BEAMING UP TO THE CLOUD
Mission briefings for open source IT leaders on hybrid cloud, auto-scaling, and microservices
To boldly go where no one has gone before, IT leaders beckoned by cloud computing need a new kind of starship. Like a superior vessel going into the final frontier, a first-rate cloud platform helps you navigate new and uncharted cloud territory with advanced tools to quickly build, deploy, and manage apps, databases, and services for any open source–powered enterprise.

Science fiction heroes command starships they can count on to get through the nebula. IT leaders need the same kind of figurative spacecraft—a fast-moving, innovative, enterprise-class platform that lets you and your team harness the power of the cloud. With cloud computing, development shops can act with greater agility and evolve their apps in previously unavailable and unfeasible ways, gaining the most from the cloud by using hybrid technologies to build on existing innovations and assets.

Even the most impressive fictional starships were refitted to meet the march of progress. So, too, today’s industry cloud players are bringing their offerings to the next level at an astonishing clip. IT leaders are seizing this evolution by using industry additions to upgrade their development environments in order to raise their competitive profile via open technologies, platform-agnostic interfaces, and architectural innovations.

This eBook will set your course for exploring new capabilities built in the cloud, with concrete ways to address pain points and architect applications to leverage the cloud’s flexibility. For IT leaders interested in empowering their teams to develop in scalable and sustainable ways, deploying a platform that offers faster and simpler solutions isn’t only important, it’s mission critical.
YOUR JOURNEY BEGINS: FINDING SUCCESS IN THE CLOUD.

Meet today’s demand on IT to do more, faster, by building out cloud muscle.
In a fictitious, futuristic universe where starships engage in deep-space exploratory, diplomatic, and military missions, commanding officers value a fully operative and reliable vessel.

In the real world, IT leaders from landscapes far and wide are being driven by their own missions to support critical operations of their enterprise, power the newest bright idea of their developers, spin up development and test environments at warp speed (and on an as-needed basis), and meet growing or fluctuating demands on their team and their computing bandwidth. With increasing demand on IT to do more, faster, it’s truly a new frontier. And if there’s one thing you need to take seriously in today’s fast-paced, mobile-first, data-driven reality, it’s this: your success—yours and that of the enterprise—is highly influenced by your choice of the right cloud and your ability to use that cloud platform to catalyze developer potential to get applications quickly to market.
The power of cloud convenience

More devices, more apps, and more data make IT more challenging than ever. But it’s never been a more exciting time for developers. Mature computing platforms built upon the wisdom of communities that seeded them are allowing developers to focus on building innovative solutions, thanks to scalable, clonable technology. The days of building totally new architectures to enable applications are long gone, and the cloud is playing a big part in freeing up developers for more interesting and meaningful design questions.

Along with cloud-based development platforms come efficiencies you just don’t see with traditional computing, specifically the ability to:

- **Self-provision** development and testing environments, so you can get moving on application builds without having to coordinate with IT to provision instances.
- **Align** teams of developers, architects, and designers on app development.
- **Expedite** application production and reduce time-to-market by cloning development and testing environments, configuring and adjusting scaling rules throughout the build process.

When done right, cloud computing lets developers focus on shipping apps without worrying about keeping the lights on. Developers and IT leaders know all too well the negative organizational effect of reinventing the wheel when manually configuring web servers, setting up network-attached storage (NAS) and array servers, or implementing network routing solutions. The right modern cloud platform creates a level of abstraction around layers of service provisioning and computer management, and removes the manual oversight required to keep these running, enabling developers to spend their energy and effort writing and deploying applications that address higher-order customer needs.

Companies benefit from paying as they go, minimizing capital expenditure to run Internet-connected applications. With businesses spending close to $118 billion on cloud technologies in 2015 according to the International Data Corporation (IDC), it’s evident that enterprises are answering the call to do more, faster (and more cheaply)—delivering new projects and products, managing more devices easily, crunching more data quickly, and spinning up dev and test environments with unparalleled speed—all with cloud computing.

Most telling, companies that see value in the cloud are not only embracing the most basic of cloud capabilities (virtual machines [VMs] on demand), but also migrating toward more modern data storage and processing, scale-out architectures, application automation, and more. And they aren’t afraid to re-architect and re-platform existing applications—a cloud-native approach.
Cloud as a critical tool for development

The rapid spread of cloud-computing technology is empowering developers every day. Thanks to the cloud, developers can focus on coding and not have to worry about installing, customizing, and upgrading their tools. Because they aren’t getting ensnared in hardware and networking concerns, they can concentrate their efforts on building incredible programs to benefit users.

There are clear innovation benefits to developing software in the cloud:

- **À-la-carte cloud stack.** As many or as few features and add-ons as you need (run a LAMP stack, set up a Node.js website, or set up MongoDB all in one account).
- **Real-time, global collaboration.** Developers who do not share an office (or work across independent teams in an unfederated environment) now have the ability to collaborate easily on such distributed teams, provision accounts, track changes, and work on multiple branches of a code base in a shared repository.
- **Flexibility.** Developers can quickly increase their demands on the infrastructure if they need to test some code or run much larger data sets than originally expected. The cloud is elastic and can accommodate almost unlimited demand.

- **Effortless scalability.** For businesses with growing or fluctuating bandwidth demands, it’s easy to scale up cloud capacity, drawing on the cloud’s remote servers. Likewise, if you need to scale down again, the flexibility is baked into the cloud service.

