The State of Latency, Containers, & Microservices

Survey Methodologies & Results
The State of Latency, Containers, & Microservices

Methodology

**Purpose**

VMTurbo conducted this survey to better understand the role, measurement, and mitigation of latency in the modern datacenter. The investigation extended into the implementation of containers and microservices, as these architectures introduce new latency challenges. Our hope is that the results will instigate a data-driven conversation across the broader virtual and cloud community.

**Sample**

The data in this report were collected through a survey conducted from July 22, 2015 to August 3, 2015. The 554 survey respondents came from across the Enterprise IT and data center landscape. Respondents are of 18 years of age and older. In order to reveal the range in characteristics, respondents were identified demographically by their business and environment characteristics, such as role, business type, hosts in production, and virtual machines in production. This sample represents organizations spanning SMB to large enterprise, with various roles and responsibilities in those organizations.

**Procedure**

This survey recruited participants from an internal email database. Participants were given an opportunity to win a pass to a major industry conference by entering their email address at the completion of the survey. Additionally, participants were given the option to participate in a one-on-one interview subsequent to completing the survey. While the survey successfully recruited a significant sample size, the distribution of the sample weighs highly in Operations Management as a role and was well-distributed across business types. Data were collected electronically through an online survey. The survey itself was designed internally by a team, which included microservices subject matter experts, namely VMTurbo engineers.

**Survey Flow**

The data in this survey report were collected within a twelve day period. Progression through the twenty-nine survey questions depended on a respondent's level of interest in containers and microservices and and the current status of adoption. All respondents were asked the same questions about latency and its role, measurement, and mitigation within their organization. If a respondent was investigating containers or microservices as a current or future application delivery architecture, they were then asked about expected deployment time, perceived challenges and benefits, industry influence, as well as running these architectures in production or non-production environments.

**Citing this Survey**

We welcome your use of the results in this survey as you share insights with members of the broader IT community. Please reference VMTurbo and include our homepage URL, vmturbo.com, as you do so. A downloadable version of the complete dataset is available at https://github.com/vmturbo/VMTurboSurvey. Thank you.
In July 2015, VMTurbo conducted an industry survey titled *How Are You Fighting Latency?* The survey aimed to explore trends among three related themes:

1. **Latency** - The methods organizations used to mitigate latency, and the designation of latency-critical workloads across industry verticals.
2. **Container adoption** - Penetration, emerging reasons for container adoption, and emerging barriers to container adoption.
3. **Microservices adoption** - Penetration, emerging reasons for microservices adoption, and emerging barriers to microservices adoption.

Several high-level findings included the following:

**Latency Findings**

- 91% of participants agreed or strongly agreed that the minimization of latency is important to their company, with 23% of participants identifying a clear majority of hosted workloads (between 60 percent and 100 percent of workloads) as being Latency-Critical.

- Although most participants are in agreement that the minimization of latency is important to their company, a full 32% of participants either do not measure latency or do not know if their company measures latency.

- There exists a disconnect between the tactics organizations currently leverage to minimize latency, and those which they deem would be most effective at minimizing latency.

**Container Findings**

- Just 4% of participants have actually deployed containers.

- The greatest challenge in managing latency within container architectures is managing storage latency.

**Microservices Findings**

- The top reason for implementing containers is to accelerate the application development lifecycle.

- Docker remains far and away the top container standard, followed by CoreOS and LXC.

- 11% of participants have microservices-based applications in their production environment, nearly three times the adoption rate of containers, suggesting predominantly VM-based microservices deployments.

- Of those participants without microservices currently in production, 34% will deploy microservices within the next two years and 66% have no intention to deploy microservices.

The following analysis addresses these findings, their implications, and enables organizations to benchmark where they fall within the participant mix.
Results

Demographics

HIGHLIGHTS

- 46% of respondents are in an Operations Management role

- Industry verticals were distributed with majority concentrations in Professional Services, MSP, Manufacturing, and Financial Services

Role

- Operations Management: 46.2%
- IT Director: 16.8%
- Network Management: 14.1%
- Storage Management: 10.6%
- Consultant: 7.2%
- CIO/CTO: 5.2%

N = 554

Organization Type

- Professional Services: 17.0%
- MSP/Cloud Provider: 14.1%
- Manufacturing & Industrial: 13.7%
- Financial Services & Insurance: 13.2%
- Healthcare: 10.7%
- Education & Nonprofit: 9.6%
- Retail & Consumer Products: 9.4%
- Public Sector: 6.9%
- Other: 3.8%
- Energy & Utilities: 1.8%

