we don't do it enough. In IT, the temptation can be to focus on all the gadgets and the latest and greatest tools, but we also need processes and people. Otherwise, the tools will not be nearly as impactful.

MYTH # 05

You can't do change enablement without a comprehensive configuration management process in place

Short answer: Not true! While it's definitely easier to do change enablement with the support of a configuration management database (CMDB) so that you have a definitive impact on what you're changing, change enablement can still bring some much-needed control, support, and governance to your estate. It can also be used as a springboard for a configuration management process.

A lot of people argue that you can't do change enablement without configuration and a really mature CMDB in place, but the reality is that that's a very theory-driven mindset. The CMDB is a bit like a unicorn – everyone talks about it, but how often do you actually see one in real life? Sure, having change enablement alongside a CMDB is the dream scenario because it allows for really accurate impact assessments, but what do you do if you don't have a CMDB? Do you just not do change enablement? With leading industry analyst firms Gartner and Forrester both saying that around 65-70% of incidents are caused by change activity in some shape or form, skipping change enablement is just not an option.

SO HOW DO WE MANAGE WITHOUT A CMDB?

- **Look to your service desk** – Check out incident management to see what is already documented in terms of business-critical services and who supports them. Chances are the same people who are fixing incidents are going to be the same people raising changes so you can do some impact assessment that way.
- **Look at your service catalog if you have one** – Your service catalog will give you a view of all your live production services, or at least the most critical ones. Make a point of matching the services flagged in a change record against the service catalog to give you an idea of impact.
- **Make affected services a mandatory question on your change form** – Asking your change owners or the person raising the change to list the business services impacted by the change is key. If they don't know, that means talking to people until they find someone that does – whether that's the business, the service desk, relationship managers, service catalog managers, service delivery managers, etc.

MYTH # 06

I don't have time for formal change enablement in my organization

Short answer: Not true… If change enablement is seen as an inhibitor rather than an enabler then, in the nicest possible way, you're doing it wrong! Templates, models, and standard changes can all be used to ensure change can be delivered both efficiently and safely, all while saving time.

This is something I hear time and time again. When I arrive at a client's site to put in, or improve, change enablement, I'm often met with looks of distrust. The general attitude being: "We're too busy to raise changes." But in reality, you're too busy not to raise changes. Following the change process means your change is going to be deployed effectively, efficiently, and safely by making sure that all the appropriate checks are carried out. This includes answering questions like: Who's making the change? Is any downtime needed? What approvals and communications are necessary? How will the change be tested? Who's going to be supporting the change? What's the plan if something goes horribly wrong? Do we roll back? Is the person making the change that night empowered to make that decision? The list goes on and on.

"Getting all your ducks in a row" ahead of time will most certainly save time in the long run. I've seen horror stories where a change was deployed to production, time was running out, and deploying to disaster recovery was completely forgotten about. Then six weeks later a disaster or business continuity issue occurred where invoking disaster recovery was necessary but impossible because the code was weeks out of date. If you were too busy before, just imagine having to deal with that nightmare now!

Making sure to carry out the appropriate post-change verification checks ensures that everything is as it should be. And the process doesn't have to be complex. A lot of organizations complicate matters by trying to implement change enablement without looking at their specific business and environment. They just throw the ITIL books at it when there's actually a lot that can be done with models, templates, and standard changes. When I go to implement change enablement, I spend an afternoon with each support team and ask what changes they're raising all the time. For the ones that are low risk/low impact we follow the standard change route and for everything else we just template. It becomes a simple matter of selecting the right template, plugging in the dates, and then detailing any exceptions so it takes thirty seconds to a minute to raise a change instead of ten to thirty minutes and beyond. There really is no excuse. I cannot stress the importance of sitting down with your support teams and noting all of the change types, enough. You get the detail you need, it only takes them minutes to raise the change, and everyone's comfortable.

