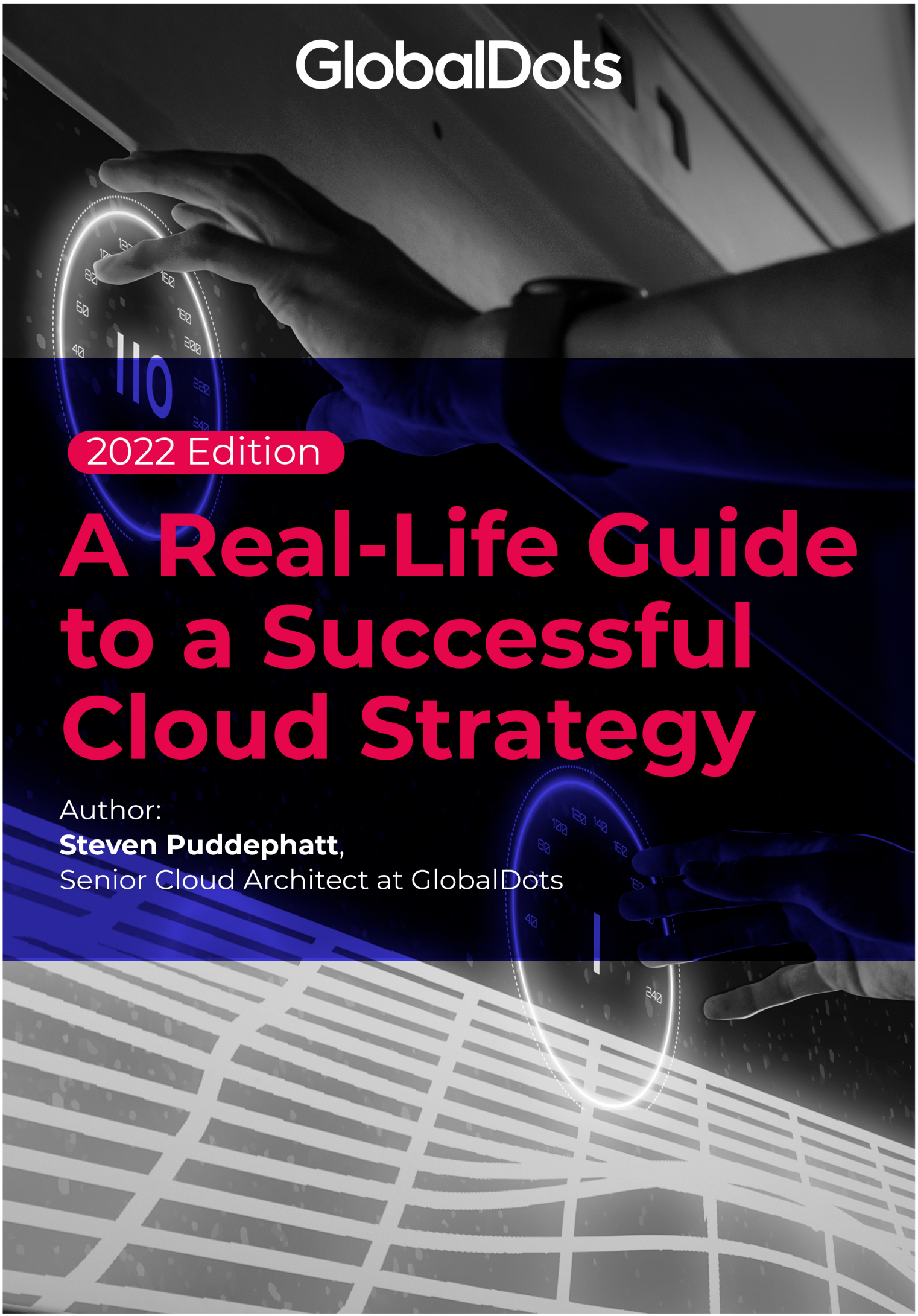
GlobalDots



2022 Edition

A Real-Life Guide to a Successful Cloud Strategy

Author:
Steven Puddephatt,
Senior Cloud Architect at GlobalDots

Table of contents

Executive Summary

This document suggests guiding principles for moving towards a successful cloud environment. It applies to both newly migrated and already cloud-based organisations.

A successful cloud environment is one that provides maximum benefit to business processes, with minimal spend and complexity, and with the utmost security.

GlobalDots has been helping organisations successfully migrate to the cloud for years, and in that time we have seen businesses fall foul of the same mistakes over and over.

