Kubernetes as an enterprise multi-cloud enabler

Building out your environment to create a full lifecycle delivery platform

FREEFORM DYNAMICS in association with **vmware**®

Why this discussion, and why now?

This paper was conceived in response to a lot of chatter we were picking up among enterprise application delivery teams around an emerging set of Kubernetes-related challenges.

By way of background, we noted a year or two ago that conversations taking place at specialist industry conferences and within expert communities were often not representative of the broader developer and IT pro universe. Outside of the advanced practitioner circles that tend to drive the overall narrative in this space, Kubernetes adoption is much less further along. This means the issues are different, with less focus on the absolute latest ideas and best practices, and more on the fundamentals of how to manage and scale Kubernetes beyond your first one or two clusters.

That said, mainstream adoption is now happening at pace. A common story we hear, for example, is that early initiatives have proven the value of Kubernetes, so more projects are now being fired up to exploit it. At the same time, activity is edging ever more into micro-service architecture territory, further underlining the role of advanced container orchestration.

While all this is very positive, the biggest challenge we are hearing about now is how to keep control as the size and number of Kubernetes clusters continues to grow. A contributing factor here is the typical adoption pattern of individual teams recognizing the value of Kubernetes, then going through their own separate learning curves as they each progress in their own way. Before long, you discover that each Kubernetes cluster is configured and managed differently, often using home-grown tools and integrations developed by each team based on their local needs, skills and experience.

The resulting diversity is fine to begin with, and individual project teams may be happy with what they are doing. At an overall organization level, however, this 'unique silo' approach translates to significant duplication of effort, limited reuse and portability of assets and expertise, and an inability to exploit cross-team synergies. Put simply, many mainstream organizations will see costs and risks escalate while frustrations increase and ROI falls if they don't take steps to standardize and harmonize sooner rather than later.

If any of this sounds familiar, then this paper is for you, but we aren't going to focus purely on Kubernetes. This is because there's a broader discussion going on about multi-cloud which provides important context. In practical terms, our firm view is that it makes sense to define requirements and invest at that level, embracing the Kubernetes-related issues mentioned above along the way.

Read on to see what we mean.

Kubernetes in context

The double-edged sword of modern software delivery

If you're involved in delivering software today, it's very likely you are working to strengthen your capabilities in key areas such as agility, repeatability, scalability and security. That typically means adopting cloud platforms, public cloud services, cloud-native architecture, DevOps, micro-services and containerization.

All of these have something valuable to offer, and when used together, they can really make a difference in both the development cycle and operational phase of any application. As many have discovered, however, if these modern methods and delivery models are not used in a coordinated and well-managed way, chaos can quickly ensue. Many of the benefits are then undermined by elevated costs, risks and distractions, and an inability to exploit synergies between projects and teams. And the longer chaos reigns, or even just a lack of proactive coordination is allowed to persist - the harder it becomes to address it down the line.

But how do you gain and maintain control without curtailing the freedom and flexibility needed by delivery teams to innovate and make the technology choices necessary to deal with unique application requirements?

Enter the Kubernetes-enabled multi-cloud approach

Whether 'multi-cloud' is a philosophy, approach or strategy is a debate we won't get into here. Much more important is the premise upon which it was founded and the principle it encapsulates. These are that no one cloud will ever meet all of your requirements, so you need the ability to mix and match clouds in a way that maintains control and efficiency, and promotes synergies across your organization.

To act on this, you need a few things in place, including a set of policies and practices, plus the finance and governance processes to go with them. You'll also need a range of technologies to provide the automation, management, monitoring and reporting capabilities needed for efficient and effective implementation.

It doesn't take most organizations long to identify Kubernetes as a key enabler here. With its ability to orchestrate the deployment and running of containerized applications at scale, it's an important linchpin of most multi-cloud environments.

Necessary but insufficient

While it's relatively easy to spin up a Kubernetes cluster in the cloud and start gaining experience, deploying and managing Kubernetes in a high-scale business-critical context is a different matter, especially as you accumulate more clusters. This generally mandates an additional set of tools to help deploy, secure and operate the Kubernetes environment itself, along with other components to deal with the delivery cycle automation, orchestration and management needs not covered by Kubernetes out-of-the-box. This is where modern multi-cloud delivery platforms come in.

