



Planned Nimbleness: A Practical Guide for Asset Managers

July 2023

INTRODUCTION

Broad acceptance of cloud computing platforms in asset management has provided a wealth of opportunities for improved security, scalability, reliability, and performance. While cloud computing has also lowered barriers to entry for startup managers, pitfalls, and areas of concern in its adoption need to be kept in mind when designing a reliable and agile system. This article will present practical ideas around how best to take advantage of the breadth of services available while managing issues like cost, security, and flexibility, all with the goal of providing robust investment services to clients.

Nimbleness, for asset managers, is a competitive advantage. The ability to test investment ideas quickly and comprehensively, to implement changes to an investment process easily, and to monitor efficacy and improvements allow the investment team to put their best thinking to work confidently, without unnecessary delay. The idea of 'planned nimbleness' refers to how your infrastructure and software are constructed to allow quick, robust changes to your system with high confidence and low risk. One important dimension of infrastructure nimbleness relates to scalability. This is primarily what is provided by a cloud deployment. The ideas presented here are good practices for nimbleness regardless of using the cloud but ease of scalability – and hence use of the cloud – is something that should be considered in planning for growth.

INFRASTRUCTURE AS A SERVICE (IAAS)

A helpful attribute of cloud products is that they provide an interactive console that gives access to all of the services offered. While this is an easy way to explore and experiment with what's available, using the console as a primary interface to stand up and shut down resources can quickly become tedious. Configuring through the console is time-consuming and error-prone because of its repetitive nature. Standardized configuration norms have to be entered each time to go through a workflow. A remedy to this challenge comes in the form of cloud service's ability to script resource management in a way that makes it much less error prone and easier to enforce configuration defaults. There are third party products that allow an architect to generalize an infrastructure across vendors, which helps avoid vendor buy-in. These scripts can be used to stand up the infrastructure in a different cloud region, or even a different platform, for testing, business continuity planning (BCP) or migration.

This kind of IaaS scripting gives the flexibility for codification of infrastructure in a way that is executable, versioned and easily tested. While the console is a useful tool to get familiar with a service and better understand its capabilities, it's important to move its deployment to a scripted implementation as soon as possible.

NETWORK ARCHITECTURE

Security is a critical concern in any cloud deployment and it's important to understand the implications of network architecture design. The default settings provided when creating private clouds, subnets, and compute instances may not provide the level of protection that an organization needs. It's important to decide on those configuration values and use the IaaS scripting functionality to standardize the creation of network enabled components in an environment. Areas to focus on include public IP addresses, VPN, segregation of duties (development vs. test vs. production), and access to the internet.



SOFTWARE ARCHITECTURE

The advent of development environments like Jupyter Notebooks and scripting languages like Python has significantly reduced the barriers to entry in learning to program. This is useful for those working to build tools and algorithms to perform investment research and portfolio management. However, good software architecture becomes even more important as a result of having more programmers of varying skill levels in the mix.

In much the same way that IaaS scripting helps the architect take a step back from using the interactive console as the primary interface, building a software library to perform low-level repetitive tasks can alleviate the risks of having developers re-invent the wheel every time they start a project. Investment professionals span a wide swath of programming skills. It's not surprising to see beginner programmers working alongside programmers who have been coding for decades. By abstracting away low-level functions, a common library allows developers to focus on what's important to the investment business logic.

Automation around library packaging and deployment makes it easy to do this and provides a mechanism to standardize activities across the organization. These activities include file access, logging, database connectivity and credentials management, to name just a few.

What works well is to start small and design the libraries to be easily extended and redeployed. This is where automation and testing are important. The architect should continually review the functionality built by developers and identify what should be promoted into the library.

CONTINUOUS INTEGRATION / CONTINUOUS DEPLOYMENT (CI/CD)

Very closely related to software architecture is the process of continuous integration. This is the idea of taking every commit to the source code repository and sending it through a pipeline of checks and tests, before allowing the code to be made available in production. These checks include style guidelines, formatting, type checking (for scripting languages), and, most importantly, test cases. If the pipeline fails any of these steps the deployment is halted.

CI/CD in cloud platforms are typically serverless, which makes it simple to control costs. Almost all programming languages have a test framework that is easily plugged into these CI/CD pipelines. In terms of nimbleness, test cases are one of the most important tools for being able to quickly push changes through to production.

COST CONTROL

Organizations have many reasons to explore moving to the cloud. One common reason is cost. On the surface, the idea of pay-as-you-go seems like an easy decision in terms of cost compared to building and managing a data center. The concern is the fact that an organization pays for services by the hour (or GB) but be warned: costs can spiral out of control without constant monitoring. If an organization uses scripting to deploy resources, it's simple to build scripts to ensure compliance with expectations.

Serverless architectures (containers, lambda) are another way to perform regular activities without having to deploy compute instances. The organization pays only for compute time and avoids costs related to idle time.

SECURITY

The financial services industry was a late adopter in the movement to the cloud, largely due to security concerns. As a result of advances to security in the cloud, issues related to multi-tenancy and public access to resources have been greatly mitigated. As with any tool, it is possible to configure a system to bypass default configurations meant to protect data. It's important to be aware of how data and resources could be made publicly available. Organizations should continually review what security is in place and take advantage of any security surveys that the cloud provider offers. Additionally, organizations should seriously consider the use of multi-factor authentication (MFA), single-sign-on (SSO) and the principle of least access.

CONCLUSION

While starting from scratch makes it easier to implement these best practices, it's possible to take an existing deployment and, piece by piece, migrate to a more nimble architecture. The flexibility and scalability of the cloud allows organizations to start small and thoroughly test along the way, at little cost.

Cloud service offerings continue to grow quickly. It's important to stay current regarding products and service enhancements which can often simplify workflows, as well as continuously monitoring the deployment to avoid unexpected costs or inefficiencies.

Nimbleness is a key advantage to investment firms because it allows quick and reliable deployment of new ideas. Architecting the environment using the practical ideas outlined above will make a team more efficient and confident as changes are promoted to production.

Mauricio Bustos, CFA—Partner, Head of Data and Technology, Radiant Global Investors