The cloud is the soil in which transformation blooms. Consider, for example, the everyday workflows of developers. The cloud doesn’t merely lift and shift repetitive actions. Cloud development rationalizes efficiencies of scale inherent in its design; it brings new tools, techniques, and languages to bear in order to transform the way teams work, and allows your company to keep pace with the breakneck speed of development today. And, in a way, the advent of the cloud is the answer to its own challenge. No reverting to on-premises geo-redundancy, with manual commits to each site. Cloud computing is the challenge and the answer—with nary a change-management project in sight.

When cloud computing centralizes, automates, and distributes your development workflow, development becomes so much easier. “Agile in the cloud” is a game changer for how quickly enterprise can serve clients.

Two of the core tenets of agile build, Continuous Integration (CI) and Continuous Delivery (CD)
workflow, an everyday fact of life for developers, fully pay off on their scalable potential in the cloud. CI automatically triggers a build whenever a developer checks in code to the source repository. CD takes this one step further: after a build and automated unit tests are successful, you automatically deploy the application to an environment where you can do more in-depth testing.

Because you’re running everything in the cloud, you don’t have to buy or manage servers for your builds or your test environments. And you don’t have to wait for a server to be available, in order to do your testing. The cloud enables you to minimize the cost of maintaining a test environment because you pay for the environment resources as long as you’re using them. Your CD process can set up the test environment when you need it, and you can take down the environment when you’re done testing.

The cloud makes it possible for continuous software iterations and improvements to replace lengthy development, testing, and release cycles. Each step in the process is usually small, so iterations move through the development pipeline quickly. As a result, a continuous application development model accelerates the overall development process and improves efficiency.

Expect more from the cloud

The cloud provides not only a framework, but also a platform-agnostic software ecosystem to support developers. Software platforms that simplify essential programming tasks and provide consistent interfaces enhance every programmer’s productivity. Developers need to look for tools that power them to bring innovation to market faster, and provide the deep foundation needed for enterprise use.

As an open source enthusiast, a good question to get started with is “What exactly do I want to do with the cloud?” Consider these options: Do you want to test your web applications? Do you want to launch a Linux server virtual machine? Or Windows to test for that platform? Maybe you want to deploy a database. Or do predictive analytics or perhaps automate mobile back-end hosting for fluctuating traffic. Or are you looking to use the cloud to test out something new? And that might mean running MapReduce jobs on Hadoop or using R for your machine learning projects. It could also mean or running a Debian VM to communicate with IoT endpoints running Raspberry Pi or setting up Red Hat CloudForms to manage hybrid cloud containers—the kind of projects where you need more agility than
what is regularly delivered in your conventional environments.

The right cloud-based development platform will enable you to execute on any one of these scenarios, and one of the most exciting: instantly spinning up a test environment by simply using an automation script, running acceptance tests or more in-depth tests against it, and then automatically de-provisioning your low priority queues when you’re done, saving valuable hours of labor.

If you’re making the choice to adopt the cloud, choose a platform that leverages all the benefits that the cloud brings to bear (interoperability, openness, hybrid portability, automation) without compromising on capabilities that your team and stakeholders demand.

Azure is Microsoft’s open and flexible cloud platform that lets your team focus on code and use the platform’s building blocks to scale rapidly and in an agile way. You can architect complex enterprise-grade Linux-based solutions that run seamlessly across multiple cloud instances. Azure has invested to make open source a first-class citizen, which means you can build in whatever environment (Linux, OS X, or Windows Server), language (Python, PHP, Java, Node.js, .NET, C#, Visual Basic, C+++, and others), development tool (Git, Visual Studio, Eclipse, and others), data platform, or open standards you are already working in. You can build on virtually any technology and data source. You can leverage the wisdom of crowds by using any number of open source configuration management tools (like Chef or Puppet) to deploy your app across a global network of 22 Azure regions around the globe.

The Azure platform offers a complete tool set that supports interoperability, which powers portability. How? By allowing developers to take their existing applications to the cloud when needed without worrying about whether their technologies are compatible with the cloud environment.

When done right, cloud computing pays for its learning curve quickly because it helps organizations do more, faster. The cloud frees up hours of labor and datacenter resources. What’s extra in that solution set is a fundamental appreciation of how business uses technology, today, to gain efficiencies via systems they previously may not have been able to afford. Businesses are quickly recognizing this value (for DevOps, distributed teams, and other evolving work styles) and are quickly moving services to the cloud. But for larger, established companies (i.e., companies with significant investments in IT), going "all cloud" may not be feasible from day one. If your company exists with the need for both on-premises and cloud services, you’ll want to consider a hybrid cloud computing approach where you choose which workloads are good
candidates for the cloud, and which are better off remaining on site.

Look for a cloud platform that:
- **Provides flexibility** to build your application, your way, regardless of your use case, and to manage your infrastructure without compromising dependability and scalability. (Business impact: minimal process or talent adjustments needed)
- **Allows for easy interoperability with other platforms.** (Business impact: minimal retraining or hiring need; existing vendors can be maintained)
- **Powers data portability.** (Business impact: agility and speed of execution and archiving; freedom to make changes)
- **Embraces a hybrid cloud computing approach** ensuring that past enterprise IT investments are leveraged. (Business impact: guaranteed payoff on existing IT investment; clear divisions in chargeback systems maintained)

Until recently, IT leaders and developers have had three options when it came to the cloud: ignore it, block it, or go with it. The first two options are now off the table—the cloud ship has sailed. According to a new Evans Data survey, there are nearly 5 million developers using cloud as a development platform, a number projected to triple within the next 12 months. And it’s no surprise. “The era of cloud computing has enhanced the epoch of the developer,” notes Megan Swanson in *WIRED*; this technology “lets [programmers] focus on making applications that create new functionality, new business opportunities, and even new industries.” The value of the cloud is evident, the technology is solid, and the opportunity is clear. Are you in? If so, engage!
Chapter Two
BUILDING A BETTER STARSHIP: OPEN AND FLEXIBLE WIN THE DAY.