Introducing multi-cloud delivery platforms

The basic idea

The role of a multi-cloud delivery platform is to help standardize and harmonize activity across the entire native-cloud delivery pipeline, as well as between pipelines in your broader project or service portfolio. An important principle here is catering for the needs of everyone involved in delivering a modern business application. That includes developers, operations staff, release managers, SREs (site reliability engineers), security and compliance teams, and yes, your business-side stakeholders.

A well-designed multi-cloud platform will help to address many of the organization-level challenges we have been discussing, but it's about much more than that. It will also make everyone's life easier and enhance both personal and team productivity. It does this by allowing each team member to focus on the activities that are important to their role, giving them quick and easy access to what they need when they need it. Meanwhile, non-core activities are either automated or routed to someone more appropriate, getting rid of those universally hated manual hand-off mechanisms.

In order to pull this off, a multi-cloud delivery platform must embrace or interoperate with the existing systems and toolchains used by delivery teams. This includes the likes of Git, Jenkins, Puppet, Chef, as well as the IDEs and other tools that developers and operations staff love and trust - not least Kubernetes.

Meeting individual needs

In practical terms, it's important to recognize that each of the roles or stakeholders we've mentioned has differing expectations and use cases. For example, developers need facilities that make it easy to build, deploy and manage Kubernetes-based apps and clusters by offering self-service access to tools and resources, while operations teams need capabilities that make it easy to set appropriate policies then optimize, monitor and manage Kubernetes clusters. This includes tools to troubleshoot when alerts are raised or cluster metrics are heading in the wrong direction.

Similarly, security teams need a platform that simplifies the task of securing Kubernetes clusters, for example via role-based access control, encryption, and vulnerability scanning. Compliance teams then require features that enable them to meet legal and regulatory needs, such as audit logging and the ability to restrict access to certain resources.
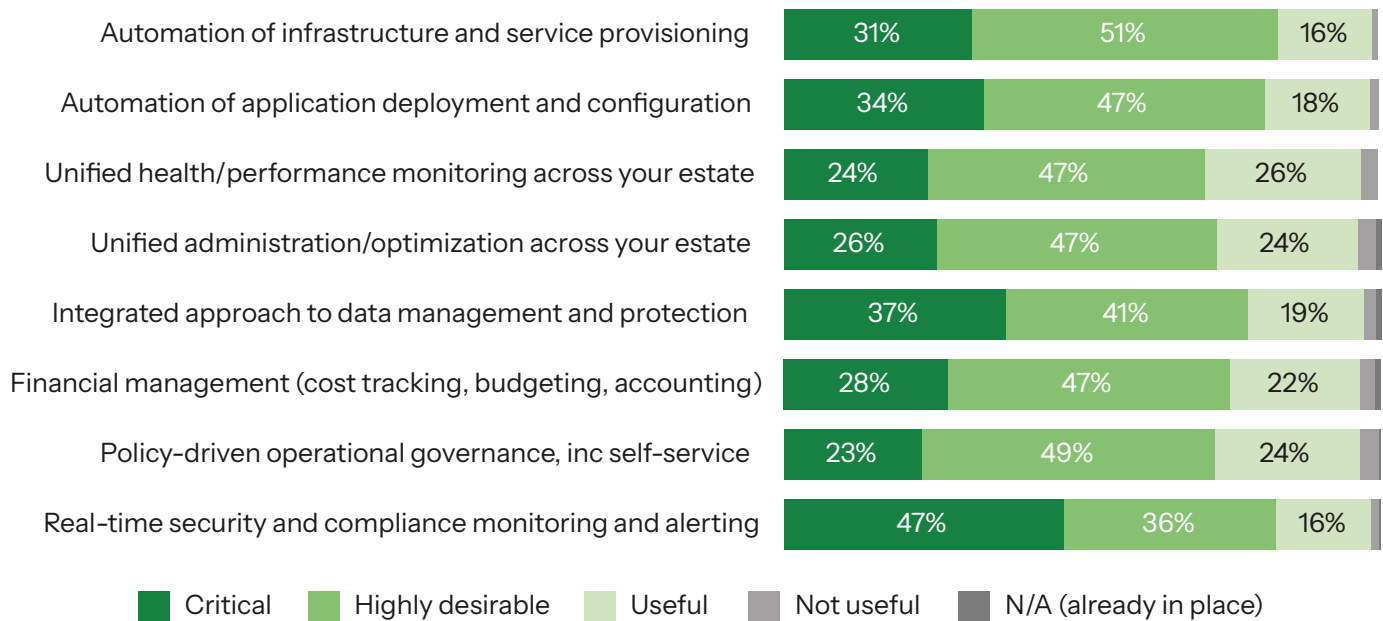
Enabling continuous visibility

A key thread running through almost all of the above is monitoring and visibility. Your platform must be designed and built with this in mind, whether it's to support continuous optimization, meet compliance auditing needs, manage service quality or discuss costs and key metrics confidently with business-side stakeholders.

