## Stop, Collaborate and Listen

Aligning IT to business

Preston de Guise

## ABSTRACT

Despite the high importance placed on IT systems by many businesses, there is often a disconnection between the goals of the IT department and the business itself. This micromanual outlines three key strategies an IT department must follow in order to remain relevant.

# Table of Contents

COPYRIGHT AND CONTACT INFORMATION	3
PURCHASING	4
INTRODUCTION	5
STOP	7
STOP SAYING NO	8
STOP CALLING USERS STUPID	11
STOP BULLYING USERS	13
STOP BREAKING PROMISES	14
STOP EXPECTING USERS TO BE TECHNICAL EXPERTS	16
STOP SPEAKING IN TECHNICAL TERMS AT THE USERS	17
STOP THINKING YOU KNOW MORE THAN THE USERS ABOUT IT	18
STOP BEING CONCERNED ABOUT WHO IS SMARTER	19
STOP THINKING USERS SET OUT TO CAUSE PROBLEMS	20
STOP THINKING IT IS INDEPENDENT OF THE BUSINESS	22
COLLABORATE	23
RESPONSIBILITY	24
RAPPORT	26
RESPONSIVENESS	29
LISTEN	31
LISTEN TO YOURSELF	32
LISTEN TO THE CUSTOMER	34
LISTEN TO YOUR COLLEAGUES	36
HABITS AND CULTURES	37
IN CONCLUSION	40
ACKNOWLEDGEMENTS	42

# Copyright and Contact Information

First edition.

This work is copyright, Preston de Guise, 2014.

It may not be reproduced whole or in part without permission from the author.

The author may be contacted by email: preston.de.guise@gmail.com

# Purchasing

This micromanual is being made available in 'try and buy' format. You can read (but not distribute) this PDF, and if you find it useful, you're encouraged to purchase it in Kindle eBook format from Amazon.

You can find the micromanual on Amazon here:

http://www.amazon.com/dp/B00JYSVYW2

## Introduction

IT is a constantly changing and evolving field, and as a consequence IT departments are faced with a constant need to evolve. The IT departments of the 70s and 80s were like the wild west – haphazard and almost lawless at times; the IT departments of the 90s were full of keyboard-jockeys; and the IT departments of the early 00s were boldly stepping into the forefront of the business.

Today there's an ever-increasing application of standard business approaches to IT, formalising the way the IT department works. More so now than any other time it's important that we come up with ways of better managing how IT works with the business.

Getting the technology right for the business is the easy part. Getting the IT department working in alignment with the business is where the real effort comes in.

Years ago, a financial institution in Australia commissioned a third party survey of staff to understand perceptions of IT systems throughout the business. The survey polled both IT staff and the business end users, and at the crux of the matter was a deceptively simple question:

Are IT systems sufficiently available?

The answer summarised much of the dichotomy between IT and end users. IT staff answered based on knowledge of system uptime, etc., and came to the conclusion that systems in the business were highly available.

End users of course see things from a different perspective. It's not whether a system is *up*, it's whether it's *useable*. To put it simply: it's the difference between *having a hammer* and having a hammer *when you need it*.

The IT staff knew the hammer existed.

The end users knew that the hammer wasn't to be found anywhere near as often as they needed it. The net comparison is that IT staff said systems were highly available, but end users, making the judgement on the *useability* of those systems, said quite the opposite. While I'd been coming to the conclusion for years that there's too much of an air gap between IT and the business in most companies, *that* story, more than anything else confirmed it.

As a consultant, I observe a large number of IT departments. Most are doing the right thing, broadly speaking, but there's always room for improvement. Most IT departments will have (at times) a challenging relationship with their business. Invariably aspects of the relationship are good, but nevertheless the relationship isn't as ideal as it should be.

A decade ago, if a business thought its IT group was too costly and too much of a hindrance, the only real way it could resolve the problem was to outsource. The costs and challenges associated with outsourcing meant that a lot of businesses suffered through a less than ideal relationship with their IT departments simply because it was seen as the only viable option.

Yet, IT departments are facing challenges now that make outsourcing look like a walk in the park. A business can completely side step its IT department, go to a cloud services provider and stand-up a new service or application in minutes. That's not to say what is stood up is as 'perfect' as what the business's own IT department *could* have provided, but following the 80% rule and accounting for efficiencies in services provided at volume, there's a very good chance such a scenario will be *satisfactory* – at least, more satisfactory than not getting anything from an IT group that's disinterested in or unable to provide a result internally.

This has led to some pundits predicting the complete demise of the IT department. While that's unlikely except in the most extreme scenarios, IT departments that have a rocky relationship with their businesses certainly aren't helping the situation.

The solution to this problem doesn't come overnight: there's no silver bullet or magic spell that can suddenly fix how the business and IT department relate to one another. It requires hard work and effort, but it's ultimately rewarding for everyone.

It's a common mistake to think that just because something will take a while it must be complex. Sometimes an activity will take a while simply because it's *necessary*, and this is true with the solution to the business/IT relationship. The solution isn't complex; it just takes a while to get it working properly.

The solution has three essential steps: stop, collaborate and listen.

## Stop

The first step towards solving the IT/business divide is to identify and *stop* doing the *wrong* things. That's only the tip of the iceberg, but it makes a big start, and of all the changes delivers the most immediate improvements to the relationship between the business and IT.

When an IT group stops doing the wrong things, it has more time for doing the *right* things, of course. Critically however, this allows the business as a whole to develop a substantially more satisfied understanding of the IT department. Further, if the relationship between the business (or business users) and IT has grown antagonistic, it's fair to say that no longer doing the *wrong* things will start to break down walls and help build trust.

So it's critical to stop:

- Saying no
- Calling users stupid
- Bullying users
- Breaking promises
- Expecting users to be technical experts
- Talking in technical terms at users
- Thinking you automatically know more than the user about IT
- Being concerned about who is smarter
- Thinking that users set out to cause problems
- Thinking IT is independent of the business

All of these are (in lay terms) "no brainers" – or at least, they should be. Yet time and time again, staff in IT departments can seemingly fail at these simple activities. Another way of thinking about all of the above is the simple message: stop *stopping* the business.

#### **Stop Saying No**

The simple fact of the matter is that IT staff say 'no' too often – particularly when the answer isn't *really* 'no'.

Somewhere along the line many in IT forgot that the job of standing at a door was to open it, rather than be technical bouncers. IT's function is to *enable* a business to work more productively – to make better use of the existing tools and be on the look out for areas where new tools can win increased optimisation.

Saying 'no' doesn't achieve this. In fact, IT departments saying 'no' is exactly why *Public Cloud* gets such mindshare in senior business managers. Forget talk about costs, elasticity, scalability or flexibility, a significant contributing factor to someone in the core business thinking about the *cloud* can be snapping when they hear 'no' once too often from IT.

IT could actually learn a lot from the service industry. Ironically, it's going to *need* to in order to survive, but it should have been watching and learning for at least the last decade, if not longer. Consider a customer in a café trying to order something that isn't on the menu, for instance. There are two potential responses:

	We can't do that	
or:		_
	Let me see what I can do.	

The end result of *both* might be *no*, but the second indicates a willingness to help. Not only that, it indicates innovation – you might not get *exactly* what you want, but there's a chance you'll get something approximating it. The first response is less likely to engender a return visit – and in fact the customer may even leave to go elsewhere. The second response will probably start to create a relationship with the customer.