Leverage open source software in the cloud to catalyze developer potential and satisfy enterprise needs.
A universe of open source is a universe where sharing is the modus operandi. Some of the best tools in software development today all integrate open source solutions and recognize the cultures that have sprung up from the globally collaborative workflows that they enable.

Open source does not have to mean patched-together fixes with limited support. In many cases, open source is the engine behind most new cloud platforms. For your missions, a formidable starship matters. And so does the engine behind it. What is the engine that will power your ship today? It’s this: innovation coming out of contribution and cooperation in the cloud.

Make-believe interstellar explorers of the future boldly go where no one has gone before, engaging with citizens of new civilizations to coexist and learn. Like outer-space explorers of the imagination, today’s developers aim to expand the boundaries of their knowledge through cooperation and contribution.

Developers place an extremely high value on sharing—you share your successes, you share what you’ve learned from your failures, you share your metrics, you share your code. You do it in such a way that others can meaningfully contribute.
Chapter Two | BUILDING A BETTER STARSHIP: OPEN AND FLEXIBLE WIN THE DAY.

Straight from the source

IT is not changing itself. The world’s 18.5 million developers are using IT to change the world, one small innovation at a time. In this Internet age, software is disrupting every industry. The agents of this change are business applications that are fundamentally transforming how organizations operate, interact with customers, and go to market.

IT leaders have the dual task of setting up the organization for success, while ensuring that this strategy considers evolving technology’s promise in the next few years. What is the responsible, thoughtful, and innovative stance to take, in a playing field that seems to shift regularly?

The constant with the cloud, and in your open source shop, is collaboration and scale. If your developers are going to build applications that leverage the cloud to meet enterprise needs, then you need a platform that supports collaboration. Open source, used in conjunction with the cloud, catalyzes developer potential. Imagine a platform that centralizes hubs of code for operating systems and essential utilities, and contributes to debugging and trouble-shooting. This platform adds horsepower to your person power; it elevates developers’ efforts on higher order problem solving, yielding powerful programs that rapidly prove out customer value.

Open source is the developer secret sauce that cuts out the learning curve when transitioning to another job, or when the workplace decides to use a different language or technology. Open source allows you to “bring your own cloud,” making it much more possible to migrate to other hosting services down the road should the need present itself.

Many of the exciting things happening in cloud development today are open source. The latest critically lauded technologies to leverage cloud development paradigms include database and cache fulcrums like Cassandra, Elasticsearch, Memcached, and Redis. But one of the most fascinating pieces of technology to emerge from the recent open source wave is Docker, which has received a tremendous amount of attention, thanks to its ability to package and port apps neatly between platforms; this has been a boon for everyone, including tech shops with limited on-premises development environments. Big companies are finding manifold uses for Docker in production, and developers claim that it frees them to do more interesting, challenging work. As the cloud evolves, developers are looking to containers as a way to build and deploy quickly and efficiently these “born in the cloud” applications because they allow for flexibility and scalability across platforms. And many believe Docker is going to shape the industry for years to come.

Open source as engine

So what’s the best approach if you want to take advantage of open source software in your cloud?
While you set your course for building applications that leverage the cloud to meet enterprise needs, your chosen platform should guide you to the right capabilities, facilitating faster app development and testing, and providing the flexibility to migrate between the cloud and on-premises. As we’ve seen, it should facilitate using almost any programming language, development framework, or tool, meeting you where you and your team are—that is, working with the tools (e.g., Docker, Java, Red Hat) that you’re already using. And it should place a high value on sharing and collaborative workflows.

Open source is not an “if”; it’s a “when.” With open source a reality for leading cloud vendors, you can expect your platform to run on virtually any server platform. Chances are your cloud provider is already using open source to operate thousands of instances. Azure certainly is.

The Microsoft cloud supports a wide range of industry-leading operating systems, languages, tools, and frameworks—from Red Hat to Ubuntu, Windows and openSUSE, MariaDB and SQL Server, C# to Java. It puts the best of all ecosystems at your fingertips so that you can build great applications and services that work with many devices.

Most of the exciting things happening in the cloud today are open source.

Today, one out of four Azure virtual machines are Linux, and more than 60 percent of Marketplace images are Linux based. Azure also supports and cleanly integrates the leading open source languages, development environments, and infrastructures (e.g., PHP, Java, Node.js, Python, and Ruby). Moreover, Microsoft supports the Hadoop ecosystem and offers Azure HDInsight, a 100 percent Apache Hadoop–based cloud service that can deploy to Windows and Linux. And with an open RESTful API for every component, SDKs for multiple languages, and a wide array of tools and automation choices, Microsoft is lighting up new scenarios, like the Internet of Things ingestion networks, big data, machine learning, and Docker-powered next-generation architectures. Open source platforms as diverse as geographers’ CAD mapping tools can be run on the Azure platform.

