EXECUTIVE SUMMARY

Digital disruption is occurring across many industries as technology improvements, ranging from mobile and social to Internet of Things (IoT), are shifting customer behavior and market context. Enterprises mastering digital business technologies are positioned to take advantage of this trend, while poorly equipped incumbents are left behind. This is so widely recognized that achieving a faster pace of innovation is commonly considered a mission-critical investment.

Achieving the goals of the investment depends on an organization’s ability to rapidly:

- **Iterate in application development** – continuously delivering improvements,
- **Extract value from data** – focused on immediate decision making and response with “fast data” as well as in-depth analytical investigation with “big data”.

Best practice rapid application development requires the use of tools like Kubernetes, which serves as a container orchestrator supporting cloud-native methods of continuous deployment, DevOps, and microservices architecture for running stateless applications such as web and application servers. To maximize value from an organization’s data in determining insights and actions requires another set of tools from a diverse, evolving set of stateful (i.e. data store) applications like Spark, Cassandra, and Kafka.

Open source tools and cloud computing are a great start, but their rate of change and complexity to operate (i.e. upgrade, scale, etc.) are challenging for most organizations to embrace. Organizations commonly seek a platform integrating a set of tools into a solution. The struggle has been to find a platform that addresses ease of running both cloud-native, stateless applications and data-intensive, stateful applications.

Mesosphere DC/OS offers enterprises a cohesive platform combining these disparate tools with automated best practices to ease the work of building, deploying, and
operating modern applications. With the recent addition of Kubernetes for container orchestration, DC/OS stands out as a platform for managing both stateless and stateful applications across hybrid cloud deployments with optimized infrastructure utilization. This eases adoption of new technologies with reduced risk and delivers considerable flexibility to use multiple cloud infrastructures cost-efficiently.

CUSTOMER CHALLENGES

As an enterprise learns from its data and the market, the goal is to turn insights into improved offer value as quickly as possible by accelerating pace of innovation and/or reducing time to market. In an application development context, the organization should be able to iteratively move from concept to deployed update within hours, not days or weeks.

Operating Orchestration of Stateless Applications

To reliably reach the desired level of agility involves achievement of a cloud-native orientation in stateless application development by applying three best practice methodologies.

First, application teams need to combine development and operations work in a DevOps approach across the application lifecycle to eliminate the coordination and delay tax of cross-organization handoff required for deployment of application updates. The application teams own timing of application deployments.

Second, teams must structure their development, test, and deployment with an expectation of continuous application update delivery in a Continuous Integration / Continuous Delivery (CI/CD) orientation. Whether “continuous” is every few hours or every few weeks – as determined by business need – the application teams need to adjust culturally and technologically with a strong toolchain to be capable of delivering repeated application improvements without delay.

Third, teams need to be empowered to continuously deliver without cross-application interdependencies that stall delivery. A microservices application architecture enables independence by dividing application responsibilities into smaller stateless components that are easier to upgrade, restart, and scale than traditional monolithic applications.
Containers have become a critical technology to ease implementation of these three methods, particularly with regard to Docker containers improving development and operations collaboration in DevOps, as well as the speed of test and deployment in CI/CD. Managing a set of containers operating per application is handled via a container orchestrator, which addresses deployment, availability, and updating of the microservices. This includes the benefit of optimized infrastructure utilization through more fine-grained scheduling, or placement, of application containers to resources than with virtual machines (VMs). The challenge for the operator of these stateless applications lies in management of the deployment, availability, and updating of the container orchestration system itself.

Operating Stateful Applications

The enterprise’s need to rapidly extract insights from its data to guide application development is rooted in recognition that “data is the new oil”. This popularized assertion calls attention to the latent value of typically difficult to access and analyze enterprise data, and how critical its ready availability and use is to digital leadership across industries.

Enterprises have made significant investments during the last decade in a “big data” orientation to more comprehensively capture and compile data from internal and external sources, providing much larger data sets than amassed previously. Big data enabled analytics for insights and development of product / service offerings was not previously possible. A natural outgrowth of big data was recognition of the opportunity to move beyond after-the-fact analytics into streaming data ingestion for real-time insights and/or automated action on the fire hose of “fast data”.

