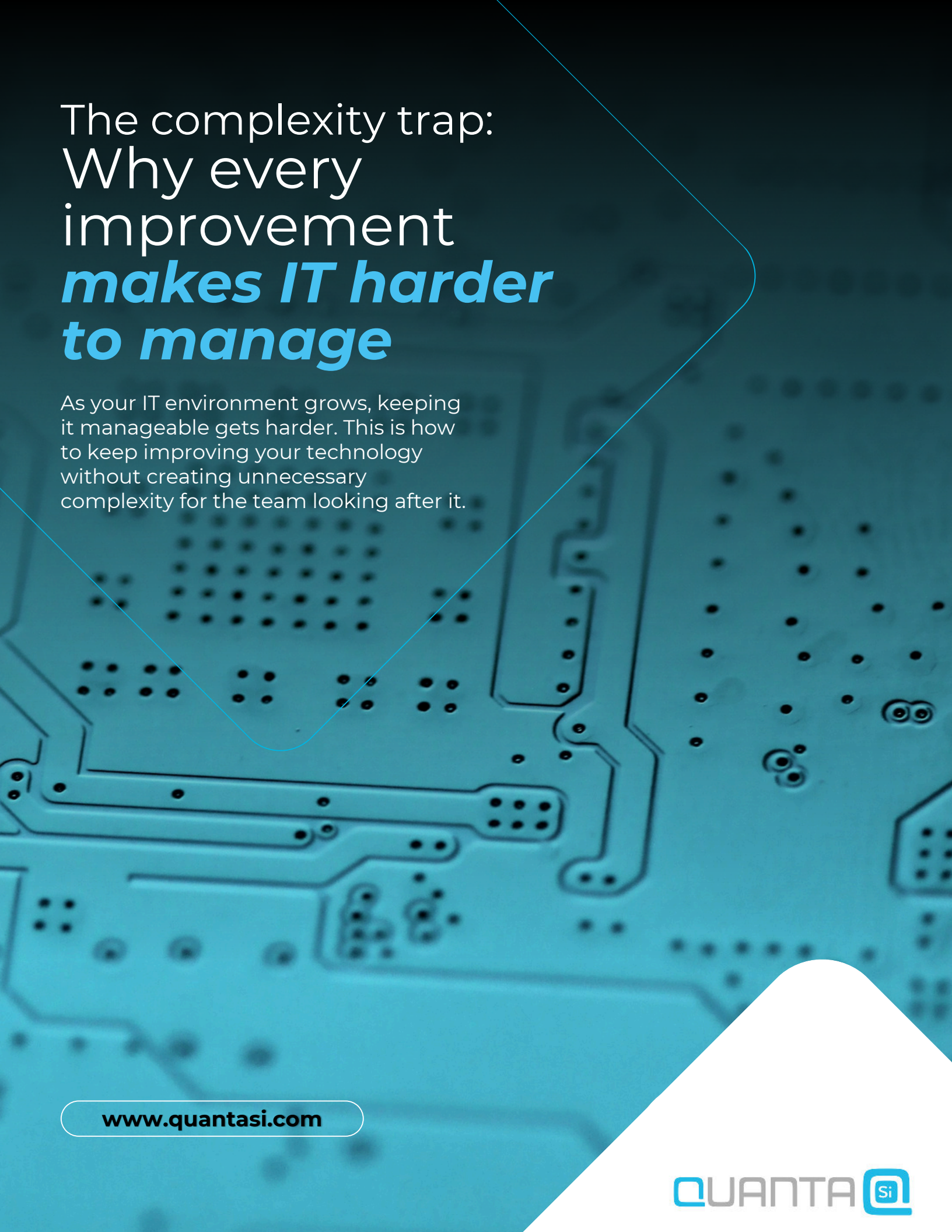
The complexity trap: Why every improvement ***makes IT harder to manage***

As your IT environment grows, keeping it manageable gets harder. This is how to keep improving your technology without creating unnecessary complexity for the team looking after it.



www.quantasi.com

QUANTA Si



There's a frustrating side effect to doing a good job in IT: You can improve the environment and make your own job harder at the same time.