A few years ago, the Danish Government’s environmental protection office, was able to take charge of its costly on-premises data stores by moving to Azure cloud-based hosting, thereby shrinking costs while removing the hassle of manual, multisite backups and compliance auditing. Like many public ministries, the office had to do a lot with little. The small team of only nine employees was charged with maintaining 14 different databases containing data intended for
a panoply of Java-based open source geographic interfaces, in addition to other third-party visualization and drafting software. With the move to Azure, they could manage servers in the field on their laptops, a handy task considering that so few staff directed the configuration and monitoring of data stores representing an entire country’s reserves of natural resources.

**Open source no longer means patched-together fixes without any support.**

**Trumpeting open source initiatives**

IT shops are delving into the open source market to meet enterprise needs. And that market is responding. To attract new developers to the open source scene, more vendors are releasing their code. Even Microsoft opened its server-side .NET stack and broadened it to Linux and Mac OS, among other projects.

At the heart of Microsoft’s cloud is Azure, an open and flexible platform that can keep pace with the rapid speed of open source technology development. Unlike other vendors that are taking the standards they originally wrote or built for themselves and now are retrofitting them to be open, Azure was built from the get-go to be open, or rather, with more open standards.

Open source is, in fact, key to Microsoft’s approach to cloud innovation. To illustrate: Microsoft is delivering new container capabilities with Docker integration, and is doing so in the open with future services like Azure Container Services. Microsoft is constantly looking for ways to improve developer and user experiences with SDKs for open source languages and an open API. Plus, Microsoft is committed to sharing its cloud insights with you and for your datacenters, thanks to Linux and open source support in Azure Resource Manager and, in the future, Azure Stack.

Azure supports the open source technologies developers already rely on by:

- **Including development and testing tools in its integration road map.** Azure supports tools like Vagrant, Jenkins, and Hudson, and integrates with multiple other technologies seamlessly via open APIs, open source SDKs, and command line tools that work on Linux and OS X.

- **Planning open source support into the product.** A number of Microsoft-managed services already build upon popular open source tools: Azure Container Services for Docker and Mesos; Redis Cache for distributed in-memory cache service; and HDInsight on Linux for big data projects. The advantage to working with these products is that none of them requires hands-on operation from developers; instead Microsoft manages each of these for you, and they are built with open source technologies.
• **Providing limited Linux support.** And thereby allowing developers to quickly remove roadblocks to using the cloud at any stage of development, from app building, testing, deploying, or in production.

• **Prioritizing hybrid and openness to ensure that workloads won’t be stuck.** This allows developers to move applications between on-premises and the cloud, knowing that their fully virtualized instances can be moved back to fully on-premises if the company so requires.

With a strong commitment to enabling developers working in open source to leverage the platforms they know and love, Microsoft, in partnership with the Linux Foundation, recently launched a Linux on Azure certification program for IT professionals looking to master Linux administration and Microsoft Azure implementation skills. A Microsoft-issued certification that includes the Linux Foundation Certified System Administrator exam allows IT leaders and developers to stand apart from their peers while being validated technically and professionally on the most interesting technologies of our time.

In our software-defined universe, if the IT machine is innovation, then the ultimate engine is open source.

Whether make-believe navigating an interstellar republic of planetary sovereignties or real-life cloud mapping, it’s your starship at the starting line. When your starship brings open and flexible forward, giving you the freedom to leverage the galaxy and cloud at any stage, then it’s a genuinely transformative vehicle that can herald the way for innovation.

While Microsoft engages with the open source ecosystem and onboards open source solutions in Azure, you and organization will be empowered by the choice of platforms and tools, the real possibility of “bringing your own cloud,” open source–friendly support, and smart management options.

Know where your own cloud plans will take you and where your open source cloud software providers are going before you make your choice.
Chapter Three

LIFT OFF: HYBRID CLOUD SOLUTIONS OFFER THE BEST OF BOTH WORLDS.

Be ready with new management tools to tackle the next big thing—hybrid cloud—head on.
Multimission operations require extensive coordination among different departments. With operational authority for your starship in your hands, you’re charged with critical decision making for important coordinated efforts for your enterprise, such as finding the best way to bridge the corporate datacenter and the cloud. You’re not alone. IT leaders are tasked with protecting their on-premises infrastructure as one does a major business investment and proprietary asset. Yet, you also understand that the benefits of the cloud are too great to ignore. The question eventually arises: what is the easiest and fastest way to get our apps into a public cloud environment while maintaining tight integration with our on-premises system architecture?
Chapter Three  

LIFT OFF: HYBRID CLOUD SOLUTIONS OFFER THE BEST OF BOTH WORLDS.

Gaining more flexible, extensible infrastructure

We all know that the case for cloud computing—built on the sound pillars of ease of use, flexibility, and cost efficiency—isn’t lost on today’s enterprises. But even if they’ve made the decision to adopt the cloud, organizations are faced with another choice: when to start leveraging the public cloud, when not to, and when to move services and workloads out of the datacenter into public clouds. The good news is the answer doesn’t have to be cloud exclusive. A hybrid cloud approach allows your team to merge the traditional datacenter presence that you know with the benefits of a public cloud. Taking the hybrid route offers the flexibility of implementing new capabilities in the cloud while still taking advantage of organizational wealth built up in the existing infrastructure.

Businesses recognize that they need both on-premises systems and cloud services. So consider joining, if you haven’t already, the growing ranks of IT leaders addressing these two needs mutually with hybrid cloud deployment—a sure-fire way to move to the cloud while getting more value out of the on-premises investment.

Getting your migration off the ground

A move to the hybrid cloud may not be as radical as moving entirely out of the corporate datacenter, but it can still mean significant changes for your team and users. There’s a lot of work to be done, so where do you start? At the heart of the challenge of merging private and public clouds is deciding which workloads to run in the private cloud, which to move to a public cloud, and how that hybrid arrangement will affect your business.