MYTH #

07 Change advisory board (CAB) meetings are just a box ticking exercise

Short answer: As a former Change Manager, I can honestly say that the CAB is one of the most important and useful meetings that a service-orientated organization can have. It sets out a view of what's happening to key services over the next week, reviews previous change activity, and looks at CSI. What's not to like? Use templates, models, and standard changes to ensure that the CAB meeting can focus on the major, high category changes that need to be sanity checked and talked through.

This is a really frustrating mindset. Sometimes it feels like you have to drag people to a CAB call kicking and screaming. But in reality, if your focus is on delivering value to your customer, the CAB meeting is one of your most important meetings. Not only does it give you that forward view of planned work, but it also lets you look at the work that you did last week. Did it go well? Did it not go well? If it didn't go well, what was the impact? And if it did go well, fantastic! Can we template it? Can we have some more change models? Is this a new standard change? It's also a place to discuss CSI. For example, if something went well, it's looking at the positives and the lessons learned that can be applied to other projects or programs.

TIPS FOR GETTING THE MOST OUT OF YOUR CAB MEETING

- **Respect people's time** – If you have special guests that will only be talking about one or two changes, then have those changes discussed first so they can drop off. There's nothing more soul destroying than spending two hours in a meeting when you've only been needed for the first five or ten minutes. Be mindful of special guests and let them drop off when they're done representing.

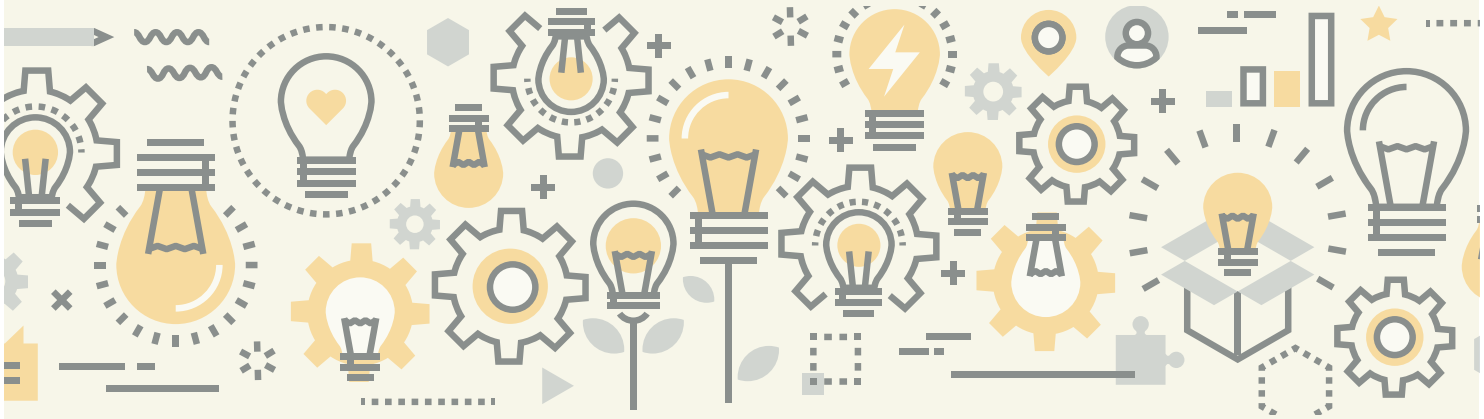
- **Don't discuss everything** – If it's a standard change where it's zero risk/zero impact, you really shouldn't be discussing it at CAB. That's something that can be pre-authorized ahead of time. CABs are there to capture the important, high-impact changes. Focus on the changes that will be visible to the business and could cause all kinds of pain if things go wrong. Don't clutter up your CAB with minor changes like making a network port live or a small desk move.
- **Keep it fast-paced** – This reminds me of a customer engagement where I was brought in to turn change enablement around. The first CAB went on for two hours and someone actually fell asleep. I had to make the decision whether to wake him up or take a picture for social media. Don't worry! I woke him up discreetly. But in all seriousness, it's about setting the expectation that the purpose of CAB is a quick recap. It should really be about two to three minutes per change. What you don't want is people getting confused or breaking out the change record verbatim. This is just the high-level information that enables the rest of the CAB to make effective decisions.
- **Have a terms of reference** – Make sure everyone knows the agenda and understands their roles and responsibilities. Then when you're in the middle of a CAB meeting, no one can say: "I'm not sure if I can actually improve this or not?" or "I don't know if I'm the right person to do this?" The person that attends has to be empowered to make decisions, and if someone can't attend, then a substitute can be sent who is empowered to make decisions on that person's behalf.
- **Share the schedule of change.** I recommend sending out a quick email with the link to the schedule of change. This reminds CAB members to read it beforehand so they come armed with all the questions and concerns that they have. The same goes for after the CAB meeting as well. Issue your report as a change schedule – a list of approved changes that you can filter by date and service, that way everyone knows what's going on. Another report that's useful to share post approval is the projected service availability report. I know, I know, ITIL 2011 called it the projected service outage report but let's take a step back for a moment. No one in a business-facing role wants to hear the word "outage" associated with their service. Outage has big scary connotations such as uncontrolled downtime, leading to incidents and complaints. Let's not frighten the horses here. Stick to projected service availability – it implies that you're in control and that any downtime associated with change activity has been planned, communicated, and authorized.