It's much easier, cheaper, and more efficient for your business to plan how you will use a cloud estate before you migrate there. In other words, spend more time planning your cloud estate before you let everyone loose in it, thereby turning it into what I like to call a 'double W' estate, or a Wild West estate!

If you have already migrated to a cloud provider, don't worry – the principles laid out here still apply. However, tidying up is a slightly more laborious task than starting from scratch.

Why is a cloud strategy necessary?

Shift problems left:

By spending time now to determine a set of rules (that are enforced by software tools), problems later can be severely reduced. By setting up rules and governance policies you can ensure your cloud journey runs smoothly right from the start.

The most common mistake we see is rushing into the cloud. It's easier to set up a safe landing zone for your cloud resources than it is to try and clean it up once production workloads are already running. Spend the time setting up in advance and make sure your cloud estate is clearly organised from the outset. That way, clean-up operations won't be necessary.

Spend money to save money:

Most organisations are reluctant to spend large sums of money on governance software before they build out their cloud infrastructure. This is actually a harmful mindset and will cost more in the long run.

SREs, DevOps and senior sys admins are some of the most expensive human resources on the planet, and it is exactly these people that you'll be asking to monitor and investigate goings on in your platform. They'll end up writing a complex set of admin scripts in order to ensure IAM roles are used correctly, API access keys are rotated, resources are spun down when they are not needed, and so on.

As your cloud estate grows your IT engineers will be swamped with requests. It's not realistic to expect them to be able to keep eyes on the whole estate. That's why it's so valuable to spend money on governance and security tools at the outset. Adding tools too late in a cloud journey will lead to a bumpy integration and security dashboards will be flooded with unremediated alerts.

Long-term thinking is a must:

Simple cloud platforms quickly become complicated, single VPCs become many and the connections between platforms can become vast and complex. When deciding on a cloud strategy the long-term effects of early decisions must be thought through, as these decisions will determine all the legacy systems, not just the new ones.

A well-governed cloud environment takes careful planning, budgeting and execution. Be prepared to fight the budget holders for money which won't reap benefits for 12 - 24 months. Be brave with proposals and use examples of horrific cloud sprawl (a quick google search will turn up results) to scare money from the company coffers.

Cloud vendor selection

The big warning!

The BIG message you should heed is that for every cloud you add, you increase complexity. It's no secret that there is a **global shortage of good engineers**. So it's crucial that the management team understand that increasing workloads in multiple clouds will create employment difficulties. Multi-cloud strategies can be attractive for a number of reasons, but we recommend you get it right in one cloud before extending into others.

Why use multiple clouds?

For the most part, features across clouds are similar; Kubernetes, databases, storage buckets, etc. Even the more legacy systems (such as AS400) can be converted to run in any of the 3 main clouds. However, there are certain applications that do benefit from a particular cloud, discussed below.

1

AWS - best all round

Amazon is the best all-rounder and has far and away the most features and integrations. It's the easy choice for most enterprises. The footprint is already large, and most engineers have expertise in using it already. Nearly all of the services currently running on-premise data centres can be made to run here.

2

Azure - best for MS services

Azure is now the 2nd largest cloud provider and offers many of the same features as AWS. However, it is really the best fit for enterprise IT applications. It makes perfect sense to migrate your existing Microsoft services to Azure; these include things like:

- Active Directory
- ADFS
- 0365
- Sharepoint

Azure is best suited to hosting Microsoft services, and organisations can convert existing licences to access cloud services at a discount. It's also unlikely that there's a requirement for high latency connections from Microsoft services (such as Active Directory and file shares) to production, customer facing services. So you can afford to have your production workload in one location (not Azure) and have your enterprise IT services (AD and file shares) in Azure.

3

GCP - best for big data

As with Amazon and Microsoft, the standard computing components are catered for. But what makes Google special is their big data abilities. It's for this reason that it can make sense to put data warehouses and business intelligence (BI) tools into GCP. GCP is also particularly friendly on price when it comes to managing and querying these large data sets. They also invented Kubernetes, so you can expect these (managed) workloads to be supported by the top engineers worldwide.