When determining your options, you’ll want to take into account the size of your workload in addition to compliance and security considerations. The architecture of services associated with a certain application may make it poorly suited for use in the cloud; it might be best to include those workloads in your on-premises architecture for performance or financial reasons. Similarly, systems that require strict security parameters, specialized configurations, or dedicated hardware might not thrive in the cloud. Many businesses face stringent government and industry regulations that strictly limit the physical location of certain data types. Because the very premise of the public cloud is to run any workload anywhere in the world, there are bound to be conflicts between public cloud services and specialized business regulations. Consequently, companies that are bound by certain regulations will often need to delimit sensitive applications from the public cloud. Consider on-premises architecture, then, for fast, local access to large files; control over sensitive data; and compliance with certain regulations and audit agreements.
Cloud technologies are also your best bet for delivering secured mobile apps worldwide. The cloud is the ideal environment for efficiently managing end-user access and identity from virtually anywhere. With cloud-based tools, you can provide access to cloud apps in the field with single sign-on; manage access policies, allowing secure delivery of confidential documents on any device; set up security alerts triggered by suspicious behavior before damage is done; and save time and reduce help desk calls with user-directed account and password resets.

Consider the cloud, then, for data backup and storage; hyper-scalability on demand; and secured, distributed access to mobile applications.

Maximizing capabilities

A well-run hybrid cloud can rapidly deliver public and private resources, providing control and visibility to IT departments and the on-demand self-service that developers and application users expect. Through a more flexible hybrid cloud infrastructure, enterprises can accelerate time-to-market, meet ramped-up demand, and cut costs because a hybrid approach maximizes three key computing capabilities, namely:

- **Load balancing.** An unexpected increase in user activity might cause poor application performance when a data center exhausts its computing capacity. But a hybrid cloud can move some workloads from the private cloud to the public cloud (including “cloud bursting”) and back again as demands
dictate. Thus, a business can ensure adequate computing capacity at all times while paying only for the additional capacity when it’s in use.

- **Scalability.** A certain amount of scalability is always present in the datacenter, but the features are often too expensive and time consuming to integrate and manage for a private cloud. Because scalability is better in the public cloud, a hybrid cloud arrangement allows a business to use the computing resources that a public cloud provides to react faster than it could with a private cloud or traditional datacenter. The pay-as-you-go model prevalent with most public cloud providers also means you will pay only for the resources you use. And it ensures that your provider does not limit your use of a payment tier, but that you get true “on-demand” scalability.

- **Automation.** This is a key capability in the cloud, easing tasks like provisioning and ongoing maintenance. To use automatic scaling features in a cloud environment and allow for cloud bursting, you need to enable automatic provisioning and configuration of applications on cloud servers. Many organizations acknowledge the need for automation, but they cannot find the time to implement it in their own datacenters. A hybrid cloud approach can add automation and monitoring tools to help businesses implement better processes, while simultaneously helping businesses justify the initial implementation work.

Hybrid cloud deployment is a sure-fire way to move to the cloud while getting more value out of the on-premises investment.

An optimal hybrid cloud solution will allow you to do this quickly and easily in a control center with preloaded middleware elements. By deploying a best-in-class hybrid cloud solution, you can help your enterprise gain the flexibility it needs to implement new capabilities while getting more value out of its on-premises investment. Disrupting the environment of a successful system that your company spent years building, by moving or re-architecting applications, might be unnecessarily time consuming and costly, when you can simply extend that system by adding new features in the cloud.

When you get support from a cloud platform that provides this kind of flexible hybrid infrastructure, building solutions for your business can be quick, adaptable, affordable, and painless. Here are the benefits laid out:

- **Speeds time-to-market.** Developing new applications in the cloud means trying new things quickly, with very low risk and cost. Your team can help the business speed time-to-market by developing new applications in
Chapter Three  
LIFT OFF: HYBRID CLOUD SOLUTIONS OFFER THE BEST OF BOTH WORLDS.

The right hybrid cloud provider will help through a range of existing and new automation features to extend traditional capacity, performance, availability, and security management process to your hybrid cloud services.

Deciding on a hybrid cloud provider is trickier than choosing a pure-play public or private cloud offering. When considering a private or public cloud, you compare features and functionality based on one environment. But with a hybrid cloud, you need to look at the solution provided in both your environment and what is external to your environment. Tread carefully—transparency between internal and external resources doesn’t require workloads or performance to be mirrored. The hybrid cloud should be viewed as an extension of your internal resources. However, security, SLAs, and management should come as close to mirroring your internal environment as possible.

IT leaders need to take heed: crossing between vendors can create support challenges. Although many hybrid cloud providers can support multiple hypervisors and workloads, you won’t always get the additional features offered for each platform or the best performance when you’re crossing vendor lines. That’s why it is important to find a dyed-in-the-wool hybrid cloud vendor that supports not only workload migration, but also automating the operations element of the dual computing scenario: someone still needs to keep the cloud; it actually can be faster deploying to a platform like Microsoft Azure than to your own internal test environment.

- **Increases scalability.** In addition to becoming more flexible, a company can handle fluctuating workloads more easily. The Microsoft Azure platform provides you with scale. It mitigates the unpredictability of launching a new app or feature that turns out to be just the thing everyone was waiting for and is now demanding. So if yesterday you had 10,000 devices connecting to your app, tomorrow you can handle 2 million.