There are efficiencies to be had in conformity – that's for sure. Equally though, Bring Your Own Device (BYOD) is proving that there's other efficiencies to be had in allowing users greater flexibility. Sure, it creates a bit more work for IT, but IT isn't there to wag the dog.

This is something everyone in IT can work to address – stop saying *no*. That doesn't mean that staff have to say yes wherever they would have previously said no. Instead, it means learning to:

- Say how, or
- Say when.

That is, 'no' isn't the answer. The answer is either 'not this way', or 'not now'. If an IT support technician is solving an issue with Word at a user's desktop and they ask for a second monitor, the answer isn't *no*, it's *you request this from <role>*, or *you need to ask <at this time>*.

Neither answer is 'yes', but equally, neither answer is 'no'. Of course, there's a myriad of other potential responses, but they all boil down to one thing: giving the user other options.

An unqualified or seemingly intransigent 'no' is the most toxic word in the vocabulary to the business/IT relationship. Invariably, *no* arose for several different reasons:

- Users asking for services the IT department currently don't provide
- Users asking for equipment the IT department doesn't allow
- Users asking for things the IT department doesn't have budget for
- Users asking the wrong IT person for something

A desire to quickly answer the question and move on to the next task assigned makes 'no' the easiest response to a request that can't be immediately satisfied. Ultimately that's a self-perpetuating behaviour that minimises the potential for improvements within the organisation, since:

- The chances of a service being provided are much better if it's officially requested
- The chances of equipment becoming available (or alternatives being made available) are much higher if it's officially requested
- It's substantially easier to offer services or equipment if it's been budgeted for (and budget is rarely allocated without an explanation of what the budget is to be used for)
- It's considerably easier to get something approved when the request goes to the right person.

In each scenario, there's likely to be a better answer that IT workers can provide staff, such as any of the following:

- I'm not authorised to provide that, but you can request it from <role>.
- I don't think we have budget for that. Can you log an official request?
- We're not allowed to provide that, so you'd need your manager to talk to the IT manager.
- I can't do that, but <X> can. Do you want me to have <X> contact you?

None of those answers *commit* to a change of IT policy or allocation of budget (nor should they be seen to), but they indicate a willingness of the IT department to try to find solutions for individual users and the business as a whole. Switching to these types of responses allow the IT department to be seen as an enabler, rather than blocker of business functionality.

#### Processing the process

Much of this topic has centred on replacing 'no' with alternatives that allow users to formally make their requests. This is to ensure the formal requests are actually logged, and in turn provide an impetus for IT doing more.

This creates two new challenges:

1. It's important users feel a request doesn't go into a black hole. They need to be kept in the loop on its progress.

2. It's important that staff take the time to explain to users why they're being asked to request the service or action more formally – and that's not because "it's the new rule" or "it's the new process". Instead, make sure IT staff explain to users that by formalising the request for something that had previously engendered a 'no' response, the IT department wants to better service that request and correctly answer it. Again, this isn't about immediately saying yes, but demonstrating a determination to change, and be measured on that change.

In order to stop saying 'no' to the business, IT has to:

- Gracefully explain why a service or product can't immediately be provided
- Indicate if the request can be made differently in order to be facilitated
- Log the request if it can't currently be actioned
- Follow up the request

'No' may be the easiest word to say back to the business, but it's also the easiest (and most rewarding) behaviour to fix.

## **Stop Calling Users Stupid**

Negative statements and negative behaviour become self-perpetuating if they're allowed to continue indefinitely, and people venting by calling end-users *stupid* is a relatively common symptom of conflict between the IT department and the rest of the business.

What begins as a way of letting off steam, if repeated regularly ceases to be about momentary stress relief and becomes *the accepted belief*. As we know from politicians, pundits and spin-doctors, an untruth repeated often enough will indeed gain believers.

This is challenging enough in an IT department with minimum staff turnover – however, in departments with higher staff turnover, it becomes a serious problem. New staff will invariably hear stories of this or that user behaving foolishly, and *expect it* when they interact, thereby tainting their experience with the bad habits of others.

That belief isn't conducive to good relations between the business and IT – quite the contrary. Look around the IT department – you'll note that the staff who get on most with end users are those who don't spend time judging the intellectual capacity of those users.

#### Words have meaning

I have an odd word association condition that comes from how I learnt to talk. Growing up, I had a severe speech impediment: it wasn't a lisp or a stutter – it was almost verbal dyslexia. It was so severe that I spoke gibberish until around my fourth birthday, when it became necessary for me to visit a speech therapist, who quickly diagnosed what I was doing wrong.

However, since I'd learnt to speak incorrectly, I had to learn how to speak again. It wasn't just a case of a couple of quick pointers, it was painstaking lessons repeated daily. Core to those lessons were flashcards: I'd be shown a photo of a cat with the word "cat" underneath it, and I'd have to say "cat" a half dozen times until I got close to the correct pronunciation. Then "dog", and so on. It took months but eventually I learnt to speak properly.

That meant I learnt to read as I learnt to talk, but there was a more subtle consequence that took me decades to learn: I didn't just think words, I thought images associated with words. Not all words of course, just key nouns and strong emotive words in particular.

Hate is one of those words. Anyone who has read the Stephen Donaldson books, The Chronicles of Thomas Covenant, the Unbeliever would remember the ur-Viles and their bowls of vitriol, hate distilled to a black tarry substance. After reading Donaldson, I found that I visualised hate as such a bowl of vitriol.

It wasn't pleasant visualisation, so I eventually stopped saying the word. The use of language shapes thoughts, and I learnt that as I stopped saying hate, I stopped hating.

Use (or non-use) of language really does influence how we think.

The concept that language affects thought is often referred to as linguistic relativity, or the Sapir-Whorf hypothesis. While linguistic relativity is still a topic routinely debated, it is widely popular and continues to be actively explored in a variety of circles, both scientific and fictional. (For instance, 1984 by George Orwell explored linguistic relativity on the basis of Newspeak.)

It doesn't matter if it starts off as a joke or a means of venting steam, it becomes a wall between the IT department and the business: Stop calling users stupid. If you're an IT worker, stop saying it. If you're an IT manager, stop *allowing* it.

## **Stop Bullying Users**

Bullying comes in many forms, from the very subtle to the blatantly obvious. The more subtle forms come from a creative reshuffling of activity priorities based on requesting user. Is *that* user annoying? Then that ticket goes to the bottom of the pile, and constantly drops when there're other tasks to be done.

Then there's the full bullying to be found in just a few departments. It's rare, but not unheard of – and usually classified as "practical jokes" by the bullies. Accounts get randomly locked, proxy access is mysteriously broken for a day, or print jobs disappear. More of a symptom of the jockey days of IT, this still lingers in some departments.

Being in a position of trust comes with responsibilities, and abusing those responsibilities should have consequences, rather than chuckles around a lunch table.

As with all bullying, those who silently allow the bullying to go on rather than speaking up against it are as complicit as those who are engaging in the campaign – regardless of whether they're a colleague, subordinate or manager. Ultimately the work environment is no place for aggressive or bullying behaviour, and to be perfectly blunt, staff who engage in it need to be given a very short opportunity to correct their behaviour or removed from their positions.

### **Stop Breaking Promises**

IT has its fair share of disorganised or haphazardly organised employees. Just like any normal department, some IT staff will tell a user something will be "looked into", forget to note that down, then become distracted and forget about it.