MYTH #

08 Service configuration management is an all or nothing game

Short answer: Not true. Normally I'm the first to say "either go big or go home" but that's not the case with configuration management; it's much too important. Done well, configuration management is the process that can help you manage, control, and protect your production environment and has the power to improve every other ITSM process.

Often when I meet with a client to discuss putting in or fixing configuration management, they start having flashbacks to when they first tried to accomplish this task years ago. Maybe an expensive configuration management tool was purchased or previous consultants were brought in but for whatever reason it didn't work. After years of experience, I can confidently say that nine times out of ten the reason the project did not succeed was because they never set a scope or their scope was too broad. With configuration management it's so easy to say that you're going to do everything under the sun – until things have snowballed into this unmanageable endeavor. The problem is long-term sustainability. When you run out of tools, people, or budget, all of a sudden you've spent massive amounts of effort and money without actually achieving anything. The key is to start small and work your way up.



Here are my own tried-and-tested recommendations for configuration management

- **Start with your most business-critical process** – I always recommend to build up configuration management over time. Start with the service that everyone uses and couldn't survive ten minutes without.
- **Talk to the business and your service desk** – Not sure what your most business-critical service is? Look at your service catalog or your service level agreements (SLAs). If you don't have that, it's time to speak to your end users. Not an option or not sure who to ask? Talk to your service desk. Ask them which service is a nightmare to support. Find out which one falls over the most often and takes the longest to get back up and running causing an angry mob to start gathering with their pitchforks.

- **Map out your service from end to end** – Now that you've figured out which service to start with, get the process mapped out perfectly. Check with everyone involved from a support perspective and get it documented and added to your CMDB.
- **Play with the data** – Ask your technical teams to try to impact-assess changes and have your service desk document the impact of incidents. Be sure to have people do a sanity check and play with the data you collect. Look at the best ways of collecting it and make it repeatable by using templates. Or, have a core set of data and attributes that you plan to collect along with the relationships.
- **Repeat** – Use that first service as a prototype. Once you have that service nailed (and sanity checked), you've now got an approach that makes it easy to repeat the process again for the next most painful service, and so on. Keep going until you have captured all your most critical systems.
- **And again, keep it simple** – It's always easier to build up over time or add things as you go along. Whereas if you start trying to collect a hundred different things about each individual attribute, before you know it, you'll have tons of data that is adding no value.

MYTH # 09

The tool will fix everything

Short answer: If only! The temptation to throw a tool at a situation in the hopes that it will solve all your problems is a big (yet common) mistake. Like Julia Roberts says in the shopping scene in Pretty Woman, "Big mistake. Big. Huge." A tool is only as good as underpinning information and processes. If you don't get those right, then your tool will just sit there gathering dust.