- **Cuts deployment costs and simplifies management.** For a company moving to a hosted datacenter, it can take years and cost millions of dollars. Expanding from two datacenters to four with Microsoft Azure might take a company one afternoon and a credit card.

- **Improves customer satisfaction.** By delivering new services faster and more affordably through a hybrid cloud infrastructure, companies are making their customers happier. The faster your business can drive change and try new things by using a supportive cloud solutions provider, the quicker you can arrive at the right combination of services that makes the customer experience the best it can be.

**Implementing success**

Don’t underestimate the scope required to integrate cloud services into your existing IT processes.
the lights on. The challenge many companies face is realizing too late that their cloud vendor leaves them with the burden of operational maintenance, as if the shift from on-premises hadn’t happened. True hybrid support must bake automation and best-in-class architecture into the solution architecture; they must provide the benefit one looks for in such layers of abstraction.

Azure, once thought of as purely a Windows platform, includes not only support for porting multiple technologies, it also features full support for your Linux virtual machines or Docker containers on internal and external clouds. But that doesn’t mean you should start a patchwork exercise with all your vendors just yet. For example, using Microsoft Hyper-V and extending it to a hybrid cloud other than Microsoft Azure is supported; however, that doesn’t guarantee you’ll be trouble free, or that you and your team won’t engage in the vendor blame game. The only difference now is that part of the technical problem is no longer on site.

If you are already considering Azure or have a subscription and are looking to get started, don’t be misled into thinking that you are limited to Microsoft development tools. You’re free to extend your cloud investment with a mosaic of applications and services, made bespoke thanks to the Azure Marketplace catalog of plug-and-play middleware and integrators that allow you to pay off on your Internet as a Service configuration. Enable your architects to augment existing assets with the technology that makes the most sense, not the technology that aligns with your vendor’s tool set. Being limited by a single vendor need not be a concern, though there is a case to be made for creating platform and language consistencies for repeatable operations and batched instances. Besides support issues, there are ease of deployment incentives inherent in an integrated middleware layer from a single vendor ecosystem.

However, some shops may want to go with a single vendor enterprise product to take advantage of enterprise SLAs. An enterprise configuration (such as Microsoft Azure cloud’s full integration to Windows Server 2012 R2 running System Center), can be used to complement an OSS environment, facilitating even broader workload consistency when moving from internal to hybrid cloud environments. For example, Red Hat integration gives Azure customers combined dual-vendor support with coordinated issue escalation, combining the expertise of a widely trusted open source distributor and a market-leading cloud platform.

Most businesses live with the reality that they need both on-premises systems and cloud services. On-premises datacenters don’t provide scaling options and cost efficiencies to the degree that public clouds do, but enterprises also want the option of keeping some data behind
their firewalls. A hybrid model offers the best of both worlds. By deploying a hybrid cloud delivery model, you can leverage and maintain past IT investments. For IT teams, getting public cloud elements and your infrastructure to work together and grow over time can be challenging. But the continued evolution of cloud standards, practices, and tools should make hybrid cloud integration and growth easier.

Vendor excitement about the “hybrid cloud” promise is great news for open source shops. Nevertheless, excitement today doesn’t always translate into solutions. Microsoft offers a pure-play solution for hybrid cloud computing.
Chapter Four

KEEPING YOUR SHIP ALOFT: STAYING AUTOMATED AND ELASTIC.

Get automated the right way to adapt to fluctuating workloads.
Running a tight ship is made easier when computers can control the vast number of critical systems responsible for the necessary functions of your vessel. If you were a starship commanding officer even with the most robust crew, you would no doubt welcome technological advances that helped lighten the task load of engineers and team members.

While we evolve to a world of Infrastructure as Code, it becomes imperative that the technology powering complex, web-facing applications be automated for high availability, compliance, and “smart” threshold scaling. Manually monitoring performance and scaling a system to adapt to fluctuating workloads can be a labor-intensive process. And it may not be feasible to work this way at scale. This is where the value of threshold scaling comes in. Threshold scaling allows for adding and removing capacity in a cloud infrastructure according to usage demand/volume, without any human intervention.
Handling load increase and unpredictability

The ability to scale on demand is one of the biggest advantages of cloud computing. Today, a website’s ability to respond to massive surges in visitor traffic has rapidly become a top concern. And although it’s true that scaling vertically is the easy solution—simply boost your RAM, add a few cores, and instant gratification is yours—the problem with this approach is that it isn’t future proof. This solution is only as good as the instance’s size and room for growth. The better alternative is to invest some consideration and time into a horizontally scaled software architecture.

Of the two types of scalability, vertical scalability (i.e., scaling up) is the traditional and easiest way to expand—by upgrading the hardware you already own (that is, buying a more robust and expensive server) or redeploying the solution using alternative hardware that has greater capacity and performance. The downside is that eventually you will get to a point where either the cost becomes prohibitive, or you max out available hardware capacity. Also, scaling up can be a disruptive process that requires making the system temporarily unavailable while it’s being redeployed. At times, it’s possible to keep the original system running while you provision the new hardware and bring it online, but it’s more likely there will be some interruption while the processing transitions from the old environment to the new one.

In contrast, horizontal scalability (i.e., scaling out)—where you deploy the system on additional resources (e.g., more servers with fewer processors and less RAM) to automatically engage, thus meeting demand at a given time—allows the system to continue running without interruption while these resources are provisioned. Another upside is that horizontal scaling enables you to increase the points of failure for critical applications, providing automatic failover capabilities in case of downtime. Not the least important, scaling out is usually easier to upgrade and cheaper than vertical scaling, and it allows you to scale infinitely.

