Tips for taming the public cloud

Strategies for reducing complexity, lowering costs and improving IT governance and security
IT governance in general is complex. Cloud governance is even more so because the whole point of cloud is to give up some level of control to your application developers. That means instead of a small handful of trusted admins performing every action, a wide range of individuals in many different roles may have self-service capabilities.

Organizations follow a variety of models for maintaining governance over IT environments, and they all share the common denominator of being complex and difficult to truly understand — and put management and governance seemingly at odds. For example:

- **Governance** is meant to ensure that things are cheap, easy to manage and secure.
- **Operations management** is making sure things are working the way you want them to while governance is making sure that things are easy to manage.
- **Security management** is making sure that you keep private what should be private while governance is making sure that it is an attainable goal.
- **Cost management** is controlling and reducing spend while governance is making sure that people follow rules that make cost management achievable.

Note how each element splits management — which is defined as the actions a team takes daily to make things work the desired way — from governance, where a set of rules is defined and enforced to make management possible and efficient.

Ultimately, IT organizations should keep three goals in mind: Make it easy, make it cheap and make it secure. These goals are highly related. For example, overly complex environments (failure to make it easy) are difficult to secure and make it hard to find ways to reduce spend. This insight especially applies to multi-cloud environments, where an enterprise might have two major public clouds, with multiple platform-as-a-service (PaaS) providers, in addition to a couple of virtualization platforms, disparate automation and private cloud stacks, and more. Finding the right mix of choice and control is hard. Clearly defining your governance goals makes it a lot easier.

**Make it easy**

First, to make managing an environment easy, we need to reduce complexity. Some of the simple examples of where complexity creeps in are OS options, network configurations and the authentication mechanisms allowed.

For OS options, it is best to limit the choices to a handful of approved standards across the business and make it difficult to seek exemptions. Then take that option and bake in a single, standard way of deploying that OS. We get into a rabbit hole here quickly, with prebaked images vs. deploying components at provisioning time, but regardless of how you approach this problem, you want to define a single standard and stick with it.
Next is consideration of network configurations. The biggest single thing you can do for network governance is to have an IP address management (IPAM) tool that can be used across your cloud environments. This stores information about what networks are in which environments and used for what purpose. This information becomes invaluable for managing the environments.

Finally, we come to authentication. One of the biggest issues with managing multiple clouds and different environments is managing credentials. Having a governance standard that defines a single identity provider per class of user (i.e., one for employees, one for customers, one for IoT devices, etc.) makes management a lot easier.

These are just a few examples of reducing complexity by defining standards as part of your governance regime. The next step is ensuring compliance. This can be achieved through the use of either cloud-native tools such as Amazon Web Services (AWS) Config or a third-party security information and event management (SIEM) tool.

**Make it cheap**

The next high-level governance goal should be to reduce costs. Some basic guidelines are essential from day one:

- **Set limits.** Limit the types (sizes) of instances (or servers) that can be created by users. Both AWS and Azure have a lot of options, but due to the way that cost controls work in both clouds, limiting users to a handful of options (four or five) makes cost management drastically more effective.

- **Actively manage development environments.** Define and enforce what a development environment actually is. Do you need a high-availability database for development? Probably not. Same with high-performance clusters that are better suited to preproduction/quality assurance where load testing occurs. Maybe everyone insists that Oracle Enterprise Edition is needed, but it probably isn’t, and staff members could get by just fine with Amazon Aurora, which would save money. This plays into the same themes as the previous section about making it simple. We want to reduce choice intelligently without stifling innovation.

- **Tag resources for tracking.** One thing that is very important is tagging all your resources to make cost allocation easier. (This is very much a governance goal.)

- **Evaluate the business value.** Create a standard for how to express the business value of a project. Have that model include elements such as opportunity cost — if another platform was used, for example, which would cost more development time? Also include the value of common platforms’ features to the specific project and how they influence the business value of the project. Yes, it can be difficult, but this step lets you make informed decisions about what to allow, where and when to allow it and why — based on what’s best for the business.
Make it secure

Finally, the third high-level governance goal is security. Governance always prioritizes simplicity, visibility and compliance, and these are especially important in securing multiple environments.

Enterprises should take a two-pronged approach by (1) intelligently limiting options at the front end so that obeying the rules is the easy option and (2) providing visibility into who is doing what to ensure that employees are making compliant decisions.

Following the good governance practices described in this paper will improve an organization’s ability to secure the environment. A simple example is how limiting OS standards makes it easier to patch, harden and monitor activity on those operating systems.

Standardizing identity providers and the view of network information allows you to more easily understand the who and where of actions being taken. Reducing the deployment options in development for cost-control purposes also reduces the types of acceptable events you expect to see in your SIEM solution, which drastically simplifies your security engineers’ work.

By making the environment simpler and easier to understand for an engineer or architect, the entire security discussion becomes far simpler to address.

The right balance

By thinking of governance as the act of defining and enforcing policies and standards that make management easier, you can focus at the front end on creating the right balance between choice and simplicity.

One word of warning: Don’t let perfect be the enemy of good. There will be valid exceptions to every rule, and defining how to determine when an exception is warranted is an important piece of your governance strategy.

About the author

Eric Moore is chief technologist for the AWS Integrated Practice at DXC Technology. He is experienced in cloud computing, architecture and automation, with a background in operations, business consulting and security.