You strengthen security… and now there are more tools to configure and monitor.

You give the business access to a better cloud service… and inherit another set of permissions, logs and dependencies.

You automate a manual process… and gain an integration that somebody needs to keep working.

Of course, you'd probably make most of those decisions again. But carrying all of that at once isn't easy.

An environment that was manageable a few years ago can become much harder to run as more platforms, security tools and integrations are added.

The headcount may not have changed much, but the amount of IT that needs understanding and maintaining certainly has.

That leaves IT directors in a difficult position.

You still need to improve things; standing still isn't an option. But every improvement has a long tail of work attached to it, and eventually somebody must find the capacity to deal with that.

Layered *SECURITY TOOLS*

Layered security makes sense.

If one control misses something, you want another opportunity to catch it.

But when a security stack becomes so broad that simply operating it properly turns into a significant job of its own, it's a problem.

A new tool doesn't arrive empty-handed.

It brings policies to configure, alerts to tune, integrations to maintain and updates to keep on top of.

Someone needs to understand what "good" looks like within that platform, rather than relying on the fact that it has been deployed.

As the stack grows, maintaining that level of knowledge across every product gets harder.

Your internal team may be excellent at running the environment, but expecting the same people to develop deep expertise in every security product they touch is a big ask.

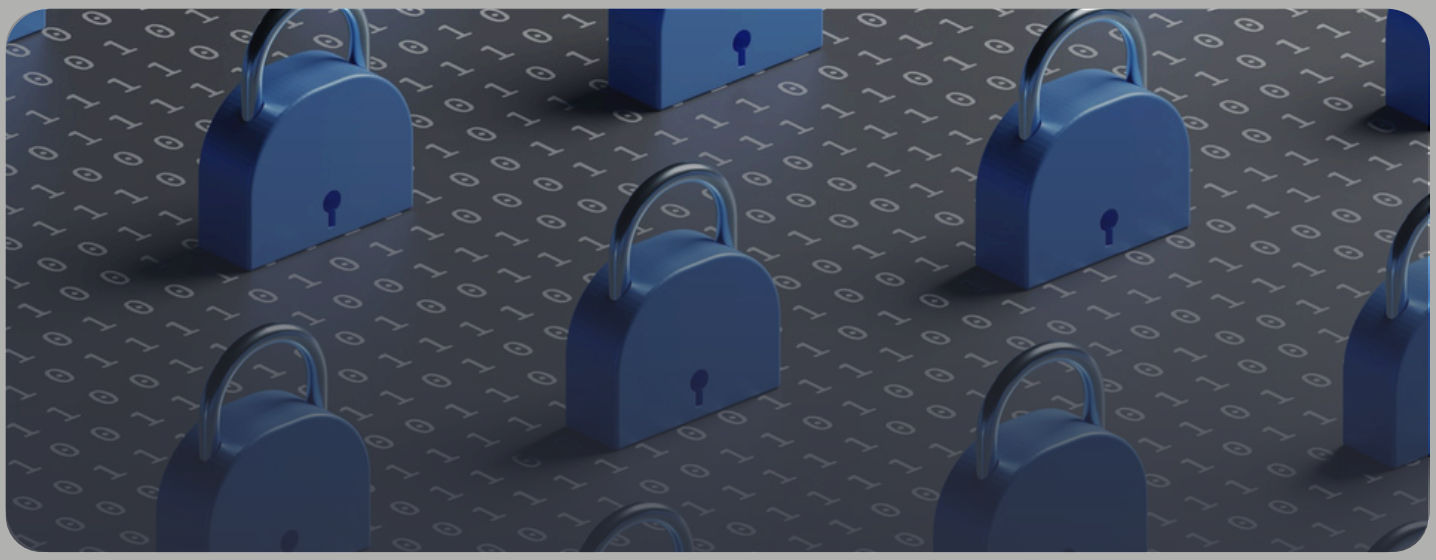
A tool that only needs occasional attention can be particularly difficult because nobody spends enough time in it to become familiar with it.