However, IT departments are service departments – and like all forms of customer service, forgetfulness can have repercussions.

The forgetful barista
My nearby café has two or three main baristas. One of them is particularly forgetful. This was an exchange I had with her recently:
Her: What would you like? The usual? A latté?
(I never order a latté)
<b>Me</b> : No it's a cappuccino. Large, strong cappuccino please.
<b>Her</b> : What was that?
<b>Me</b> : A large, strong cappuccino please.
(less than a minute later)
Her: Do you want sugar in your latté?
<b>Me</b> : No, and I asked for a large, strong cappuccino.
(a minute later)
Her: What did you want again?
<b>Me</b> : A large, strong cappuccino please.
Her: Any sugar?
<b>Me</b> : No
<b>Me</b> : <walks a="" away="" latté="" with=""></walks>

That's at the extreme end of vague behaviour in customer service, but it has a detrimental effect on my opinion of the café, since I constantly have to remind the barista what I ordered. While from my perspective, it seems a simple enough process, that's not going definitely going to be the same for her: she's got at least 2 or 3 orders in front of her, listening out for a chef who may need an order taken to a table, and potentially overwhelmed by the noise of the coffee machine. I understand all that, but it's still an unrewarding experience.

That's why cafés and restaurants that strive for high customer satisfaction will be sure to record the customer's order. Our memories are powerful at times, but can equally be flawed when we're under stress or juggling multiple activities. On paper, "Large, strong cappuccino" is impossible to confuse with "Latté with sugar".

Similarly if a user asks an IT person about this or that other issue and the response is "I'll look into it", the user is likely to understand the IT person is busy, but they'll equally get frustrated if they never hear about the issue again.

Formalisation of requests remains an important solution here – staff should promise the user the issue can be looked into and do one of either two things:

- If they have time and access, immediately log the issue at that point so it goes into a tracked queue
- Request the user log the issue so it goes into a tracked queue

Of course, someone who is forgetful or gets distracted before they can deal with a request isn't *technically* breaking a promise, but the end result from the customer's perspective is the same: they asked for something, they were told it would be investigated, and they never heard back again.

In addition to logging requests, there's another aspect to be considered: *overcommitting*. We've all done it at one time or another in our past: we don't want to say no, or we want to appear to be helpful, and so we take on more than we can handle, but as a result some things fall through the crack.

Obviously, this walks a delicate line with *stop saying no*, but it *can* be managed. Those requests not only have to be logged, but they have to be prioritised, and if necessary, delegated.

In order to stop breaking promises to end users, one doesn't have to become a superhero capable of working a hundred hours a week – the secret isn't so much *more* work, but better recording of the week, and prioritised actioning of that recorded work.

As any manager will know, the recording and prioritisation of queued tasks is an essential input to calculating required staff numbers and workload. Thus, it's doubly beneficial: it can be used to justify more staff *and* it helps keep the business users assured their requests are going to be dealt with.

#### Stop Expecting Users to be Technical Experts

When practiced mundanely, IT is a science. There's exactness to what we do, how we work – a precision that comes from repetition and from knowledge of predictable systems. In that sense, it's no different from other forms of sciences – that's why Computer Science degrees will typically originate from either an engineering or science department in most Universities.

IT staff are paid to be experts – to be computer scientists or information scientists or information professionals for exactly one reason: so someone else doesn't have to be. A person might work in IT because they're a bit of a geek and find technology interesting, or because they have an absolute passion for computers, or perhaps even because they're simply good with computers and it's just a way of earning money.

But it's *their* job, not the job of the end users.

End users are accountants, managers, HR offices, clerks, blast furnace operators, police officers, doctors, professors, students, and all manner of other things, *but they're not IT staff*. It's not their job to be experts in IT – otherwise they'd work in IT.

If someone goes to a doctor and doesn't understand the difference between a myocardial infarction and a stroke, that doesn't mean they're lessened. Doctors work hard for their degrees and society expects *them* to be the experts in medicine, not the patients. Indeed, doctors who expect their patients to be medical experts don't have many patients.

Except in a small set of scenarios, end users aren't expected to be IT experts. That's not their job.

When practiced expertly, IT is an art ... an architect, senior consultant, senior DBA, senior system administrator or veteran programmer (or any other of a number of such roles) wields IT knowledge like Sherlock Holmes. A snippet of information there, a recollection of a memory of a story told fifteen years ago when they were just starting in IT, a crafty ability to optimise a search engine query and a voluminous hoarder of information yields an information artist – someone who works wonders while barely breaking a sweat, and topples the mundane without even looking at it.

End users aren't expected to be information artists, either. They shouldn't know that this error occurred 11 years ago and is vaguely related to a patch implemented by a now defunct vendor for Y2K compliance 3 years before that – that's the information *artist's* job.

Ultimately, the business employs IT staff and has an IT department so that end users can get on with *their* tasks.

Regardless of whether the IT being practiced is a science or an art, it's the role of the IT department to practice it, not the end users. Equally, it's important to remember that intelligence, expertise and capability in one field doesn't immediately translate into expertise in another field. It doesn't matter that the end user is a whiz with spreadsheets because she's the head accountant for the company, if she doesn't understand how RAID systems work, that's *OK* because it's not her job to. It doesn't matter if the head of HR can use the company's CRM like a wizard but he can't program in Java – because it's not his job.

End users aren't paid to be IT experts – that's the job of the IT department.

### Stop Speaking in Technical Terms at the Users

Following from our previous point, different people will have varying levels of technical skill, and it's important we communicate with them at a level that allows them to understand. My father, aged 70, uses a computer but doesn't really understand the entire GUI metaphor. The icons of the hard drives on the desktop are the "boxes", the dock at the bottom of the screen is just simply "that thing", and folders are, well, a bit too much. Despite repeated attempts to explain things a little better for him, he resolutely maintains just enough knowledge to use iTunes and is perfectly happy with that.

Consequently, if I'm trying to talk my father through an issue, I have to adopt the language that *he* understands in order to get the message across. So he doesn't get told to "double click on the System drive", he gets told to "quickly click twice on the box at the top right hand corner of the screen". I don't tell him to "go to the Dock", I say "go to the bottom of the screen". In the end it's easier. It may require a bit more concentration from me to map what I want to say to what he needs to hear, but he gets the information faster and the problem is solved faster.

The average end user understands GUIs somewhat better than my father, but the lesson remains the same. Speaking technically at a user who doesn't understand the concepts is a waste of everyone's time:

- It's a waste of time for the IT worker, because it's necessary to repeat explanations;
- It's a waste of the user's time because they're not getting back to their core task soon enough.

Not only that, it's likely that the net result is sounding like a condescendingly smug git. There can be a vast gulf between *talking at* and *communicating with*. Just as we shouldn't expect users to be technical experts, we shouldn't expect them to *talk tech* just because we find it convenient.

Returning to the doctor analogy, a good doctor would tell you that you suffered a "myocardial infarction", but would go on to explain that in layman's terms, that's called a heart attack. It's not necessarily precise, but the doctor understands that their end-user – the patient – doesn't come with a medical dictionary in his or her head, and that a healthy response comes from comprehension.

This has direct parallels with IT. A user asking why their message bounced back could be told:

- "You violated RFC 1870, generating a 552 response from the target host" or
- "Your email was too big. You need to break it up into smaller ones, or send the files another way."