If all of this is starting to sound a bit idealistic, don't worry, we are about to get into some of the real-world practicalities.

Anatomy of a multi-cloud delivery platform

In a recent survey of 774 senior IT professionals, we introduced respondents to the concept of multi-cloud delivery platforms then quizzed them on how important it was for them to include a range of specific capabilities (Figure 1).

Figure 1: Perceived importance of key platform capabilities

Capability	Critical	Highly desirable	Useful
Automation of infrastructure and service provisioning	31%	51%	16%
Automation of application deployment and configuration	34%	47%	18%
Unified health/performance monitoring across your estate	24%	47%	26%
Unified administration/optimization across your estate	26%	47%	24%
Integrated approach to data management and protection	37%	41%	19%
Financial management (cost tracking, budgeting, accounting)	28%	47%	22%
Policy-driven operational governance, inc self-service	23%	49%	24%
Real-time security and compliance monitoring and alerting	47%	36%	16%

Legend: ■ Critical ■ Highly desirable ■ Useful ■ Not useful ■ N/A (already in place)

Some of the functions we see listed here are self-explanatory, such as automation of infrastructure and service provisioning, which includes, for example, standing up Kubernetes clusters. Automation of application deployment and configuration, either directly or via integration with some of the previously mentioned third party tools is another fairly obvious capability that's required.

With regard to monitoring, administration and optimization, the word 'unified' and the phrase 'across your estate' are particularly important. It's capabilities like these that underpin your ability to break down silos or avoid them springing up, which in turn is critical to harmonize, standardize and achieve those sought-after efficiency, risk management and synergy related benefits. The integrated approach to data management goes hand-in-hand with this.

Add real-time security and compliance monitoring and you can already see how the scope of a multi-cloud platform goes far beyond standard Kubernetes functionality, at the same time allowing you to starting treating your various cloud instances and Kubernetes clusters as a unified whole. As well as enabling estate-level efficiency and harmonization, this also makes managing and optimizing distributed applications (with components spanning multiple clouds or clusters) much easier, and the benefits from a workload migration perspective are also very clear.

Lastly, with so many organizations already experiencing runaway costs as a result of uncoordinated cloud usage, you can see the logic of incorporating financial management and policy-driven governance into your multi-cloud delivery platform.
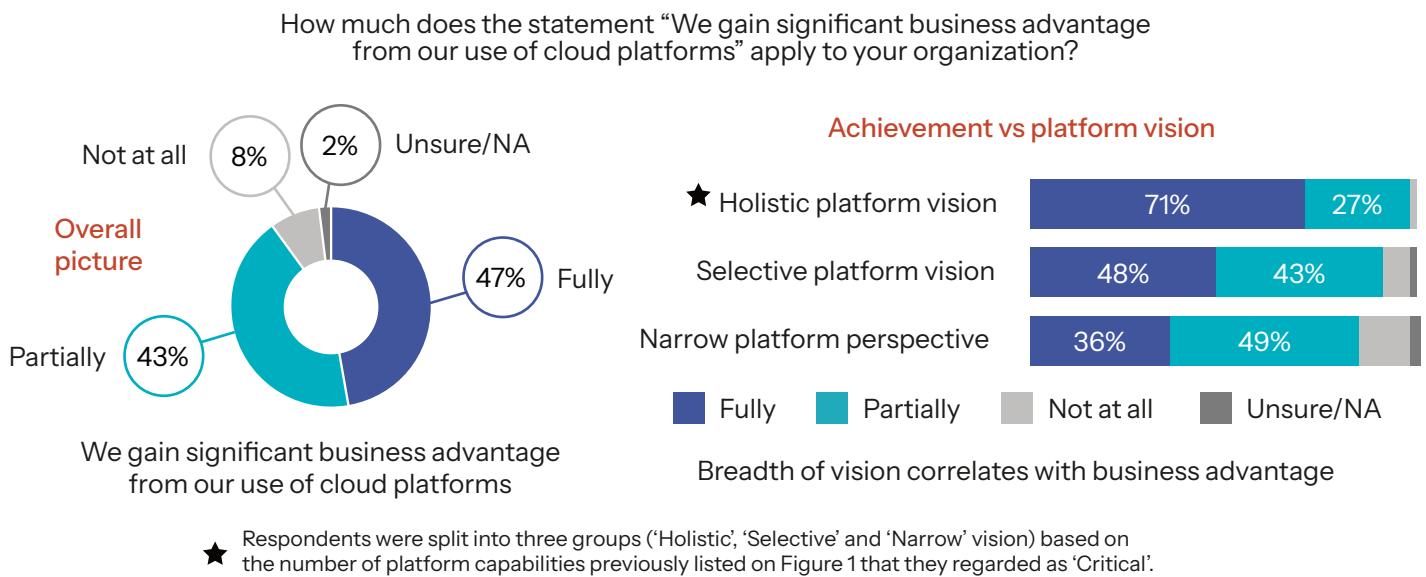
Platform implementation options