4

General rules for all clouds

Whichever cloud provider you choose, there are certain rules that apply to them all:

- A **Cloud Centre of Excellence** is required to make important decisions
- Good **governance** is required to keep infrastructure manageable
- Cloud **security** requires new tooling
- Cost optimisation should be considered at every step

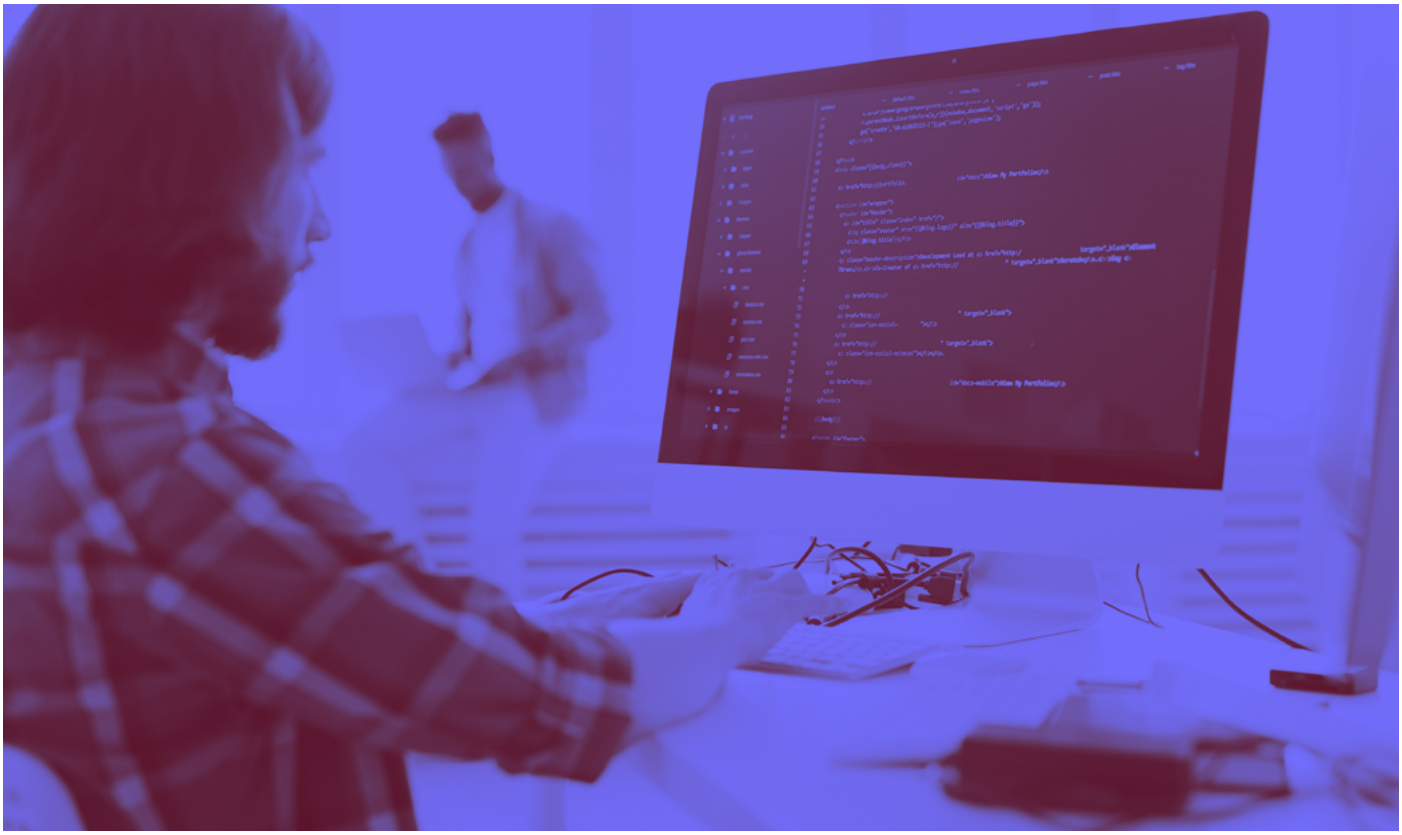
These subjects are covered in detail later.

Migration tooling

No matter which cloud(s) you decide to expand into, there will be a large amount of infrastructure that needs to be migrated at the VM level. This 'lift and shift' of virtual machines can be done manually (by DevOps teams), but it's highly recommended that tooling is used for this.

GlobalDots regularly explores new & leading vendors in the cost reduction sphere. Some of the latest tools even include automated cloud migration offered for free. While established category leaders are higher-priced, emerging competitors might perform identical tasks equally well.

It can be tempting to migrate the VMs manually, but when the number of VMs to be migrated is higher than 10 or 20 it makes more sense to use a tool.