Both are technically accurate responses, but like telling someone with an average IQ and no medical knowledge that they'd had a myocardial infarction, telling a user about RFC 1870 and response code 552 isn't *helpful*. It's not *communication*, it's a data dump. Arguably it's not even *information* because information is interpreted data, and the user isn't interpreting the data he or she is being supplied with in those circumstances. Yet, these conversations occur with surprising regularity amongst some IT departments where a driving factor seems to be an ongoing demonstration of one's knowledge to regular staff.

#### Stop Thinking You Know More than the Users about IT

Many IT professionals may have a broader knowledge about information technology than any of their end users. *May* – this is not actually guaranteed, depending on the end user or even the industry.

It's foolish to automatically assume that someone in IT will know more about IT than the day-to-day users of the technology. After all, we see this reflected in IT all the time with certifications – passing an exam and becoming certified in a particular stream of technology is no guarantee of expertise unless the exams are rigorously practical and require real world experience, and there's very few of those sorts of certification exams. For the most part, certification demonstrates an ability to regurgitate rote learning.

Just because the business users aren't being paid to work in IT doesn't mean they don't know anything about it. In lay terms, we usually consider such people to be *power users*. They're not part of the IT team, but they could be – at least for a particular area of technology.

While it's important not to assume people are technical experts, it's equally important to adjust the way they're engaged with when it becomes apparent they understand the technology particularly well. After all, anyone who has worked in IT and then been forced to go through first level phone support for a PC or internet connection knows just how mind-bogglingly annoying and tedious it is to be asked the most basic questions *even after* explaining they work in IT, deal with the problem regularly, and want to get past the *have you turned it off and on* questions.

Similarly, when an end-user gives clear signals he or she knows substantially more than initially expected, it's important to work at an adjusted level. Ultimately this is beneficial to everyone involved. (Too often the expression, "knows enough to be dangerous" is applied, without consideration that equally, users with more experience than average of particular systems are highly valuable.)

It's not surprising that core business users end up becoming extremely knowledgeable about the systems they use. A long-term graphic designer could very well know more about monitor colour calibration than any IT help desk staff member. A person who works in Excel 6 hours a day could very well know more about Visual Basic macro programming than someone in application support who dabbles with it every now and then to massage data coming out of a database. And of course, people increasingly change career paths. That just-hired junior librarian may have previously been a Unix systems administrator and wanted a change of pace.

Power users are a blessing in IT. They often act as a bridge between more regular users and technical teams, and they're quick at spotting new or unusual issues. The *business* regularly recognises their importance by placing them in advisory positions on technology improvement projects.

They're also much more proficient at providing the kind of information about an issue that can lead to it being resolved faster, and they have a better feel for the systems they use – often they'll notice changes or issues first. Similarly to the IT person calling up about PC or internet connectivity issues, they'll want to brush past the conventional questions and get to the root of a matter as quickly as possible. Stop ignoring them.

## **Stop Being Concerned About Who is Smarter**

Less of an issue in the traditional IT department, something that crops up regularly in any form of IT consulting is a need to *outrank*.

In this scenario, someone will go out of his or her way to *prove* their intellectual superiority. For some, it will be by the number of certifications held. For others, it'll be by the number of degrees, or years of tertiary study. The comparison techniques used are as myriad as they are pointless to the customers.

This behaviour is highly counter-productive.

Few people enjoy the feeling of being talked down to, yet the IT worker who tries to compete with the end user or customer on IQ is doing exactly that – looking for an excuse to be *on top* of the chain.

One of the better areas where this is exemplified is in Internet user forums, where so often all sense of civility is thrown out the window thanks to a pseudonym.

**Question**: I'm looking for a way to extend my wireless network so it's accessible on all four floors of my house. I'm currently using a WidgetsInc WiFi hub. The local computer store is selling a WidginInc WiFi Extender. Does anyone know if that's compatible with the WidgetsInc WiFi hub?

**Answer 1**: You should never use a WiFi Network. WiFi is like throwing your doors open and leaving a sign up saying "take anything you want"! Go invest in a Gigabit switch and wire your house up.

**Answer 2**: 700 years ago, WidginInc is rumoured to have made crossbows that killed some very good people. You shouldn't use them.

Answer 3: You can't extend WiFi, you can only BRIDGE it.

**Answer 4**: Just get a WaddleInc 4G dongle for each floor and do WiFi sharing from that!

Like it or not, those conversations don't just happen in anonymous internet forums.

However, regardless of *where* they happen, they occur for a simple reason: a need to "outdo" the person making the request, and be perceived as the smartest person in the room.

A healthy relationship is built on mutual respect. Needing to be the smartest can (and usually will) prevent that mutual respect from occurring – it's certainly not *showing* respect, and it's likely not to *earn* it, either.

The best way to earn respect from business users is to stop concerning yourself about who is smarter.

## **Stop Thinking Users Set Out to Cause Problems**

#### The Lost Document

A while ago I was working on an important backup network design for a customer. I'd had a flurry of inspiration and outlined a halfcomplete version of the document in just an hour before I had to head out for a meeting. As I was preparing to leave, I got a call, so I quickly saved the document, closed my laptop and headed out.

When I got to the meeting I opened my laptop and was greeted by a crash dialog for the diagram application, and sure enough, when I came back I couldn't find the document – it didn't appear in the recent items list, and wasn't where I thought I had saved it. After running a search of my laptop for some of the document keywords I still couldn't find it, so I had to start from scratch.

Almost a year later, when I was migrating between machines, I stumbled across the document, saved into the applications folder, which wasn't included in the search path in Spotlight.

Somehow I'd the wrong key sequence in the save dialog, saved the document to the applications folder, triggered an application crash by hibernating the laptop too quickly, and couldn't find the document.

We're all users at some point or another, working with imperfect systems and imperfect situations.

Most people at some point in their lives hear about Occam's razor, which states (paraphrasing):

The simplest explanation is usually the most likely one.

Occam's razor is a valuable lesson in life, science and philosophy, but it's not the only razor of importance, particularly in IT.

Less well known, but equally important is "Hanlon's Razor", which tells us:

Never attribute to malice that which is adequately explained by stupidity.

Of course, we don't want to think of end users as being stupid, so a more apt version of Hanlon's Razor for IT in particular should be:

Never attribute to malice that which is adequately explained by a lack of understanding.

Print this out. Post it on *every* desk in IT. Make it up as business card inserts for people's wallets and purses. Start meetings by saying it. End meetings by saying it. Repeat it until everyone in the IT department knows it by heart.

Make it the working **mantra** of the IT department.

End users don't sit around all day *plotting* about how next to cause a major disruption to systems. They don't plan to introduce viruses into the environment, run a search that

brings the fileserver to a grinding halt, or crash the mail servers by participating in a replyall storm. They're just trying to use the tools they've been given, often without comprehensive training. Assuming end users are acting maliciously is confrontational, counter-productive and in almost every single case entirely incorrect.

Further, the systems the users are accessing *aren't* perfect, no matter how much we may wish them to be. Users who log cases about corrupt or missing spreadsheets or documents aren't necessarily forgetful or careless, but instead are just as likely to have suffered the ravages of a buggy piece of code sending their documents into oblivion.

With many enterprise apps moving data entry into web-forms, additional layers of software and controls where a malfunction could happen are equally introduced. Since the systems the users are using aren't perfect, so it's just a waste of time behaving as if the users should be perfect too.