If you’re like many developers, you can’t predict 100 percent what your load will look like from one moment to the next because it’s dependent on customer needs that can change frequently. But you need to manage all future increases. At the same time, you don’t want to overpay for computing resources when you don’t need them. When a new product or service is first launched,
you might expect fast growth that comes with a surge in new customers. Over time, such activity may transition into seasonal load, characterized by cyclical periods of heavy load followed by lower load (from one month to the next, or from workweek to weekend or, as in the case of Netflix, from evening to morning). For some tasks, like scheduled jobs or daily reports, you would see an on-and-off load pattern, where at times there isn’t any load at all. In the face of unpredictable or competing scenarios, threshold scaling can help with estimated handling of future load.

**Considerations for implementing threshold scaling**

Most cloud-based platforms provide built-in threshold scaling mechanisms that address common scenarios, but it’s important to note that threshold scaling is not an instant solution. Simply adding resources to a system or running more instances of a process doesn’t guarantee that the performance of the system will improve. Building infrastructure with threshold scaling requires careful assessment and thoughtful configuration according to your service's needs to provide value.

Consider the following points when implementing a threshold scaling strategy:

- **Your system must be designed for horizontal scalability.** When scaling a cloud service or website horizontally, avoid specifying instances to your processes. Instead, uplevel your specifications design solutions that do not specify which instance will run what subprocess.

- **Different business workloads will require different configurations.** Distinct scaling policies for different parts of your application may be required when background tasks run on separate compute instances. For example, if the system offers various levels of service (such as basic and premium), you may need to scale out the compute resources for premium service packages more aggressively than those for basic service packages to meet various levels of SLAs.

- **Set an upper limit.** Limit the maximum number of instances that can be automatically added in your scaling configuration. This is to prevent excessive build-out and place an upper limit on costs associated with potentially running many thousands of instances.

- **Set up for monitoring events.** The threshold scaling mechanism should monitor the scaling process and log the details of each scale event (for example, what triggered it, what resources were added or removed, and when). You can use this information to identify usage patterns, help measure the effectiveness of your strategy, and modify the strategy over the long term while the requirements of the application evolve.
Threshold scaling in Microsoft Azure as a solution

Azure threshold scaling enables you to scale your services dynamically according to a set of easy-to-set rules. Using this feature, you can automatically add and remove instances of Azure Web Apps features and Azure Virtual Machines. You can quickly configure high-level scaling policies in the Azure Management Portal, or for the next level of control, use the Azure Monitoring Services Management Library to configure more granular threshold scaling rules.

There are two approaches for configuring automatic scale rules in Azure: 1) metrics based and 2) time based.

With the metrics-based approach, configure threshold scaling according to runtime metrics such as average CPU utilization over, for example, the last hour, or the backlog of items in a message queue that the solution is processing. You configure the parameters, monitor the performance of your system, and if necessary, adjust the way in which the system scales. Keep in mind, though, that threshold scaling is not an instantaneous process; it takes time to react to a metric such as average CPU utilization exceeding or dropping below a specified level. An advantage of Azure is that it keeps you from setting finely balanced thresholds that could attempt to start and stop instances too frequently by permitting only one scaling action to occur in a five-minute (or longer) period. You can increase this period if you find that the system is still overreacting.

Configuring the second method, time-based threshold scaling, ensures that additional instances are available to coincide with an expected peak in usage, and will scale in once the peak time has passed. This strategy enables you to have sufficient instances already running without waiting for the system to react to the load.

Threshold scaling could be a great value to your organization by helping you achieve high levels of availability and resource optimization. When you are ready to get started, Microsoft Azure offers plenty of resources for developers to help you decide if threshold scaling can work for you in addition to multiple approaches for threshold scaling configuration. Thoughtful configuration and careful implementation are essential if you decide to use threshold scaling for your infrastructure.
Chapter Five

OUTWARD EXPANSION: POWERING SPEED AND AGILITY WITH MICROSERVICES.

Power enterprise applications and gain agility and resilience at scale with a microservices architecture.
Imagine you are on a starship boasting the most advanced 23rd-century technology. Your imaginary starship is impressive, not in the least because it integrates advancements in warp power technology and science instrumentation. One of the most technologically sophisticated spacecrafts of its time, your starship is also known to be one of the mightiest, capable of rendering a planet lifeless in a matter of hours. That’s a lot of firepower.

In your real-life universe, you want a high-control starship like this—one that provides foundational, battle-hardened technology for running core infrastructure while powering mission-critical cloud services. You want a platform that takes care of all the superhard rocket science behind the scenes so that you and your team can focus on doing what you do best: building applications with a high degree of customization and scalability.
Moving away from the monolith

The cloud has changed how enterprise delivers services by providing the ability to scale on demand. To benefit from one of the biggest advantages of cloud computing, developers have been motivated to reconsider application design, replacing traditional complex enterprise applications with a microservices approach. That means developing a single application as a suite of small, autonomous, and scalable services that provide easy-to-use APIs for a particular business function. Container technology such as Docker delivers this capability and fulfills on the new cloud technology’s promise to provide process-level abstraction (unlike conventional virtual machines), and the agility that organizations have come to expect from IT.

You’re probably familiar with the common design pattern of a line-of-business application: a web server running user interface code, a monolithic application running all business logic and services, and a single database for all persistent storage requirements. If you recognize this architecture, then you’re probably also aware of its shortcomings: specifically, a change made to a small part of the application requires the entire monolith to be rebuilt and deployed.