No matter which cloud(s) you decide to expand into there will be a large amount of infrastructure that **needs to be migrated at the VM level.**

Cloud Centre of Excellence

It's common for large enterprises to have a Cloud Centre of Excellence (CCE) where new technologies can be discussed and best practice rules can be established. With larger estates it's necessary to have a team that sits across all areas of the platform.

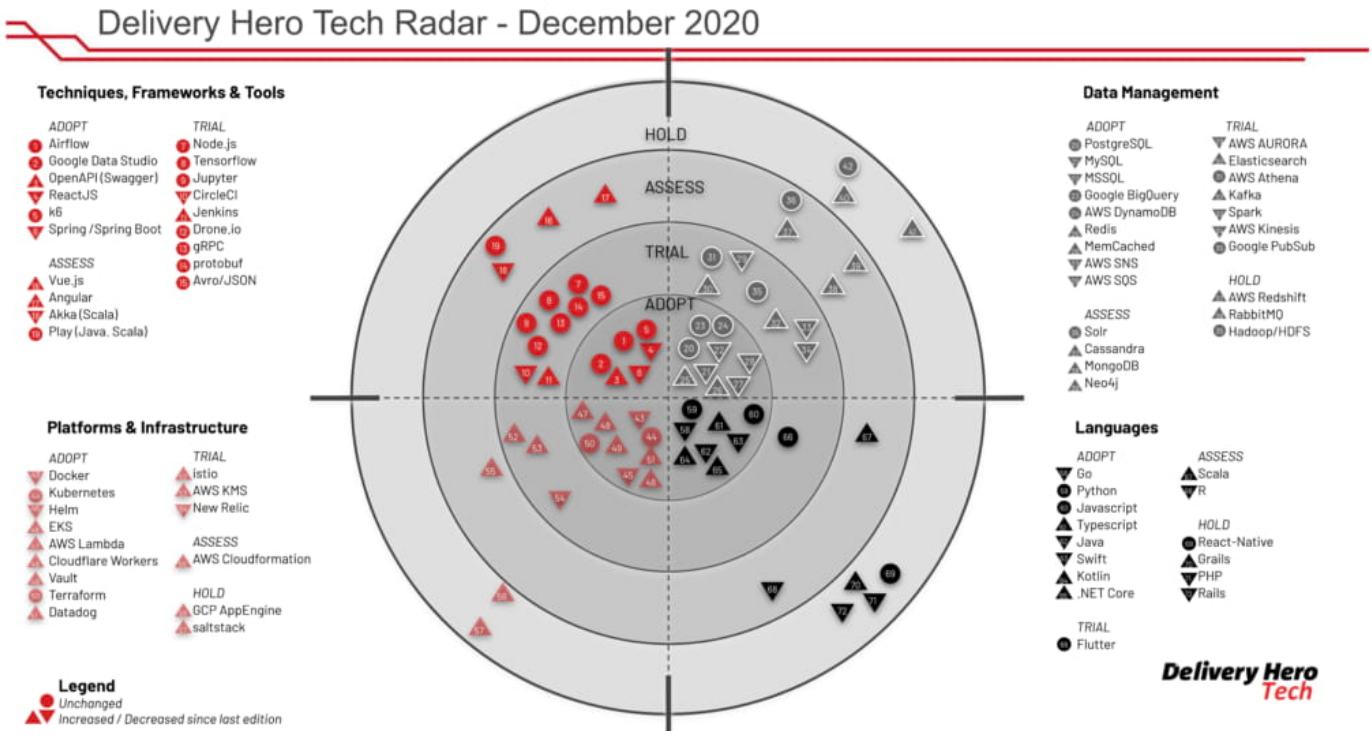
Examples of decisions the CCE might make:

- New technologies to be evaluated
- Which tools to be employed by BUs
- Which apps to move to SaaS and which to remain in-house
- Which technology vendors to partner with

Examples of best practice rules the CCE might enforce:

- All infrastructure to be tagged
- AWS VPC endpoints where possible
- Use VPC peering over VPN
- Multi availability zones only in production environments
- Kuberentes pods to be tagged with BU and purpose
- Kubernetes to run only on spot instances

A CCE would be responsible for evaluating new technology and deciding if it is right for the group. The team would also be responsible for mapping out a technology radar for other teams to follow. Below you can see an example by Delivery Hero that shows which technologies they are on hold with, assessing, trialling and adopting.



The CCE team would be composed of members that have deep knowledge of the various departments within an organisation; they would be responsible for bringing future architecture changes to attention and hold a holistic view of the company.