This is the single most important lesson I learnt as a junior system administrator: *never* attribute to malice that which is adequately explained by a lack of understanding. (In fact, it applies not only in IT but also to almost all of life in general. Most people don't go around trying to make problems.) People make *mistakes*. As such, it's important not to take their mistakes personally – it doesn't help anyone.

That's not to say that every mistake an end-user makes that causes issues for IT just gets laughed off as typical user behaviour. Mistakes will repeat unless education takes place. If a few users crash a mail server with a reply-all storm including the entire company address book, they need education about appropriate systems usage, not assigned malice of Medusan proportions. (Equally, jokes inside the IT department about education via a solid piece of timber applied rigorously to the cranium may *seem* humorous, but aren't, and don't contribute to the solution of the problem either.)

Hanlon's Razor should be the motto of every IT department, and should be instilled in the workplace attitudes of all staff within the department.

## Stop Thinking IT is Independent of the Business

Years ago I heard a simple piece of business advice:

If you want to see how indispensible you are, stick your finger in a glass of water, then measure the size of the hole you leave when you pull it out.

IT departments would do well to keep this lesson firmly in mind.

Equally, this aligns to the dog metaphor – does the dog wag the tail, or does the tail wag the dog? Obviously, the dog wags the tail, and the tail plays an important part, but it doesn't really get a say in where the dog goes or what the dog does. This is equally the case of IT and the business. IT is the tail to the business dog, and IT plays an important part in keeping the business balanced and operational, but it's not in charge. Nor should it be.

IT departments that forget they're a *part* of the business are just as likely to be replaced by the business. More than anything previously, Cloud acutely proves that the IT department doesn't *have* to be a mandatory feature within a business, and that the IT department isn't indispensible. Workgroups and even individual teams, stymied by IT policies and procedures that prevent rapid deployment of new systems or pilot programmes can turn to Cloud vendors who offer monthly billing with limited fuss and overhead. In such situations the IT department can be reduced considerably to a few policy and security advisors.

Of course in these situations where parts of the business turns to the Cloud rather than the local IT department, it can be that there's been insufficient time, budget or resources allocated to the IT department to be able to respond effectively. The best defence IT departments can offer on this is to be facilitators and enablers rather than impediments. That means keeping track of what users are asking for and asking for the budget and authority to provide those services, regardless of where those services come from.

Just as Cloud brokerage services are starting to gain momentum, an IT department looking to stay relevant to its business also needs to be prepared to position this option (as well as the pros and cons of it) to the business when the required solution can't be delivered internally.

In other words, this comes back to the first stop lesson, namely: stop saying no.

# Collaborate

From India comes the story of the blind men and the elephant. Several blind men are allowed to touch just one part of an elephant, and then they have to describe what the elephant is. The one who touches the trunk describes a snake; the one who touches a foot describes a tree trunk, and so on.

An IT department where the staff don't appropriately understand the business is like the blind men and the elephant. Each component, individually, might be able to work with the part of the business it deals with, but true awareness and synergistic cooperation can only come by actually understanding the business.

Thus, collaboration is about the three R's:

- Responsibility
- Rapport
- Responsiveness

It's critical to remember that the purpose of IT within an organisation is to *enable* the organisation to function more effectively. That can only happen when the IT department functions as a collaborative component of the entire business.

#### Responsibility

When we talk about IT departments being *responsible*, the first thought that comes to mind is an IT department behaving *sensibly*. There's no doubt whatsoever that sensible behaviour is critical, but that's not the core meaning of responsibility in this scenario.

The first and most important stage of collaboration is sharing a sense of responsibility for the state and success of the business. Looking at it from a purely fiscal sense, an IT department can't exist without budget from the business, and that budget will be limited (or not even forthcoming at all) depending on the state of the business.

Logically it becomes important for the IT department to be invested in the success of the business. Where the business succeeds and grows, so can the IT department. A business which is failing is unlikely to invest substantially in IT, but will choose to *make do* with whatever resources it currently has (or even consider reducing those resources).

Of course, it's not guaranteed that a growing business will grow the budget and scope of the IT department; what *is* guaranteed is that a shrinking or failing business will be incapable of properly sustaining IT.

It's not possible to merely state: "we're responsible for the success of the business" and leave it at that. Instead, it requires:

- Understanding the goals of the business
- Understanding the strategic direction of the business
- Setting a goal of business enablement
- Ensuring the strategic direction of the IT department facilitates the strategic direction of the company.

The first two aspects can only come through effective communication from the business **to** the IT department. This should come from the board through to the CIO, then down from the CIO to the various managers. What's critical however is that the goals and strategic direction are then communicated (or communicated in a distilled form) to the individual staff within the IT department.

That's not to say the staff in the IT department should be briefed *more* than the staff in the core departments of the company – such briefings will be part of any successfully operating company. These might be filtered down briefings that start at the top and finish with per-group team meetings, or they might be *town-hall* style meetings where as many people in the company as possible get together in one place for a strategic pep-talk.

It's not uncommon to see the communication *stop* at the border of the IT group, and that's a substantial mistake for the business to make. It's increasingly rare to find companies where the IT group is as basic in its contribution to the organisation as say, stationery procurement, yet cutting IT out of that fundamental understanding of the business purpose is the first and most dangerous step in creating a Business/IT air-gap.

#### **Finding focus**

As anyone who has participated in discussions regarding a corporate mission statement would know, coming up with a useful broad statement regarding a goal or purpose is a not the most straight forward of tasks. If an IT department is operating on its own, divorced from the goals and strategy of the business, it must try to intuit the most appropriate goal on its own – and can easily fail at that process. When the IT group is aware of the goals of the company and the strategic direction the company wants to take in order to achieve those goals, setting comparable goals and strategies is substantially easier.

The strategic directions of the IT department should optimally be developed around a combination of the strategic direction of the business and the critical business functions that enable the achievement of those strategic directions.

Consider for instance a company that derives a substantial portion of its revenue via the sale of goods and services online with an eCommerce system. In this scenario, the company might develop a strategy to increase the volume of sales by 500% over the coming financial year. This can immediately translate into IT strategies, such as ensuring:

- The hosting service can handle the increased bandwidth
- The eCommerce servers can scale to meet the increased load
- That availability targets are made more rigorous for the environment

The great thing for the IT department is that it's not really rocket science working out those strategies, because they're not isolated from the business strategies. With the business strategies defined, defining the IT strategies becomes a simple process of *enablement*. Furthermore, with IT strategies aligned to business strategies, it becomes substantially easier to present a case for the appropriate budget levels. Instead of budgets being about system availability and Full Time Employees (FTEs), they become statements around meeting aligned business strategies, a language the senior business managers and board can much more readily understand.

An IT department that tasks itself with the responsibility of a successful business is most aligned to the business, and therefore most *helpful* to the business. Once an IT department decides its core purpose is enabling the successful operation of the business, everything else relating to collaborating with the business falls into place.

### Rapport

The second stage of collaboration is building a rapport with the business units, and developing an understanding of the work the business does.

IT departments usually deliver a host of services to the business, such as:

- Email
- Intranet
- File servers
- Name resolution
- Printing
- Databases

Regardless of how many other services that are provided, one thing will always be abundantly obvious in most businesses – the services the business *expects* aren't necessarily the services the IT department *provides*. That's not to say that IT provides wholly inadequate or irrelevant services. Rather, an IT/business disconnect can occur when we look at the services in the way of how *business* perceives them versus how the IT department perceives them.