Internet-scale companies known for their excellent ability to operate high-volume, high-velocity websites have elected to adopt a more modular and loosely coupled approach based on a microservices architecture, effectively decoupling the application into smaller functional pieces. This produces a fine-grained and self-contained, stateless microservice that is an ideal match for the cloud. Ideal because containerization lets you easily update, add, replace, or remove services from your app with minimal effect on other features or functionality. And Docker containers are a perfect vehicle for this journey. Why? Because they provide effective packaging, portability across environments, high application density for existing resources, and a thriving ecosystem in the Docker Hub.

For example, an e-commerce application could pull multiple Docker containers for the search, caching, messaging, and data back ends, grouping them in clusters that scale horizontally and splitting the application functionally in containers. Containers are organized at the service level: one to retrieve product information from the catalog, another to check inventory, and another to submit a shipping order to a third-
party service. Each of these containers can be described in a Docker file as if they were code— and replicated across environments. Needless to say, this not only speeds up testing and deploying cloned instances, but also provides teams with a safety net because it is connected to threshold scale capabilities. If load suddenly increases, more stateless web servers could be added. Or if an existing stateless service dies, it could be replaced with another. The advantage, when done well, provides increased agility and resilience.

Azure’s cloud platform offers broad support for Docker and the Docker ecosystem.

Of course, this solution presents some challenges. Today, the container-based architectures powering web and mobile apps are typically stateless. But there are many systems where state—that is, persistence or storage—is simply unavoidable. Any system that saves data—and there are many, including not only databases but user accounts, shopping carts, and job queues—has to manage state. As a result, stateful tools for application design continue to increase and evolve. MongoDB, RabbitMQ, Redis, Apache Cassandra, MySQL ... the number and variety of data services that developers can use today for different parts of their application in a microservices architecture goes on and on. Elasticsearch for millions of log messages to parse. Redis for job queues. MySQL for customer sign-ups. All this can happen in the same application. In most cases, though, developers use containers only for the stateless parts of the application, while any stateful services, like databases, are managed outside of the normal application life cycle. This approach ultimately limits the benefits to be gained from fully embracing container-based microservices for the entire application. In the age of distributed teams and locale-specific services, containerizing a whole application provides compelling benefits, such as speeding time to new markets or reducing migration downtime. For example, Docker will greatly expedite datacenter migrations for your organization. Docker allows you to move an entire containerized application easily between datacenters—paying off on the hybrid and portability promises of the cloud. Because it’s traditionally been so challenging to move large data sets long distances, the ability to do so without re-architecting the application as part of the migration is a boon to development teams.

Deployment the way you need it

When your organization leverages containers in your application architecture, you gain the ability to decompose that application functionally, to scale horizontally (for stateless services), and to easily port your application across environments. This leads to an increased container footprint, but alongside it, a challenge to have proper tooling and processes to manage them. These are cloud
service management tools that you would expect to find all under one vendor.

Today's commercial cloud platforms give developers pay-as-you-go access to compute power and storage along with easy access to a suite of common application services, such as relational and NoSQL databases, in-memory cache, and performance analytics. Cloud services make it easy to benefit from automated development, test, staging, and production environments, providing the foundation for continuous delivery. But not all commercial cloud solutions are made equal when managing applications built in a multitude of languages and environments—and open source tools.

Azure's cloud platform, for one, offers broad support for Docker and the Docker ecosystem. It starts with the Docker Extension for Linux Virtual Machines (which also works with tools like Machine, Compose, and Swarm) and includes specialized Linux distributions (for containers like CoreOS, management and monitoring offered by the Operations Management Suite). It then adds on the Azure Container Service (available in private preview) to enable the full range of containerization possibilities. In this way, Microsoft lets developers and organizations focus on delivering great application experiences for their customers, with Azure running the minutiae of efficiency-boosting containers.

There once was a time when developers lacked the foundation for dealing with all the common aspects of developing, running, and managing services at scale. Today, a platform exists that intrinsically understands the available infrastructure resources and needs of applications. Whether providing application life-cycle management capabilities so developers don’t have to re-architect applications while usage grows; solving hard distributed system problems, such as state management; or providing the benefits of orchestration and automation for microservices with a new level of app awareness and insight—Azure cloud allows you and your development team to refocus on developing the functionality that actually matters to you.
All roads lead back to the starship

Building an enterprise-worthy developer environment is a lot like commanding a starship. Your success—and that of your crew—hinges on your ability to architect for the cloud of tomorrow.

• Is your cloud platform equipped to power productivity and innovation?
• Does it support the open source technologies used by your team?
• Will it allow your organization to maintain its on-premises investment?

When it comes to adopting cloud computing, Microsoft believes you shouldn’t have to make compromises—on ease of use, feature sets, hybrid cloud deployment, or portability. Azure is built for openness with the strength of enterprise-level monitoring and support; it is directly integrated with industry standard open source solutions, some of which have been used by open source practitioners for decades.

With Microsoft Azure, you get the best of the cloud, on your terms, without compromising your team—or your capabilities.

Try Azure today, for free, and find out for yourself.

Learn more about Azure open source solutions.
https://azure.microsoft.com/en-us/overview/choose-azure-opensource/

Get up and running with Azure

Architects and developers of all walks and technologies can deploy cloud-based applications on the Microsoft Azure platform. Watch a real project take shape in this Azure OSS webinar.

URL: tbd