A CCE team also brings transparency to what both the current estate and the future estate look like. The CCE will also make important decisions that if left unmade would leave projects in limbo. Far too often projects are stalled because nobody in the enterprise has the courage or the authority to make big decisions. In these instances, the CCE steps into this role and keeps business moving forward.

Cloud security

Next generation security products should be part of any cloud strategy, and there should be a security officer in charge of investigating tools to protect your estate. GlobalDots is a world leader in bringing new security products to market and we have been doing so successfully for 15 years. We recommend looking at some of these next generation cloud security tools. The list below is not exhaustive, but covers the most important aspects of cloud security for most organisations.

Zero trust access & SASE

As organisations expand their cloud estate the boundaries of the company will become more blurred. This happens for a number of reasons, including taking on more SaaS providers, more remote workers and the increase of BYOD (bring your own device, i.e. phones and tablets).

It's no longer acceptable to use VPN technologies in order to connect users and it's recommended that organisations look to decommission all VPN access. Look to replace outdated VPN user access with Zero Trust tools, which only give users access to the specific applications they need, and can be linked to permissions in existing directories such as MS Active Directory or a SAML identity provider.

Site-to-site VPN access can remain, as this is more secure, although in reality organisations should look to combine all their networking into a Secure Access Service Edge (SASE). A **SASE** would move all networking configuration to the cloud and replace SD-WAN, MPLS and Branch VPNs. A SASE would also cater for WAN optimisation, Zero Trust access and mobile device access, as well as provide a gateway for SaaS applications, enabling you to lock down tools such as Salesforce and Office365 to only your employees.

Open source security

It's a common misconception that using the 'latest' versions of open source software means that they are safe and/or bug free. As you move toward a microservices based environment it's natural that the number of open source dependencies will increase.

A typical example is Nginx, which is probably one of the most common docker packages used globally. In the **latest test** "nginx:latest has 106 known vulnerabilities", without sufficient tooling you risk unnecessarily introducing vulnerabilities. Luckily, there are tools available to catch these vulnerabilities in the IDE (internal developer environment), CI/CD pipeline and in the code repos themselves, ensuring bugs don't enter production.

Watch our recent video for more information.

Cloud workload & Kubernetes protection

As workloads in the cloud increase, sprawl is inevitable. With new resources constantly popping up, an organisation's attack surface is greatly increased. Trying to keep a watchful eye on so many moving data points becomes impossible, and thus it becomes necessary to use a data-driven cloud workload protection tool.

We at GlobalDots review and assess these tools (**click to watch our CWP demo**) so we are familiar with the latest developments. With a cloud workload protection tool in place you'll be able to spot major security vulnerabilities, over-permissioned users and threat intelligence timelines.

If you're serious about security, we recommend Cloud Security Data Platforms as a **cloud workload protection** tool. This 4th generation of solutions uses machine learning and smart algorithms to spot anomalies and find your security 'needle in a haystack'. Its main benefit is the ability to take billions of events and distil them down into a manageable number of alerts. This emerging category defines its ICP (Ideal Client Profile) as young, growing companies (50-5,000 users) with a minimum of $100K monthly cloud spend.

Dedicated **Kubernetes security** tools are another option to look at. However, at GlobalDots we tend to recommend solutions that allow tooling consolidation, such as Cloud Security Data Platforms. Such a strategy proves more beneficial as the company grows, along with complexity potential.

Cloud workload and Kubernetes protection is an area where organisations will have to budget and plan early to stop attacks and data leaks in the future. **This infographic** might help (switch filter to 'poor security') to see where cloud workload protection would have helped.

API protection

The final part of our recommended new security stack is **API security**, which has been an official part of **OWASP** since 2019. APIs have become the de facto way for systems to interact, and as such are now a prime focus for hackers. Many companies already use tools like WAF protection, but those traditional tools won't catch API based attack vectors. In order to properly catch zero day attacks it's necessary to have specific API traffic inspection.

Cloud governance

What is cloud governance?

Cloud governance is essentially a set of rules to maintain your estate and ensure compliance in a consistent way. One example of this would be not allowing engineers to make a storage bucket that is open to the public. Without cloud governance we may state “no storage buckets open to the public”, but there is no system to actually back up the rule, and so it happens anyway (this kind of thing is in the news every day).