Consider email. For some users in the organisation, *email* may be deemed a service, but by and large, email is part of the broader *communication* service the business requires. Phone systems, web access, email, instant messaging and so on all fall under the umbrella service of communication. That may be a named service, but equally it may be seen as an *enabling* service for the actual business functions, such as:

- eCommerce
- Customer contact
- Phone sales
- Post-sales support

Those services are the *real* business functions, and they're the *real* services the business is looking to the IT department to actually enable. Understanding there's a difference between IT services and the services the business requires of IT is where the rapport comes in. It allows the IT department to *map* IT definitions of services to business functions, and as a result, present an interface and description to the business of the services which it provides *in the language the business understands*.

That's step one. Step two is an appreciation of *what* the business does, and *how* it goes about it.

If the members of the IT department don't understand the broader processes involved in the eCommerce business function, or the customer contact business function, how effectively can they provide services to enable those functions, *and* prioritise the diagnosis and repair of those services when they experience failures? There may be some good guesses, and there may be some longer-term experienced staff that can point people in the right direction, but it's not necessarily something that's going to come naturally.

In order to best provide services to the business (that in turn allows the business to fulfil its functions), the IT staff and management *must* have sufficient knowledge of how those functions work.

There are two distinct solutions to this problem:

- Documented procedures
- Trained staff

The first is somewhat less controversial. It's reasonable to expect that at some point, the processes involved with the various business functions should be documented. As new IT staff come on-board, part of that induction process should involve studying the various functions of the business they'll be required to interact with and support. This occurs by first seeing what those business functions are, then having a broad view of the processes involved in fulfilling those functions.

It's important for IT staff to know how to do their own jobs, and we've already discussed that standard business staff shouldn't be expected to know how to do IT work. However, the reverse is not really the case. In order for IT staff to successfully support business functions, it's essential they understand how the various business functions operate.

Beyond documentation, the second option (training) is a little bit more controversial, but I personally think more important. Where possible, new IT staff should *learn* at least some of the business functions they'll be supporting. That starts with reading the documented procedures, but it finishes with actually *using* the systems to gain a tactile understanding of how they work.

In a call-centre, for instance, that would involve the IT staff getting on the phone and taking real customer calls. In a sales organisation, it might involve IT staff sitting with a sales administrator and watching the data entry process.

This doesn't necessarily have to be a lengthy process, but it's important to create true rapport (and perhaps even empathy) for the tasks at hand. It's one thing to read a process manual that describes how to do a task, but it's an entirely different thing to actually *do* the task. An interface on screen being used *for the business task itself* is substantially more tangible than a series of screen-grabs, and encourages a better connection between the tasks performed by regular business users and the actions the IT group have to do to support those tasks. Further, this often bridges the gap between understanding whether systems are *present* or whether they're *available*, in terms of the end-user's perception.

Finally, rapport is critical to ensuring user concerns are not trivialised:

Developers at a software company had long rejected calls by power users and partners to allow editing of a configuration item related to hardware used by the application. If any aspect of the hardware changed, the configuration for the item would need to be completely deleted and then re-created from scratch within the application. This was a tedious function that could take 10 or more minutes, and an error at any point would result in failure and needing to restart the process.

Despite numerous RFEs and bug requests being filed, developers insisted that it wasn't necessary to be able to edit the configuration – it was simple enough to delete it and recreate it.

This went on until such time as a developer was required to attend a major customer site for a critical incident, and while on-site had to delete and recreate the configuration dozens of times.

The next version of the software included a configuration editor.

Rapport allows IT staff to better understand the real-world impact of more subtle issues that plague users. Hearing that it takes three minutes to jump between pages in a customer order form by itself may not strike a chord, but having *used* the form and knowing that on average it's necessary to jump between pages 4 times to complete the order gives the rapport necessary to better prioritise finding a solution.

There's another benefit to building the rapport this way – since the core business participates in the training of IT staff, it becomes well understood that the IT department is significantly invested in understanding business requirements and processes. This can help to eliminate "them and us" segregation between the business and the IT department, thereby improving overall cooperation and adding to the sense of shared responsibility.

Finally, it's worth keeping in mind that when IT staff understand how the business processes work, and how their systems support those processes, it will be easier for them to spot where improvements and efficiencies can be made. Instead of IT systems lurching forward only when big changes are made, a culture of continuous improvement that focuses on business efficiencies can develop.

### Responsiveness

Responsiveness originates from a rapport with the business functions (and the people who fulfil them) as well as a shared sense of responsibility towards the success of the business. In itself though, that isn't enough – an IT department that functions solely on the rule of *drop everything to work on a customer issue* is rarely going to engage in proactive measures, and its staff will constantly feel pulled in a dozen or more directions simultaneously.

Equally, an IT group that focuses on internal tasks as the highest priority while pushing customer issues down to the bottom of the work pile isn't going to work effectively with the business. There has to be a middle ground.

Truly effective responsiveness is developed through documented service level agreements (SLAs) with the business, and in turn the appropriate contracted vendor SLAs and intra-group operational level agreements (OLAs).

When a system fails or experiences issues and there are no SLAs, it's a potluck scenario where one of three different things can happen:

- The IT group responds appropriately
- The IT group responds too quickly, jeopardising responsiveness to other systems and issues
- The IT group responds too slowly, diminishing the capacity of the business to perform the function that has failed or is experiencing issues.

Consider the above list – without SLAs it becomes immediately apparent that it's entirely plausible for the IT group to *consistently* respond ineffectually. Indeed, if the response is random or performed by a different person each time, it's entirely possible that it'll be appropriate only a third of the time.

This becomes even more problematic for an IT group tasked with simultaneously supporting internal and external customers, and therefore must be managed very carefully. In such situations, it's not only important to establish SLAs for response, but actually establish *comparative* SLAs for responses, so both internal and external customers can be serviced at the appropriate priority at all times.

#### Who gets support?

Many smaller companies that perform IT support for their customers end up allocating the support of their own staff to the same help desk.

This is usually done as a perceived efficiency. After all, if there are dedicated help desk staff available within the company, with a broad technical knowledge base, then they're likely going to be able to perform internal support with minimum fuss. On paper, it looks like a win/win scenario for the business.

If not managed correctly, it can be disastrous. In such scenarios, if internal customers are ignored at the expense of external customers, the external customers are happy but business users may be left to fend for themselves when experiencing issues. If external customers are ignored at the expense of internal customers, the business is likely to lose money as dissatisfied customers look elsewhere for better service levels.

Unless a business is incredibly careful with setting up stringent issue prioritisation, and establishes employee numbers against both internal and external customer requirements, this sort of policy will result in miserable users, miserable customers, or both.

So for most businesses in this situation, there comes a time for a business that has grown from a very small size to bite the bullet and either shift their internal systems to the Cloud so all staff can focus on external customers, or to hire dedicated internal IT staff that are independent of the customer help desk(s), regardless of how well meaning the members of those teams are.