After spending money and time on a new tool, people expect that it will fix everything. But in reality, it's just a tool. You can have the most state-of-the-art ITSM tool on the market, but if you don't have the right data, people, and processes in place to support it, it won't do any good. The tool is not going to get the job done and it won't perform to optimal levels.

It's frustrating to hear people say, "We bought this tool and it's going to solve all of our problems." That's when I start asking questions like: Did you work with the vendor? Did you train people? Did you hire a Configuration Manager? Typically, these clients respond "No" to all of the above, thinking that everything would magically work once the tool was switched on. Unfortunately, it doesn't work like that. Success hinges on your people and processes.

- **People** – You'll need a full, experienced team in place to do configuration management. You'll also need your supporting players, like change enablement to make sure that everything is under control. For example, if you don't update your configuration management system, or your CMDB, every time you make a change, it's going to get out of date.
- **Processes** – I recommend having touch points across the processes. For example, in change enablement a change should not be marked off as successful until the entry for what's been changed (a server, a piece of software, a patch, etc.) has been updated in the CMDB as well. That way everything stays in sync.

ADDITIONAL CONSIDERATIONS FOR SUCCESSFUL CONFIGURATION MANAGEMENT

As you can see from the above examples, configuration management is not just a tool. Anyone who says "Buy this solution and you're all good" is totally off base. Having the right people and a solid process in place is vital. That said, there are a number of other considerations to take into account such as:

- **Planning** – Do you have the right plan in place? It's important to develop a plan that details your scope, including what you're going to attack first.
- **Baselining your existing environment** – Can you use automation? Can you use a tool? How do you verify that? What checks do you have in place? This step is all about identifying the key services captured at the planning stage and adding the details to your CMDB. By carrying out a baselining exercise, not only do you have your key services captured, you also have a "snapshot" of your environment at that point in time, something that will live and breathe as your configuration management process matures.
- **Control** – If you're not sure about what controls to put in place then look to your change enablement policy and process. Eventually, everything under the control of change enablement should also be supported by configuration management, that way, what you have in the CMDB really does match what you have in your real-life environments. If your production environment is going to be under the control of change enablement, what about other environments such as development, test, or quality assurance (QA)? If they're not under change control, is it worth adding them to the CMDB because there are no guarantees that this information is going to be kept up to date.
- **Looking at the lifecycle of things** – Make sure that one of the key attributes you capture is where something is in its lifecycle so that you can tie into things like IT asset management (ITAM) and software asset management (SAM) from a licensing perspective. This is a really quick way of realizing benefits.
- **Verification and order** – Trust but verify. Make sure everything is built into the data logs so that instead of having a painful audit once a year, you can carry out spot checks. That way, a list of completed changes can be compared to what's been updated in the CMDB on a monthly basis and things will stay in sync.
- **Reward** – Working in IT is awesome. Think about it, we get to save the world one Windows update at a time and provide awesome levels of service to our customers; so who says configuration management has to be a zero-fun activity? Is there a way to incentivize your team to encourage them to follow the process? For example, one recent trend that I really love is gamification – merit badge or mini quests for updating the CMDB correctly or adding information about key services can make following the process much more appealing. Having a small reward at the end of the month (for example an Amazon or Starbucks gift card) can generate a healthy sense of competition so that your teams not only follow the process, but take pride in doing it well. That's a cool way to make it fun and get buy-in.

ITSM.tools

MYTH # 10

Putting in a CMDB is too difficult and expensive for very little gain

Short answer: Ok, I get it. Configuration management is probably one of the most daunting ITSM projects to undertake but for all the concerns, you need to counter with the potential upsides. You must ask yourself questions like: What's the impact of all the failed changes because we couldn't do proper impact assessments? What about all the incidents that breached SLA because it took too long to fix them due to lack of support information? How much has been spent on fines or extra licenses following an audit due to being woefully under licensed?