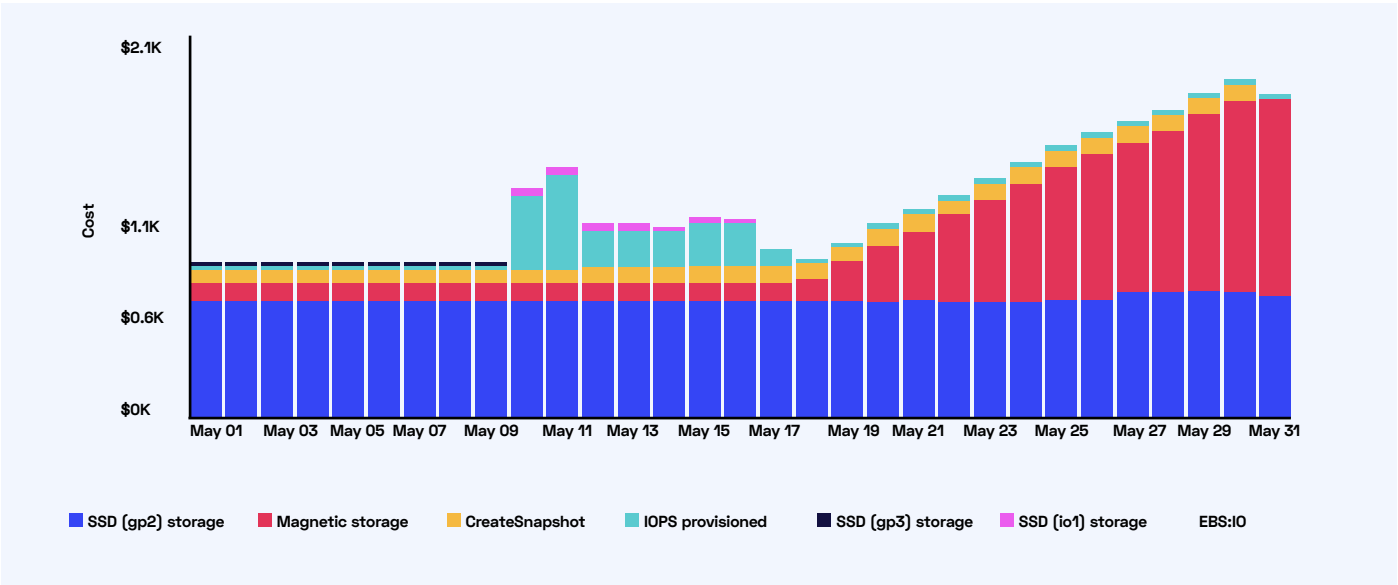
With cloud governance in place an ‘all seeing eye’ will sweep your cloud environment, constantly looking for misconfigurations (infrastructure that doesn’t meet your governance policies). If it finds any such misconfigurations, it will then either send an alert or remediate the misconfiguration. In the example I gave, a storage bucket made open to the public would automatically be closed to public access (unless an explicit exception has been made).

Why do you need it?

Good governance is one of the most important things currently missing from most estates. Without it, simple mistakes is made that either causes costs to spiral or leaves infrastructure in an inconsistent state. The below slide shows one example where a small configuration left behind 5000 EC2 zombie volumes. It was captured as part of an anomaly-spotting project, but with good cloud governance this problem might appear once, then we set a rule and then it never appears again.

EBS anomaly detected

Saving $1,000 / day = $470k / year	Anomaly found - costing **$1,000 / day**	**5000 unnattached volumes cleaned up - launch template modified to fix this moving**



Resource identification

In order to achieve good resource identification we really need to know 3 things:
What services are you are consuming - cloud features
What applications are using these services - company apps
Who owns these services/applications - unit or tech team responsible?

The first two of these require an organisation to have a configuration management database **(CMDB)**. This is essentially a list of all resources you're consuming (configuration items, CIs), ordered into logical layers, as appropriate for your company. Once all CIs have been discovered, it would likely make sense to group them into applications and then assign them to business units (BUs):

Tagging

In order to achieve effective governance of your estate, tagging is absolutely crucial. Each organisation will need to decide what tags are useful for them, but as a bare minimum we would suggest the following top-level tags:

- APP (i.e. Catalog search)
- BU (i.e. Retail)
- ENV (i.e. Dev)

Other tags that would likely be very useful for you include:

- OWNER (specific person or team that owns the resource)
- CREATED_DATE (automatic time/date stamp when something is built)
- CREATED_BY (i.e. Joe Bloggs, if created by specific team members)
- CREATED_PURPOSE (open text field to explain why a service was created)
- JIRA_TICKET (link to Jira ticket explaining why something was made – in theory all of your infrastructure should have a Jira ticket, so it makes sense to link it)
- CONFLUENCE_LINK (link to any corresponding internal wiki information)

By tagging the environments in this way we will far more easily be able to apply rules and budgets to:

- Services
- Environments
- Applications
- BUs

On top of that, engineers will more easily be able to identify what a particular piece of infrastructure is doing and who's responsible for it. As new members join your team this convention will help them understand which infrastructure the apps are being supported by, and who owns them.