Entrenched knowledge and business function awareness will *help* to achieve better responsiveness to issues, but the only way to measure and guarantee responsiveness is to document the service level agreements for the various functions. In the case of either cascading issues or multiple issues, it allows the IT group to prioritise the triage and repair activities. Knowing that one system has an SLA of 4 hours to become operational after a failure and another has an SLA of 24 hours allows for *easy* prioritisation of the activities to be performed if *both* fail. Without SLAs, responsiveness may simply come down to *who shouts loudest*.

The SLAs also help IT staff understand when business focused tasks should be worked on in relation to back-end tasks. It may seem that a broad task being worked on for IT systems is important compared to an individual user saying that another system is *slow*, but if the broad IT systems task is part of a 6-month project and the system being reported as slow is *critical* to successful business functions, it becomes readily apparent that the former should be set aside to resolve the latter.

More than just determining which item should be worked on first, SLAs allow for a more intelligent processing between the *three* functions of:

- Remediation (repair to workaround)
- Resolution (full problem solution)
- Business as usual

It may be that with a true understanding of the responsiveness required, an IT team member can work first towards achieving remediation through the establishment of an acceptable workaround, *then* return to business as usual activities, and later approach a full solution to the initial problem request.

## Listen

The words of the Greek philosopher, Epictetus, neatly sum up the central lesson in this topic:

We have two ears and one mouth so we can listen twice as much as we speak.

Communication isn't a one-way street, and can only happen when everyone is prepared to listen. Many of the apparent disconnections between the IT department and the business will come from this simple mistake.

#### Listen to yourself

Years ago I spent a few months attending the public speaking organisation, Toastmasters. Time and distance meant I couldn't continue, but it was a fascinating experience. The Toastmasters chapter I was lucky enough to go to had an amazing speaker – she could stand and talk for half an hour with only a few moments warning on just about any topic ... without using any verbal fillers. Sometimes, just very rarely, one "um" or "ah" slipped in, but it was rare.

She'd achieved that by listening to herself and taking in feedback on what others told her.

Over the years I've sat in a large number of IT offices, listening to IT staff as they go about their daily work. System administrators, application and database administrators, team leaders, help desk staff, network and storage engineers. If you can think of an IT role, chances are at some point I've sat beside them while they worked.

That means I've sat beside them listening to them talk to their colleagues, and their customers, regardless of whether those are internal or external customers. The results weren't always pretty.

Many who work in IT have considerably more IT knowledge than their customers/end users. In that sense, it's no different from any specialist position, be it medical, mechanical or any number of a raft of other areas.

What seems particularly unique in IT is the ability for many IT workers to come across as extremely exasperated when explaining concepts to non-IT workers. As much as anything, this comes from enunciation patterns in speech.

Consider, for example, the challenge of dealing with someone whom you're trying to explain a solution to, but gets easily distracted and either goes off on a tangent or attempts to jump ahead in your explanation.

The solution is to calmly take control of the situation – with emphasis on *calmly*. Instead, in so many situations, the exasperation shines through like a white-hot beacon, and instead what the user hears is something along the following lines:

"Yes I KNOW you got an error But I NEED you to STOP and LISTEN and do EXACTLY what I tell you."

Or, equally bad:

"LISTEN. Just STOP. STOP doing that. YOU need to WAIT until I've FINISHED explaining."

And:

NO, NO, NO! STOP. I NEED you to STOP. You're wasting MY time.

Those aren't theoretical sides to a conversation, by the way. They're actual examples of IT staff talking to their end-users I've overheard.

Under such circumstances instead of sounding like a professional, one ends up sounding like a prissy and condescending school-teacher talking to the least-favourite student. The least favourite student hears that tone, and so does the end customer.

Take time to listen to yourself. Unless you're working in a call centre it's unlikely your calls are being recorded, but there's nothing preventing you from using a smart phone or digital recorder to just record *your* side of the conversation. Capture a few conversations and play them back to yourself later, outside of work, and listen to how you sound.

If you're communicating incorrectly and not taking the time to listen to yourself, the chances are that the customers will stop listening to you too.

#### Listen to the customer

By the customer I mean either an actual customer or an end-user.

Non-IT workers in particular can at times have different interpretations of errors and situations that may occur, and a useful way of expediting the communications flow is to pick up on those interpretations where possible and run with them.

For instance, it doesn't matter if an IT professional works with files and *directories*. If the end user works with documents and *folders*, then the conversation with the end user should be about documents and folders. That way, the end user can concentrate on what he or she needs to say, and hopefully have a greater chance of remembering the important details.

Listening to the customer is more than just allowing the customer to work within their own nomenclature framework. It also includes:

- Validating the user by giving them time to talk
- Taking the time to understand what they say

Validation is important. Very few people could be said to *enjoy* being cut off or interrupted while they're speaking – keep in mind how we feel if we get cut-off mid-sentence as we try to describe something.

Imagine a scenario such as the following with your doctor for instance:

You: I've got this funny cough that...
Doctor: Right, so you've got a cold
You: No, I've got this funny cough that only happens...
Doctor: Of a morning? Probably sinus related
You: No, I've got this funny cough that only happens when I've been
Doctor: Ah, running? You need to hydrate more when running.
You: No, I've got this funny cough that only happens when I've been in the shower and...
Doctor: You can't be allergic to water, you must be imagining it.
You: ...

Those sorts of conversations aren't uncommon between IT departments and end users.

There are two distinct phases to listening: hearing and comprehending.

It's difficult to comprehend something if time isn't taken to hear it in the first place. Even after it has been heard, time has to be taken to comprehend it.

There's no doubt that at times customers will start explaining an issue or a scenario that's familiar, and it's possible to short-cut the problem description. It's important even in those situations that the customer remains validated – that their input is welcomed. In such scenarios it's important to focus on open-ended leading of the customer.

That means not only explaining what is *considered* to be the issue, but also making sure the customer feels comfortable disagreeing with the assessment. If the customer *does* disagree with the assessment, it's important to then give them the time to be heard rather than immediately jumping into the next probable cause.

Knowing when it's acceptable to interrupt someone who is talking is a sometimes difficult to acquire skill. When engaged in a face-to-face meeting, body language quickly tells us whether the person we're talking to is frustrated at being interrupted. Over the phone that becomes a little more challenging. Indeed, this is what can lead to the sort of situations as described in the previous section:

"Yes I KNOW you got an error But I NEED you to STOP and LISTEN and do EXACTLY what I tell you."

There's a simple rule of thumb if you're unsure whether it's not OK to interrupt someone: *don't*. It can sometimes be very difficult to tell whether someone has paused because they're not sure what to say next or paused because they're gathering their thoughts. In the first instance, it would be more acceptable to interrupt. In the second instance, it's not.

If it is necessary to interrupt a customer, it's important to phrase it respectfully so they feel validated. Again, the following isn't an acceptable way of interrupting:

"I NEED you to STOP and LISTEN and do EXACTLY what I tell you."

Instead:

"Sorry, I don't want to interrupt, but from what you've described so far I think I know what the solution is. Do you mind if I quickly outline it and see if we can solve it?"

The net effect is usually the same, yet in the former instance the customer will almost invariably feel they've been trampled over in the conversation, and in the latter the customer will feel validated – the customer will feel that they've been *heard*.

#### Listen to your colleagues

It's important in IT that we take the time to listen out for each other. The best IT departments succeed by being a collaborative, knowledge-sharing team. In such environments, there tends to be a strong *crowdsourcing* of solutions, and a tendency to speak up when an action someone else is doing can be meaningfully contributed to.