There's also the question of how the different layers work together.

Two products may detect similar activity but classify it differently. One configuration change can affect another control further down the chain.

An investigation may involve moving between several consoles before you have enough information to understand what happened.

This is where having access to expertise beyond the permanent internal team is useful.



There are parts of the security stack where deep specialist knowledge is needed regularly, but not regularly enough to justify adding another full-time role.

That gives the internal team more freedom to concentrate on the areas where its knowledge of the organization is most valuable, while still making sure the more specialist parts of the security environment receive proper attention.

Layered security still gives you valuable protection.

> ***But you need enough expertise and capacity around the team to operate all those layers as well as you intended when you bought them.***



Multiple ***CLOUD PLATFORMS***

Few businesses set out with the intention of building a complicated cloud estate.

It happens as different requirements appear.

Azure suits one workload, AWS makes sense for another, a department adopts a specialist SaaS platform, and an acquisition brings its own technology with it.

Before long, the internal team is responsible for several environments that need different knowledge.

You can see the problem when something needs changing rather than when everything is working normally.

A permissions issue in Azure may need a completely different approach from a similar problem elsewhere.

Logging, security controls, networking and cost management all vary between platforms, and the services themselves keep changing.

Keeping expertise across all of that is difficult.

There's a big difference between knowing enough about a platform to administer it and knowing it well enough to spot a poor configuration, investigate an unusual problem or make a good architectural decision.

That can leave IT directors with an awkward staffing question.



You may need somebody with deep knowledge of a particular cloud environment, ***BUT ONLY FOR A SMALL PORTION OF THE MONTH.***

Hiring every specialist skill into the permanent team doesn't make practical or financial sense.

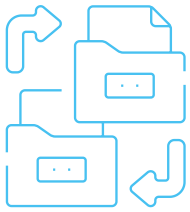
The alternative is to think more broadly about the expertise available around the internal team.

Specialist knowledge can be brought in where the environment demands it, while the people inside the business retain ownership of the architecture and the decisions that affect it.

That becomes even more useful as the cloud estate grows.

The internal team doesn't need to be expert in absolutely everything it operates. But it should know where deeper expertise is required and have reliable access to it when it needs it.

Duplicate ***FUNCTIONALITY***



One of the harder things to spot in a busy IT environment is how many tools are doing variations of the same job.

A security platform adds features that used to require a separate product...

Microsoft includes new capabilities within an existing license...

A department buys an application for a specific requirement, then gradually starts using more of what it offers...

A few years later, you may be paying for the same capability in several places.

Finding that overlap takes time that internal teams don't always have.

When you're responsible for keeping the environment running and supporting current projects, reviewing whether an existing platform could replace another one can stay low down the priority list.

It also helps to have some distance from the decisions that created the environment.

Someone outside the internal team who works across different technology estates can bring useful context.

They may have already consolidated the same combination of products elsewhere, know where a vendor's built-in capability is good enough, or recognize where replacing a specialist product would leave a gap.

That outside perspective still needs the internal team's knowledge.



Two products can look almost identical on a feature comparison while serving totally different purposes inside the business.

Understanding how people use them, what processes depend on them and why they were introduced is vital before anything is removed.

A periodic review gives you the chance to make those decisions deliberately.

It can uncover unnecessary licensing, but the bigger gain may be reducing the amount of technology the team has to administer, understand and keep secure.

Integration *DEBT*

Integrations have a habit of becoming more important than anyone expected when they were first put in place.

A connection between two systems might start as a convenient way to remove a manual task.

A script gets written to move data somewhere useful, or an API is used to feed information into another platform. It works, people begin relying on it, and attention moves elsewhere.

Then a few years later, the business may depend on dozens of these connections.

Knowing exactly what depends on what gets harder.

And it becomes more obvious when an application is replaced or upgraded.

You discover that changing one system affects a reporting process somewhere else, or that an old service account is still authenticating an integration nobody realized was business critical.