Here are a few examples… Big data capture is important to enable training of machine learning models, while fast data capture unlocks the value of machine learning for applying AI in applications using streaming data. The proliferation of IoT sensors across industries from manufacturing to transportation is generating data from which immediate insights and action is not only valuable for improved operations, but also can enable new business models and offerings. In infrastructure management, the aggregation of monitoring and logging data enables immediate infrastructure incident or security threat identification with the potential for automated mitigation.

Taking advantage of big data and the increasing business need for fast data presents a challenge in managing an evolving set of technologies. First, streaming data ingestion
from a growing set of sources requires an elastic, high availability (HA), and secure data platform. There are common applications for capturing and storing the data via “SMACK stack” tools, including Spark, Cassandra, and Kafka. Cloud infrastructure allows for highly available, secure implementation with elastic scaling. The challenge is in deploying and operating these stateful applications as data services on the infrastructure. Each tool has its own best practices in how it is deployed, monitored, scaled, and updated without downtime. Ease of enterprise operation of these tools is critical because most are distributed systems built by web-scale companies with in-house expertise beyond what general enterprise technology teams can afford in time and cost to staff (e.g. Apache Hadoop at Yahoo, Apache Cassandra at Facebook, Apache Kafka at LinkedIn, etc.). The set of tools in use continues to evolve quickly with operators asked to add new tools. Rather than staffing the expertise necessary to integrate and operate each of these tools, enterprises need a platform that eases adoption of these new technologies and their integrated operation.

Using a Platform to Operate Stateless and Stateful Applications

When looking across enterprise needs for operating stateless microservices and stateful data services, there is a set of common operating needs:

- **Orchestrated application management** delivering custom deployment and operation, including high availability through updates and infrastructure failures,
- **Security policy enforcement** via identity and access management, container isolation, secrets management, etc.,
- **High availability control system** underlying the orchestration and security policy enforcement systems, including infrastructure fault tolerance and non-disruptive upgrades,
- **Optimized utilization of infrastructure** through shared scheduling of application containers across all datacenter level resources (i.e. at a region level in cloud across availability constructs / zones), not just the app or container cluster level,
- **Hybrid cloud deployment** allowing use of multiple cloud infrastructure platforms, including public and private, without ties to proprietary cloud application programming interfaces (APIs).

A platform providing these capabilities enables an organization to easily adopt and operate new stateful and stateless technologies with secure, cost-efficient resource use across multiple cloud infrastructures.
Many cloud adopters are aware of these needs and see the benefits, but choose to bypass the last item of hybrid cloud deployment capability in order to speed their initial proof of concept (PoC) delivery and/or time-to-market. Instead, they select a single cloud infrastructure provider and leverage the cloud provider's portfolio of platform/application level services. These services present a proprietary API—often based on open source applications—that provides custom orchestration, security policy enforcement, and optimized infrastructure utilization managed by an HA control system. Unfortunately, any short-term benefit achieved comes with a lock-in to proprietary APIs that constrains future flexibility to benefit from new capabilities and/or cost optimization opportunities available on other cloud infrastructure platforms. This approach cedes more control in capability and cost, as well as availability, to a single infrastructure provider than is optimal for most enterprise applications.

**DC/OS: A SINGLE PLATFORM FOR STATELESS AND STATEFUL APPS**

Mesosphere DC/OS is a particularly interesting open source option to consider. Unlike most cloud platforms, it started with strength in managing stateful applications at scale and has grown into common use in management of stateless applications via container orchestration.

At the core of DC/OS is Apache Mesos, an open source distributed system kernel that abstracts VM and bare metal compute resources (e.g. central processing unit (CPU), graphics processing unit (GPU), memory, storage) to enable ease of elastic, high availability operation of distributed system applications like the common big data, fast data, and container orchestration tools. It is designed with a flexible, dual-level architecture that allows the use of frameworks that encapsulate application-specific operational logic to enable the native capabilities of stateful applications and container orchestrators for stateless applications. These frameworks enable application-aware scheduling (i.e. placement) across available infrastructure resources optimal for each application's architecture as shown in Figure 1.
All best practices are turned into code in the framework for deployment, scaling, tuning, updating, and recovery from infrastructure failures. This delivers a strong set of benefits, including:

- Optimizes application availability through scaling events, updates, and failures
  - Ex: Netflix’s projects managing stream processing, microservices, and machine learning pipeline applications,
- Automates many common operational tasks, reducing staffing needs and freeing up more resources to focus on application development

Source: Mesosphere

Figure 1: Benefits of Apache Mesos
Ex: Yelp’s fleet management of Amazon Web Services (AWS) EC2 Spot Instances,

- Reduces the enterprise operations need to develop expertise across each new application
  - Ex: Uber’s multi-datacenter platform for Cassandra grows to include Kafka and their custom sharded version of MySQL,
- Provides an evergreen orientation to enabling all the native benefits from the latest technologies (e.g. with Kubernetes)
  - Ex: Adobe’s use of OpenWhisk for serverless application development,
- Decouples the application scheduling from infrastructure resource management, which enables platform scalability from 10s to 10,000s of nodes
  - Ex: Apple’s Siri runs on a custom Mesos framework supporting applications on 1,000s of nodes with a Hadoop file system (HDFS) backend,
- Decouples the application from the infrastructure via the abstraction of resources, which allows hybrid cloud portability
  - Ex: Royal Caribbean’s platform developed with Ernst & Young for mobile applications across land on AWS and cruise ships on VMware vSphere,
- Maximizes infrastructure utilization by scheduling multiple applications, especially with different workload characteristics, onto shared server resources while maintaining resource guarantees and isolation in a way of running like AWS (i.e. a public cloud provider) in maximizing infrastructure utilization
  - Ex: Autodesk’s Cloud Platform and NBC-Universal’s Audience Studio Platform each pooled similar event streaming and data ingestion workloads into a single DC/OS resource pool that increased utilization enough to cut their AWS bills by >50 percent.

Most importantly for organizational speed and agility in use of data and iterative application development, the combination of these benefits enables faster adoption of new technologies with reduced risk.

**DC/OS Adds Kubernetes Support**

Mesosphere has long invested in multiple container orchestrators, including its early participation in the Kubernetes project, and as a founding, platinum member of the Cloud Native Computing Foundation (CNCF). As the leader of the DC/OS open source project for its platform offering, Mesosphere is adding support for Kubernetes to DC/OS
in view of Kubernetes having achieved maturity for broad enterprise use as well as
becoming the most broadly adopted container orchestrator. The addition of Kubernetes
is achieved using a Mesos framework as a container orchestration option alongside the
existing DC/OS container orchestration option, Marathon, as shown in Figure 2.

**Figure 2: Kubernetes with Mesos Framework**

This framework will provide native Kubernetes support with full functionality exposed
and direct use of the Kubernetes cluster interface once deployed. The addition offers
organizations using DC/OS comfort in extending its use to encompass providing
Kubernetes to their application teams. It offers organizations considering adoption of a
platform the option of DC/OS for use across their stateful applications and their cloud-
native, stateless applications operated with Kubernetes. Mesosphere released this
support into Beta in September 2017 with the intent of introducing features for
Kubernetes use based on existing DC/OS capabilities for General Availability by early
2018.
DC/OS Versus Alternative Platforms for Operating Kubernetes

There is an argument to be made, given the ecosystem strength and market traction of Kubernetes, that organizations should consider the reverse of this approach and adopt Kubernetes itself for managing stateful and stateless applications. Though Kubernetes’ strength has been in the orchestration of stateless Docker containers, feature work has been ongoing for the past year to better support stateful applications, and vendors with supported Kubernetes offerings are now pressing this use case. There are two factors to consider with this alternative.

First, this is less a choice between DC/OS with Mesos and Kubernetes, and more a choice of vendor-supported platforms. The vast majority of enterprises need vendor support for any platform they use to manage their applications, because they lack the internal expertise necessary to build and manage open source Kubernetes or DC/OS on their own. It is also not strategic to their business to build that expertise, which means they will need a platform from a vendor. Therefore, the choice is between Mesosphere DC/OS for running Kubernetes (alongside other applications on Mesos) versus alternative vendor Kubernetes-based platforms.