Build vs buy

When it comes to implementing your multi-cloud delivery platform, one option is to source the components you need individually then assemble them into a coherent whole. A few years ago, this was really the only option, and it was the basis upon which the discipline we know today as 'platform engineering' was originally founded.

The 'DIY' approach provides you with complete freedom to include whatever you want in your platform and define the way it works. As with any bespoke solution, however, you need to factor in a significant amount of cost and effort to deal with the necessary integration, testing, maintenance and support activities.

The other potential downside is that the platform ends up skewed towards capabilities that reflect immediate priorities and/or the things with which the implementation team is most comfortable. And this matters because the research demonstrates that breadth of vision when it comes to platform scope impacts the benefits you realize over the longer term (Figure 2).

Figure 2: Realization of platform value

How much does the statement "We gain significant business advantage from our use of cloud platforms" apply to your organization?

Overall picture

Not at all 8% Unsure/NA 2% Fully 47% Partially 43%

We gain significant business advantage from our use of cloud platforms

Achievement vs platform vision

★ Holistic platform vision — 71% | 27%
Selective platform vision — 48% | 43%
Narrow platform perspective — 36% | 49%

Fully Partially Not at all Unsure/NA

Breadth of vision correlates with business advantage

★ Respondents were split into three groups ('Holistic', 'Selective' and 'Narrow' vision) based on the number of platform capabilities previously listed on Figure 1 that they regarded as 'Critical'.

With the above points in mind, it's not surprising that we've seen the emergence of commercial platforms in this space that deliver pretty much everything you are likely to need, pre-integrated to operate out-of-the-box as a fully functional whole. Of course you will need to pay a license or subscription fee if you go down the commercial route, but the costs here are countered by the benefits of faster implementation and a reduced ongoing level of risk and distraction.

Another advantage is that commercial platforms are more likely to be scoped with the future in mind, so you're starting with a balanced set of capabilities. Make sure, however, that the platform allows components to be swapped out and replaced with alternatives more suited to your needs or existing point solutions you want to keep.