Documentation helps, but keeping it current is another job competing for the team's time.

This is an area where extra capacity can make a practical difference.



Mapping integrations, checking ownership and documenting dependencies doesn't necessarily require the internal team to do every piece of the work.

Much of the discovery can happen around them, with the team providing the organizational knowledge needed to establish which connections matter and what would happen if they failed.

Once you have that picture, integration debt becomes much easier to manage.

Older connections can be reviewed, undocumented dependencies dealt with and ownership made clearer.

It also makes future changes less painful. Before replacing a platform or altering an authentication method, you have a much better idea of what else might be affected, rather than finding out when something unexpectedly stops working.



Alert ***FATIGUE***

> ***The more systems you add, the more things want your attention.***

Security platforms, cloud services, backup systems and network monitoring all generate alerts.

Some deserve immediate investigation. Plenty turn out to be routine activity, known behavior or another product reporting something the team has already seen elsewhere.

But working out which is which takes time.

Even with good tuning and automation, somebody still needs to deal with the flow of information.

If your internal team is responsible for monitoring everything as well as running projects and supporting the wider business, a large amount of skilled time can disappear into initial investigation.

That includes plenty of cases where the outcome is confirmation that no further action is required.

This is one part of the environment that lends itself particularly well to a co-managed approach.

Monitoring and first-line triage can be handled by an external team that deals with this work every day.

Anything that needs knowledge of the business, a decision or further investigation can then be escalated to the internal team.

That arrangement needs to be designed properly.

The internal team should still decide what's important, understand how incidents are being classified and have visibility of what is being handled on its behalf.

Regular review is important too, because a noisy alert that is repeatedly dismissed should probably be tuned rather than repeatedly investigated.

Done well, this gives the internal team a filter between the volume of information generated by the estate and the smaller number of events that need its attention.



They remain in control of the response ***WITHOUT HAVING TO PERSONALLY WORK THROUGH EVERY ALERT TO FIND OUT WHETHER IT MATTERS.***

Simplification ***AS SECURITY***



There's a point where reducing complexity can improve security as much as adding another control.

Every application, platform and integration gives the team something else to secure.

There are accounts and permissions to manage, configurations to maintain and vulnerabilities to deal with.

Removing something the business no longer needs takes that workload away.

Working out what can safely go is where it gets harder.

An application that appears redundant may support a process the IT team rarely sees.

Two security products may overlap considerably while still having important differences.

Consolidating onto one platform can make administration easier, but you need to understand what you would lose before making the change.

Doing that properly takes time and a broad range of expertise.

The internal team brings the knowledge of how the business uses its technology, including the history behind decisions that may look odd from the outside.

An external team can add capacity to do the detailed assessment and specialist knowledge across products and platforms they work with regularly.

That combination can make simplification much safer.

The objective isn't to hand an external provider a list of applications and ask them to reduce it. The IT director still decides what the environment should look like and which risks are acceptable.

The additional resource makes it possible to do the investigation properly rather than trying to squeeze it around everything else.

The result may be fewer tools, or it may confirm that the complexity you have is justified.

Knowing that each part of the environment is there for a reason and that the team has enough capacity and expertise around it to manage the whole thing properly is what's important.

You don’t have to manage it
ALL ALONE

Complexity is going to remain a large part of running an IT environment.

New requirements will appear, the business will adopt new technology, and the security landscape will keep changing.

The question is how much of the resulting workload it makes sense for the internal team to carry themselves.



Co-managed IT gives you another option.

You can keep ownership of the environment and the decisions around it, while using external capacity for the areas that consume disproportionate amounts of time or need specialist knowledge you don’t require permanently in-house.



That might mean taking some of the monitoring workload away from the team, bringing in deeper expertise for certain platforms, or creating the capacity to review and simplify an environment that everyone has been too busy maintaining to properly examine.

If complexity is starting to consume more of your team's time than you'd like, let's talk about where additional support could take some of that weight off. ***Get in touch.***

EMAIL: sales@quantasi.com
CALL: (574) 217-4696