It is also possible to have infrastructure auto-tagged where downstream services can be linked to a common resource. An example of this would be tagging an AWS server instance, then using metadata to discover which volumes were associated with that instance, and also tag those. This is a simple example, but much more sophisticated auto-tagging rules could be put in place, e.g. pulling details from git repositories or Jira tickets and applying those to infrastructure, adding much needed context.

CMDB

Creating a CMDB is not a manual task. Using a governance tool, you can easily discover all the resources in your estate and enforce certain rules on the infrastructure it discovers.
Examples of this might be:

- Do certain tags exist, e.g.:
  - APP, BU, ENV, etc.
- Are certain settings enabled, talking specifically about S3 buckets:
  - Encryption at rest enabled?
  - Encryption in transit enabled?
  - Lifecycle policy exists?

With these governance rules in place we can then decide if you want to:

- Skip - Take no action with CI that has been found to be out of policy
- Alert - Take no action but alert the necessary team that there is a rogue CI
- Act - Remediate the CI, for example enforcing s3 encryption settings automatically or removing s3 public access, etc.

As an example, if we see the 'Dev' environment tag, you may want to just alert and not remediate, as things are frequently changing in your dev environment. However, if we see the 'Prod' environment tag you may want to enforce your policy rules.

The key thing here is that all circumstances are captured because all of your cloud estate is being monitored. So if we insist that a certain tag MUST be present on resources, then you'll know when teams do not adhere to this. We no longer need to ask, we will know.

We can also use these governance rules to enforce auto-tagging of resources (discussed earlier), so for example setting an auto-tag of CREATED_BY would ensure every resource created by an operator has their name on it. Again, this is a simple example; much more sophisticated measures can be taken including calling APIs for data, etc.

Budget allocation procedures

In order to create budgets the first thing we need to know is what to measure. Possible examples could be:

- Cost per BU
- Cost per application (e.g. web search)
- Cost of shared components (e.g. Kafka clusters)

It's hard to give concrete examples of budgeting as each organisation will be different, but it's something to keep in mind before lumping too many services together. Shared components tend to cause the most headaches in terms of budgeting and you need to work out how you plan to split these costs ahead of time.

Calculating the shared components

In some cases, calculating the cost of the shared components can be more accurate. For instance, Kubernetes environments can easily be tagged, as can their downstream resources. If a cluster has been made for a particular BU, then this is very easy to then charge back to that BU. In shared resource accounts, accurate tagging of individual pods (where they relate to a BU or APP) will enable you to calculate the precise cost of each Kubernetes cluster and divide it accurately. Again, tagging is crucial.

Other shared resources (such as Kafka or networking) are much harder to split between BUs. In cases like this we suggest selecting a particular metric to measure BUs by (for example total spend in BU-specific AWS accounts) and use this in order to create a ratio by which to split costs. This ratio method of splitting costs is not perfect, but offers a happy medium and gets most of the way there without expending the significant effort required to get the exact number.

Setting limits and tracking teams

Once there is a good tagging methodology set up inside an organisation we can then look to track spend based on tags, e.g., BU, APP or OWNER. In this way if a specific team wants a new feature (let's say it's $10k for a new 'MACHINE LEARNING' feature) we can then track all tags across the system with BU+MACHINE LEARNING. This will allow you to see if the budget of $10k was exceeded.

A governance tool can also enforce budget rules and set hard limits, meaning teams cannot create any more infrastructure if they exceed their budget. This is probably not ideal in all circumstances, but there will be some POC projects in any organisation that would benefit from a hard budget cap.

Procurement process

Methodology to employ

Most organisations will already have a very advanced procurement process, which you can extend to the governance of cloud platforms. When talking specifically about cloud resources that are to be procured it makes sense to run this via Jira (or similar). Jira can be a central source for tracking:

- Who requested the resources
- Why they were requested
- Who approved them
- What steps need to be or have already been taken

As part of a specific request it would also be possible to attach an approved budget. In this way, when resources are provisioned in the cloud they can be tracked by BU, APP, OWNER and JIRA_TICKET. Again, this makes it much easier to trace back why a resource is there and why it was acquired. This is not possible in an estate with resources that are deployed with little or no tracking.