Scoping your platform
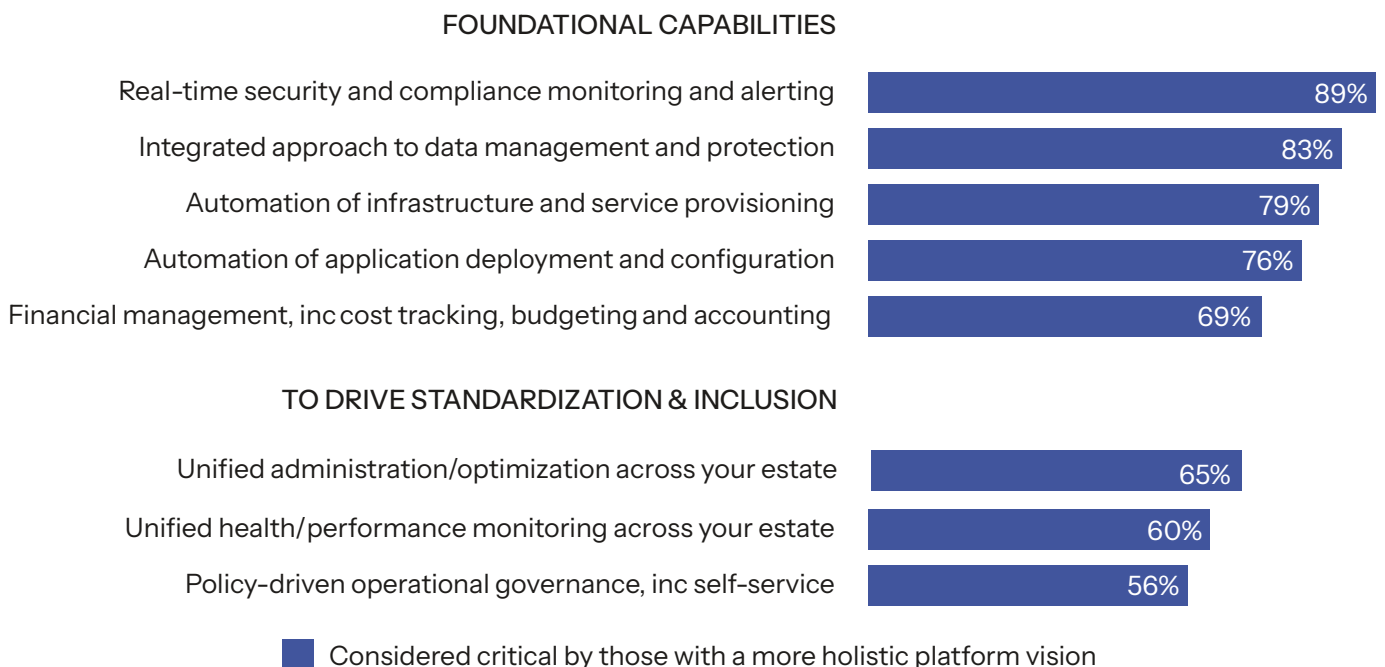
Prioritization and phasing

It is clear that the benefits of a platform approach – of consistency and standardization across teams, of automation and self-service, and of including all stakeholders, not just developers and operations staff – are real. In addition, the likelihood of gaining significant business advantage appears to grow as you take a broader and more holistic approach to platform adoption.

At the same time, it often makes sense to take a step-wise approach to implementation, of starting with a core set of functions then broadening out from there. Even with a commercial platform, if you've chosen wisely, you should be able to notionally 'switch on' individual components or functions only as you need them. Mechanisms will vary here, but however it's done, it means you phase your adoption process, incorporating existing tools from other sources where relevant as you go.

Learning from those with experience

When it comes to prioritization, you can get useful tips from our holistic visionaries. Looking specifically at that group within our survey, we found clear priorities in terms of how many considered each of our eight key capabilities to be critically important.

Figure 3: Prioritization by those with a more holistic platform vision

FOUNDATIONAL CAPABILITIES

Capability	
Real-time security and compliance monitoring and alerting	89%
Integrated approach to data management and protection	83%
Automation of infrastructure and service provisioning	79%
Automation of application deployment and configuration	76%
Financial management, inc cost tracking, budgeting and accounting	69%

TO DRIVE STANDARDIZATION & INCLUSION

Capability	
Unified administration/optimization across your estate	65%
Unified health/performance monitoring across your estate	60%
Policy-driven operational governance, inc self-service	56%

■ Considered critical by those with a more holistic platform vision

Setting your own agenda

Of course your priorities might be different from the ones listed above, and we'll talk more about who should be involved in the prioritization and phasing process shortly. Suffice it to say for now that it's important to take input from all potential users of the platform as well as key stakeholders within the business.

Tips for a successful implementation

Treat the platform as a product

For the platform to bring benefits such as efficiency and ease of use right across the software lifecycle, you must treat and deliver it as an internal product. That means you build and maintain it using product management principles, such as user research, product launches, roadmaps and promotion, and a process of continuous improvement based on user feedback.

In line with this, it generally makes sense to appoint a product manager, or define a steering group or equivalent that performs the product management function. Either way, you then need a cross-functional platform team (full or part time, depending on the scale of your operation) instilled with a product management mindset.