Second, the more immediate an organization’s need is for a platform for managing stateful applications, the better an option Mesosphere DC/OS appears to be, versus waiting for maturity in Kubernetes-based alternatives. DC/OS is a proven enterprise platform based on more than five years of high-scale enterprise Mesos use in managing stateful applications. Mesos has been proven out through years of operational experience in some of the highest scale use cases in the world. It is unclear as yet how flexible the Kubernetes architecture will be in its ability to provide the same capabilities and benefits, and in what timeframe it would be proven out in high-scale enterprise use. To the degree that Kubernetes emerges from its starting focus on stateless applications to become a good option for stateful applications, adopters of DC/OS can further leverage Kubernetes running on DC/OS to manage more applications versus through their application-specific Mesos framework.

In the meantime, past and present comparisons of market adoption of DC/OS and Mesos to container orchestrators like Kubernetes, implying direct overlap and competition, can represent a misleading oversimplification of the cloud platform technology landscape. DC/OS uses Mesos to provide a platform to serve varied application frameworks. Stateless applications operated using a container orchestration tool like Kubernetes are one type.
The one direct overlap is Marathon as a container orchestration alternative to Kubernetes in DC/OS. Marathon was developed and offered for DC/OS before Kubernetes was created as a project, so Marathon has a stronger operational use history at very high scale (up through 10,000s of nodes) towards which Kubernetes is still building. Given the ecosystem traction and investment in Kubernetes, it has already surpassed Marathon in features and is fully expected to achieve its scalability for stateless application orchestration over time, which is why Mesosphere was an early participant in the Kubernetes project and is now offering it as an alternative to Marathon for DC/OS users. There can be a steeper learning curve to Kubernetes compared to Marathon given the underlying primitives Kubernetes exposes for user configurability, which may be unnecessary for some users not looking for that level of control and customization. This variation in simplicity, as well as support for running traditional applications not yet running in Docker containers, are reasons why some current use of Marathon is expected to continue and why Mesosphere intends to continue supporting it as part of DC/OS.

DC/OS Platform Value Built on Mesos
DC/OS is Mesosphere’s platform built on Mesos. The core features of DC/OS have been available from the DC/OS project since Mesosphere began offering it as open source in April 2016. These features include:

- Infrastructure resource management across:
  - Compute: GPU-based scheduling
  - Networking: Virtual overlay network with an internet protocol (IP) per container and Container Network Interface support
  - Storage: Persistent external volumes,
- Features for optimizing application deployment and availability, including:
  - Blue-green and canary deployment
  - Automated self-healing from infrastructure failures
  - Application-level logging, metrics, and debugging,
- Ease of platform deployment and operation via:
  - Guided installation specific to each public and private cloud infrastructure
  - Unified command-line interface (CLI) and graphical user interface (GUI) across cloud infrastructure deployments
  - Non-disruptive upgrades
  - Service discovery and health monitoring,
• The DC/OS Service Catalog (formerly “Universe”) of over 100 services developed by DC/OS project community, commercial partners, and Mesosphere.

Mesosphere provides this in its supported DC/OS Enterprise offering with emergency patching, a distributed L4/L7 load balancer, and the following features for enterprise security / compliance needs:

• Multi-user identity and access management with Active Directory / Lightweight Directory Access Protocol (LDAP) integration,
• Role-based Access Control for containers, jobs, and data services,
• Service accounts and fine-grained access control list for containers and frameworks for multi-tenant operation,
• Fine-grained access control to logging, metrics and debugging APIs,
• Security audit logging,
• Secrets management (key/value and file-based),
• Public key infrastructure integration.

This full set of features in the Mesosphere DC/OS Enterprise offering eases adoption across multiple application team use cases within an enterprise. Verizon upgraded to DC/OS after starting with Mesos as its target set of workloads and their platform capability needs grew, and they have invested in further advanced networking for high-scale container use on the platform. In Royal Caribbean’s case, it adopted Mesosphere DC/OS as “proven functionality” with “ease-of-use, enterprise-grade 24x7x365 support, and leading resource utilization.”