Yet it shouldn't be limited to just IT functions. Just as someone might hear a conversation where they can participate and add technical value to, in an IT department that is focusing on improving communications with the business, so too can people work on coaching each other.

Colleagues who participate in this subtly coach each other on better communications styles. In the same way that organisations like Toastmasters are highly focused on collaborative appraisals, so too must be the IT organisation that wishes to closely align itself to and communicate better with the business.

One of the lessons in "Collaborate" was about *responsibility* – having a shared responsibility for the success of the business. This isn't limited to just the core business – it equally applies to the IT/business relationship. Consider again the service industry, where businesses such as restaurants and cafés live and die by (amongst other things), the quality of the relationship with their customers. In such situations, if one of the employees overhears or observes another employee behaving improperly towards a customer, chances are the behaviour will get called out.

This shouldn't be interpreted as creating a department of tattletales. Such reporting would only lead to paranoia and conflict *within* the department – and a department that is internally conflicted can't hope to present a united collaborate approach to the business. However, everyone should, via that shared responsibility, feel comfortable with mentioning their concerns to one another if observed or overheard behaviour feels deleterious to the relationship with the business.

Of course, one doesn't listen to colleagues merely to spot check – it's also a very effective learning tool to pick up clues on where your own communication may need to improve. Hearing a colleague for instance dealing with a specific end user on the same problem you dealt with previously can provide valuable insight into how different people approach similar scenarios.

# **Habits and Cultures**

Whether individually or collectively, the majority of our behaviours are derived from habits. We communicate in a particular way because that's how our communication has evolved, and the communication style becomes habit forming. At a completely electrochemical level, a habit is a path of synapses that fire more readily than another path.

This means it's important to understand that *breaking* habits isn't going to be easy. It's not possible to simply say, "From today we're going to communicate better with the business" and expect that change will be immediate. Change won't be immediate. Change won't necessarily even be enjoyable. Yet change *is* necessary.

As a consultant one of the lines I most often hear from IT groups is:

"Our business needs are unique."

In reality there are very few businesses whose needs are unique. In almost all cases, when someone states that, the underlying message is quite different:

"Our business is change resistant."

Affecting change is not straight forward, particularly when it involves people. Changing back-end technologies (e.g., switching from one server provider to another) is relatively simple. The entire process of the change can usually be condensed down to monetary considerations, and then each unit of change is measurable and distinct: replace Server A, replace Server B, and so on.

Changing people isn't so easy. Over time habits become cultures, and breaking down an undesirable culture requires patience, determination and senior management support. A single manager or team leader can drive a certain level of change, but everyone in the hierarchy needs to be on-board.

Replacing an undesirable culture with a positive one is a two-step process:

- Announce the culture that is to be developed
- Identify and fix the habits that create the current, undesired culture

Ultimately, most people thrive on information, and direction is vital to providing the correct information. It's not enough to say, "we're too uncommunicative with the business", because it's not a statement of a desired culture, just a negatively stated fact.

To change habits, to influence the culture in a positive direction, three things are required:

- Goals Where we want to be
- Objectives Things we'll do to get there
- Strategies How we'll get there

Like mission statements and corporate agendas, it's important to spell out what the end-goal is. This should be forward looking, focusing on the desired outcomes in a positive way, rather than reflecting on behaviour or attitudes that aren't desired, or symptomatic of the current environment.

For instance, consider the following culture goal:

"We want to foster a culture of open communication and respect between the business and IT. We want to collaborate so that the broader business goals are met, and we're part of the team."

Defining the desired IT culture is critical because it gives people an understanding of where things are heading. It also gives everyone an idea of the key behaviour and attitudes they need to focus on in order to get there. The above goal implies the following objectives:

- Open communication
- Respect between workers
- Collaboration
- Teamwork

Some or all of those objectives are in fact already going to be met in the IT department, between colleagues *in* the department, and at least partly between the IT department and the business. The purpose therefore is to fully *externalise* those objectives so they're visible and part of the broader interaction between the IT department and the rest of the business.

With the goals and objectives defined, strategies for achieving those objectives become considerably easier to put in place. For instance:

- Open communication via cross-functional and cross-hierarchical conversations
- Respect between workers by taking more satisfaction surveys
- Collaboration and teamwork by offering lunch-time and pre-work training sessions to the business users

With the end-state culture, its objectives and their strategies defined, it becomes significantly easier to break the habits that are counter-productive. This can happen both informally and formally.

Formally, these goals become part of individual performance goals and team performance measurement. As such, it becomes clear to everyone that a successful appraisal is dependent on improved relationships between IT and the business. When these goals are established at a formal level, it's important that they're tangible, measurable ones.

This may mean the introduction of survey or feedback mechanisms between the business and IT so that end users feel comfortable reporting negatives or positives.

Additionally, the goals become part of standard staff mentoring and collaboration. People have to *own* the solution, which means everyone should feel comfortable saying something along the lines of "I don't think this is helping the business relationship".

Real change comes from an ongoing reminder of the need to change. Effective change comes from developing new habits and disposing of old ones, and real change comes from instilling a culture that IT is *part* of the business.

## In Conclusion

It's important to keep in mind that few, if any IT departments will exhibit *all* of the issues outlined in this document. More so, the level to which issues occur will vary by employee. Yet like any service industry, IT suffers the same problem that a single 'customer' getting poor treatment from a single person can taint the entire relationship.

The IT industry, and by extension, the IT department, has been constantly evolving. We are now increasingly dealing with an era in IT where significant areas previously considered as speciality are now rapidly becoming commoditised.

While many aspects of IT remain reasonably complex, the maturity of the overall industry and commoditisation is leading to a new level of outsourcing – the cloud. A common requirement now is for rapid deployment and elasticity of services, and an IT department stuck in conventional processes, budget cycles and communication methods with its business will lose relevance and be left behind.

In order to stay relevant, IT departments must develop a significantly stronger serviceindustry approach to their relationship with the business. This means:

- Making customer satisfaction a critical objective
- Taking the time to understand the business functions, requirements and processes
- Aligning IT goals, strategies and objectives to those functions.

Central to achieving customer satisfaction is arresting behaviour and attitudes that adversely affect that satisfaction. The customer – the business – must be able to feel that the IT department is an *enabler* rather than a *roadblock*. It's insufficient however to simply stop doing the wrong things; the next step is to learn how the business works and to accept that the IT department shares in the responsibility of the success of the business. All of this must be combined with taking time to listen, comprehend and remain continuously focused on excellent communication.

Culture can be changed within organisations so long as management remain committed to and focused on that change. That change won't necessarily be easy, particularly if the IT department has become isolated from its business, but it *is* essential, and must be dealt with as a priority issue.

## Acknowledgements

I'm particularly thankful for the assistance of several people during the development of this document. First and foremost, Darren Woolley for his tireless reading and re-reading of drafts, as well as his excellent attention to detail with sub-editing.

Feedback on the ideas in this document was gratefully received from Stan Horwitz, Justin Warren, Paul McInerney and Johan Robinson.

Of course, I'm grateful for all the customers I've worked within over the years. It goes without saying I've seen some of the 'stop' behaviour in this document at some sites, but I've equally seen amazing staff at every single site I've worked at, and the lessons in Collaborate and Listen have come from seeing those sorts of people in action.

Given a chance, and a little attention, every IT department has the potential to become a corporate star, so I'll also thank my reader for spending the time to read this document.