Inclusivity and buy-in

Your multi-cloud delivery platform must address the expectations and differing needs of multiple internal customer groups, You must automate where possible and provide them with predictable and reliable self-service access to the tools, information and other resources they need – remembering that those resources will also include training and documentation.

Part of the platform development process is to spend time making sure the differing needs and wants of all potential platform users and stakeholders are well-understood and their expectations properly managed. Critical roles to cover include developers, operations staff, application support teams and security specialists, though there will undoubtedly be others depending on your organization structure.

This kind of proactive inclusion is important because the dictatorial ("you must use it") approach almost never works. Instead, users should actually want to use the platform, for example because it helps them do a better job, faster and more easily.

Continuous improvement

Achieving maximum platform adoption at the outset is just the start. To keep users on board it's best to think of the platform as a 'living and growing' entity that constantly evolves to meet new needs and embrace new ideas. The platform should always feel fresh, dynamic and relevant to prevent users drifting away over time.

With this in mind, it's critical to put mechanisms in place both to gather user feedback on an ongoing basis and to ensure that feedback is acted upon in a timely manner.

And we make no apologies for reinforcing the importance of inclusion. Even if you have a single product manager or tight team responsible for building and enhancing the platform and running it day to day, it's critical to have representatives of each interested group involved in the product management process.

Their willing participation is vital to the platform's initial and continued success, not to mention the organization as a whole seeing a return on its platform investment.

Final thoughts and recommendations

Start by establishing where you are at the moment

Optimizing and harmonizing your software delivery activities is almost certainly going to be a complex and many-layered process, involving not just technology change, but also organizational process and culture change. So it is essential to start by honestly assessing where you are now.

As with any other major change initiative, unless there's a clear understanding of your current situation, acknowledged by all involved, your plans and aspirations are much less likely to come to fruition.

Think and plan holistically

When it comes to the planning process itself, you will need to think holistically, considering physical platforms, people and policies, best practices, governance models, and so on. Example questions to ask along the way include: "What do we do, what do we have in place now, does it work, do we want to keep it?" That in turn means assessing the organization's current roles and skills, systems and software, functional conflicts and gaps, and of course areas of redundancy and inconsistency.

This is one of the areas where adopting a commercial platform can assist you in understanding the likely scope and functionality you will need, as well as enabling you to see possibilities that you may not have considered yet. In this it can help you avoid the kind of tunnel vision that arises if you focus too much on your current position and capability.

Don't forget the internal sell

It is essential to consider how you will describe the platform and 'sell' it internally. When doing this, focus on the benefits that the platform can provide to each team or stakeholder.

As mentioned earlier, your developers are likely to be interested in the platform's ability to help them build and deploy applications more quickly and easily, while your operations teams may be more interested in its ability to help them manage and monitor infrastructure effectively. Senior business and financial stakeholders, meanwhile, will be more concerned with service delivery, cost management and greater visibility.

And remember, don't just engage people around their priorities, but actively avoid talking about things they really don't care about, as this will just reduce the impact or your proposition to them. Instead try to find ways to expand their vision.

And lastly

Don't leave it too long to address the matters discussed in this paper. The sooner you act, the more you'll head off potential issues before they escalate, and the greater returns you'll see from your multi-cloud investments.

About

About Freeform Dynamics

Freeform Dynamics is an IT industry analyst firm. Through our research and insights, we help busy IT and business professionals get up to speed on the latest technology developments and make better-informed investment decisions.

For more information, visit www.freeformdynamics.com or follow us on Twitter @FreeformCentral

About VMware Tanzu

VMware empowers organizations to accelerate and optimize app delivery across public clouds. We offer modular, cloud native application platforms and multi-cloud management capabilities that secure app delivery across clouds, while automating platform operations and governance at scale. Our VMware Tanzu and VMware Aria solutions enable developers and operators to continuously improve cost, performance and security throughout the app delivery process. With these solutions customers can experience faster, and more secure paths to production, improved efficiency through automation of Kubernetes platform operations and more consistent governance through centralized cost optimization, actionable insights, and policy management for your public cloud infrastructure.

For more information, visit www.vmware.com

About this document

This document is based on an independent market perspective from Freeform Dynamics. It incorporates supporting data from a study conducted in 2022, during which 774 senior IT respondents were asked for their views on their evolving multi-cloud delivery requirements and the platform technologies relevant in this context. Survey respondents were from medium to large organizations spread across 10 countries and a cross-section of industries.