CALL TO ACTION

Enterprise IT Infrastructure and Operations (I&O) and Application Development leaders face significant operational challenges in enabling their organizations to rapidly iterate in application development and extract value from their data. First, they have maturing options for container orchestration for stateless applications to be managed with cloud-native development and operations methodologies in DevOps, CI/CD, and microservices architecture — at least some of which are typically new to their organizations. Second, they are presented with an evolving list of data-intensive, stateful application technologies to be managed as services on which they normally lack team expertise.
This is driving common consideration of adopting a platform that will address these challenges while:

- Allowing developers to forget about the ‘plumbing’ and spend more time creating business value, instead of managing the container orchestration systems and data services on which they depend,
- Reducing operations resources needed for deploying and managing availability of the applications,
- Enabling use across cloud infrastructures to avoid lock-in and enable hybrid cloud deployment,
- Maximizing resource utilization for cost efficiency to address ever-growing infrastructure bills that are inconsistent with pressures I&O leaders often face to reduce costs.

Moor Insights & Strategy recommends, first, that when building towards a portfolio of data services based on an evolving set of stateful applications, I&O and Application Development leaders consider Mesosphere DC/OS as a platform to efficiently operate these services for the organization. DC/OS has a strong presence and track record for operating stateful applications, and alternatives need time to emerge, including Kubernetes if it matures for this use case.

Second, given Mesosphere’s announcement of full Kubernetes support in DC/OS, consider use of DC/OS as a broader platform for managing stateless applications, including intended use of Kubernetes as a preferred container orchestration system. I&O and Application Development professionals should consider the tradeoffs between platforms optimized for stateless applications with focus on Kubernetes versus DC/OS. More specifically:

- Consider tradeoffs in organizational need for support specific to Kubernetes from CoreOS, Heptio, etc. as specialists focused on Kubernetes cluster management – versus – Mesosphere DC/OS supporting Kubernetes leveraging its broader existing platform capabilities also applied to stateful applications like Cassandra, Kafka, and Spark,
- Consider tradeoffs in organizational need for a broader application development platform capability set from Pivotal, Red Hat, etc. on top of Kubernetes – versus – the need to ease into Kubernetes use common in early organic enterprise adoption before making a decision on broader capability needs.
If considering separating Kubernetes-based platform use for stateless applications from use of DC/OS as a platform for stateful applications, also consider the tradeoffs in benefits from the Kubernetes-focused platform, versus consolidating on DC/OS for the cost efficiency benefit of maximizing infrastructure resource utilization across workloads and simplified operation of a single platform overall. The more apps that are run on DC/OS, particularly across stateful and stateless applications that typically have very different resource consumption profiles, the greater the benefits in utilization and cost efficiency.
IMPORTANT INFORMATION ABOUT THIS PAPER

CONTRIBUTOR
Rhett Dillingham, Senior Analyst In-Residence at Moor Insights & Strategy

INQUIRIES
Contact us if you would like to discuss this report, and Moor Insights & Strategy will respond promptly.

CITATIONS
This paper can be cited by accredited press and analysts but must be cited in-context, displaying author's name, author's title, and "Moor Insights & Strategy". Non-press and non-analysts must receive prior written permission by Moor Insights & Strategy for any citations.

LICENSING
This document, including any supporting materials, is owned by Moor Insights & Strategy. This publication may not be reproduced, distributed, or shared in any form without Moor Insights & Strategy's prior written permission.

DISCLOSURES
This paper was commissioned by Mesosphere. Moor Insights & Strategy provides research, analysis, advising, and consulting to many high-tech companies mentioned in this paper. No employees at the firm hold any equity positions with any companies cited in this document.

DISCLAIMER
The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. Moor Insights & Strategy disclaims all warranties as to the accuracy, completeness, or adequacy of such information and shall have no liability for errors, omissions, or inadequacies in such information. This document consists of the opinions of Moor Insights & Strategy and should not be construed as statements of fact. The opinions expressed herein are subject to change without notice.

Moor Insights & Strategy provides forecasts and forward-looking statements as directional indicators and not as precise predictions of future events. While our forecasts and forward-looking statements represent our current judgment on what the future holds, they are subject to risks and uncertainties that could cause actual results to differ materially. You are cautioned not to place undue reliance on these forecasts and forward-looking statements, which reflect our opinions only as of the date of publication for this document. Please keep in mind that we are not obligating ourselves to revise or publicly release the results of any revision to these forecasts and forward-looking statements in light of new information or future events.

©2017 Moor Insights & Strategy. Company and product names are used for informational purposes only and may be trademarks of their respective owners.