Jira could also be used to hold additional steps that need to be done in order to properly track spend and usage. For example we could use Jira to ensure that before additional cloud infrastructure is purchased it meets pre-set criteria:

1. **Budget is set**
2. **Tagging methodology has been adhered to:**

- BU owner accepts costs
- Budget is agreed
- Budget is signed off
- Governance settings have been decided

Setting criteria that a resource must meet before being procured will ensure more resources are accurately traced, tagged, budgeted for and have governance policies around them.

Of course, you do not want to tie the hands of developers and disrupt productivity, so you may set more lenient rules in development environments. Finding the right balance will require a sensible investigation of the current working methodologies of an individual company to decide on a framework that works for you.

Resource selection

An important part of governance is ensuring the right resources have been selected. While this is a somewhat ambiguous task given the wide range of services available from cloud providers, there are certain policies that can be put in place, for example:

- Choosing the right instance families for workloads (a simple change like this can reduce cost without any engineering work required)
- Ensuring resources are not over provisioned
- Only using a single availability zone in non-production environments
- Utilising spot instances where possible
- Utilising lower cost storage buckets for non-critical data
- Shutting down non-critical infrastructure at night

In all of these cases a governance policy can be created. For example, if you decided that use of 'm5d.24xlarge' instances was not allowed because they are too expensive, you could then set up a policy that scans the estate looking for these instances and takes action. Examples of actions taken might be:

- Notify tech teams there is a non-compliant instance
- Delete the instance
- Downsize the instance

Again, each organisation has to decide what policies are right for them, but having no policies will inevitably lead to a Wild West cloud estate!

Ensuring the right resources have been selected

Summary

The cloud can enable businesses to grow rapidly but without a good strategy you'll end up in a mess.

Technical staff are required to build a competence centre or a 'Cloud Centre of Excellence' (CCE). With this team in place better engineering and technical decisions can be made on behalf of the whole business. The CCE should oversee enterprise-wide technology decisions and responsible for introducing good governance of those tools.

By bridging the gap between business units, technical teams, and management, a CCE would make sure the right decisions are made, reducing technical debt and standardising the tools and technologies used in cloud environments.

Only through good governance, good tooling and good teamwork will an organisation achieve an optimised cloud environment. Time, money and effort needs to be expended up front in order that the future cloud environment is flexible, tidy and well governed.

Additionally, most tech stacks remain unprotected at some level. Therefore, there is a specific importance to cloud workload protection and open source vulnerability detection. We highly recommend organisations look at tooling to address these common blind spots.

There is an ever-growing variety of cloud solutions for performance and security. The right ones for each business are a needle in a haystack. Therefore, consulting an impartial cloud technology partner can save much of the investment in research, evaluation, and proper implementation.

Example toolset appendix

Below is a summary of example technologies mentioned throughout this document

A

API Security

Application monitoring

C

Cloud cost reduction

Cloud Governance (SDO)

Cloud migration

Cloud Workload Protection

I

Identity Management

K

Kubernetes Security

L

Log management

O

Open Source Security

P

Passwordless Authentication

S

SD-WAN & SASE

Z

Zero Trust Access

GlobalDots Your Tech Innovation Partner

GlobalDots is a world leader in discovering and implementing cloud & web innovation. Over the last **17** years, GlobalDots enabled streamlining and smart growth in over **500** business customers, providing enterprise-grade web performance & CDN; Web Security & anti-fraud solutions; DevOps & Cloud services; Cloud Security; Corporate IT; Cloud-native networking and infrastructure.

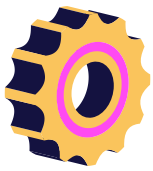
Our vendors range from world leaders to innovative, cutting-edge startups.

Our seasoned engineers test & master each solution's capabilities, pros, cons, and best practices. This allows them to quickly spot your perfect fit of technology and enable fast, smooth adoption.

What makes GlobalDots the best choice for a technology partner?

Innovation Hunters

Constantly tracking the industry to provide spot-on solution s for your ecosystem.

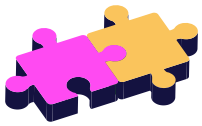


Vendor-Agnostic

Our ever-evolving portfolio and customizable solutions cater for each unique use case.

Streamlining Technology Adoption

Breezing you through from selection to deployment, exhausting every feature to your business benefit.



Holistic, Business-Oriented Approach

We align your IT architecture with your business profile, use case and goals focusing on what matters in terms of complexity and financial impact.

Do you want to know more? Contact Us