It's funny how configuration management tends to be one of the most difficult processes to sell in the ITIL or ITSM world. Everyone understands the value of incident, problem, and change, but configuration management is somehow tougher to gain buy-in. Often clients will tackle the first three but by the time they get to configuration management, they claim there is no time or money to implement and run it properly. It's my job to challenge that and show the substantial benefits they are missing out on by not implementing configuration management.

"WE DON'T HAVE THE TIME"

Let's look at how much time is wasted without it. How much time was consumed firefighting incidents that could have been prevented had we known how things fit together? Could these things have been fixed more quickly if we fully understood the impact and all the underpinning services? How about all the time we spend assessing changes? Years ago, I worked for a client in the Telecoms industry and because of the cabling, records, and age of some of the data, it could take up to 12 weeks to impact assess a certain type of change accurately. What kind of madness is that? With a CMDB, it becomes a much quicker job.

"WE DON'T HAVE THE MONEY"

What about all the fines or retrospective licenses? For example, let's say you're audited by a software vendor and you're found to be under licensed. Suddenly you're paying the fines or making up the short fall. Or what about the growing issue of zombie servers? Many organizations that I've spoken with have servers where they don't know what's on them or if they're even being used. However, IT is too scared to switch them off in case the business comes screaming. As a result, these zombie servers are left in place to continue taking up space and consuming power.

People consistently come up with these excuses for not doing configuration management but it's a vital process and the benefits of it are huge. We're not just talking your "bread and butter," it's the cornerstone of every other process and can be used to springboard a service catalog or change enablement. It can make it easier to fix incidents and problems, and it's about money savings as well. There are significant cost savings if you know what's in your environment and how it's built up. If you have your configuration documented and up-to-date in a tool that reflects what's actually being used in your real environment, you can recycle, reuse, and redeploy the hardware and software licenses.

BONUS

Keep moving forward

Your ITIL processes will improve and mature over time. Build in CSI checks into your process so that you can build in improvements at the operational, tactical, and strategic levels. In the words of Walt Disney, "Around here, however, we don't look backwards for very long. We keep moving forward, opening up new doors and doing new things, because we're curious… and curiosity keeps leading us down new paths."

Ensure that your metrics map all the way back to your process goals via key performance indicators (KPIs) and critical success factors (CSFs) so that when you measure performance you get clear, tangible results rather than a confused set of metrics that no one ever reads let alone takes into account when reviewing operational performance.

Your processes will mature over time and by continually looking to be better you'll put quality at the center of everything you do and all the services you deliver. The added benefit of embedding CSI into your processes is that the quicker and easier they are to use, the more people will use them and the safer your organization will be.

FINAL THOUGHTS

Done well, ITIL can be used to transform IT service delivery; helping you to manage, control, and protect your estate. Having the right processes in place can reduce downtime and empower your people, as well as improving overall customer satisfaction.

The key messages on ITIL adoption are:

- To keep going, and to keep moving forward.
- Communicate the benefits to your people so that you get support and buy in.
- Develop strong processes, which support and build on the success of what you have in place already.
- Get people involved, from your end users raising incidents to the technical support teams that fix them, to the developers planning software deployments.
- Do not try to do it all at once – start small and work your way through best practices in a way that's relevant to the organization.
- Ask for help when needed, trust but verify and use the industry standards, frameworks, and advice from other organizations to keep getting better.

- Ensure that your service desk or ITSM tool is aligned with ITIL best practice and – just as importantly – is a great fit to your organization's needs and ways of working.

Now that we've busted some of the most common ITIL myths, and got you headed in the right direction, let's get to giving your customers the awesome levels of service they deserve!

ITSM.tools is an ITSM-focused website and service offering independent industry analysis, advisory, content, and consultancy. Content ranges from ITSM tool reviews, blogs, and industry news, to ITSM tips and best practices.

CONTACT US

On the web: https://ITSM.tools
On email: info@ITSM.tools