N = 554
HIGHLIGHTS

- 89% of participants operate a private or hybrid cloud environment
- 53% of participants have an annual IT budget of less than $500,000 USD

![Environment Configuration Chart]

<table>
<thead>
<tr>
<th>Environment</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Private Cloud</td>
<td>56%</td>
</tr>
<tr>
<td>Hybrid Cloud</td>
<td>33%</td>
</tr>
<tr>
<td>Public Cloud</td>
<td>4%</td>
</tr>
<tr>
<td>Other</td>
<td>4%</td>
</tr>
</tbody>
</table>

![Annual IT Budget Chart]

<table>
<thead>
<tr>
<th>Budget Range</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0-99,999</td>
<td>27%</td>
</tr>
<tr>
<td>$100,000-499,999</td>
<td>16%</td>
</tr>
<tr>
<td>$500,000-999,999</td>
<td>16%</td>
</tr>
<tr>
<td>$1M-5M</td>
<td>16%</td>
</tr>
<tr>
<td>&gt; $5M</td>
<td>25%</td>
</tr>
</tbody>
</table>

N = 554
Results

Demographics

HIGHLIGHTS

- 61% of respondents have an environment with at least 25 hosts and 500 VMs
- 3% of respondents have environments with more than 1,000 hosts and 20,000 VMs

### Hosts in Environment

<table>
<thead>
<tr>
<th>Hosts (%)</th>
<th>n = 554</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 25</td>
<td>39.0</td>
</tr>
<tr>
<td>25-49</td>
<td>7.4</td>
</tr>
<tr>
<td>50-99</td>
<td>3.4</td>
</tr>
<tr>
<td>100-249</td>
<td>5.1</td>
</tr>
<tr>
<td>250-1,000</td>
<td>4.3</td>
</tr>
<tr>
<td>&gt; 1,000</td>
<td>2.3</td>
</tr>
</tbody>
</table>

### VMs in Environment

<table>
<thead>
<tr>
<th>VMs (%)</th>
<th>n = 554</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 500</td>
<td>39.0</td>
</tr>
<tr>
<td>500-999</td>
<td>1.4</td>
</tr>
<tr>
<td>1,000-4,999</td>
<td>0.2</td>
</tr>
<tr>
<td>5,000-9,999</td>
<td>0.0</td>
</tr>
<tr>
<td>10,000-19,999</td>
<td>0.0</td>
</tr>
<tr>
<td>&gt; 20,000</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Latency is the time interval between a stimulation and response, or, from a more general point of view, is a time delay between the cause and the effect of some physical change in the system being observed. In computing, latency is a physical constraint determined by the distance between networked components, their physical transmission limits and the manner in which software interacts with the infrastructure on which it runs.

Although your purview may lead you to associate latency with a given entity, storage or network for example, latency is truly the sum of all operations – including overhead inherent in application code – required to transmit the encoded impulses that constitute a service. Even as advances in compute, storage, and network architectures have reduced latency from minutes to seconds and milliseconds, so too have advances in business raised the expectations and reliance upon increasingly fast application response times.

Amid this shift in business expectations, a shift in software architectures – toward virtualization, containerization, and distributed computing – has introduced a new wave of complexities that threaten the millisecond scale on which we operate, and a new wave of challenges for the humans tasked with assuring this scale.

Our survey sought to understand the prevalence and measurement of latency-critical applications in today’s data centers, the methods used to assure SLAs, and the perceived efficacy of these methods.
Results

Latency: Measurement & Verticals

HIGHLIGHTS

- 32% of participants either don’t measure latency or don’t know if their organization measures latency
- 23% of participants classify a clear majority of workloads as latency-critical
- Top 3 industry verticals for latency-critical workloads were Managed Service Providers, Financial Services & Insurance, and Healthcare

Q7: What proportion of your production workloads are classified as latency-critical?

Q7xQ2: Clear Majority Latency-Critical x Industry Vertical (%)

- MSP/Cloud Provider: 22.2%
- Financial Services & Insurance: 14.8%
- Healthcare: 14.8%
- Manufacturing & Industrial: 12.3%
- Retail & Consumer Products: 11.1%
- Education & Nonprofit: 8.6%
- Professional Services: 8.6%
- Public Sector: 3.7%
- Energy & Utilities: 2.5%
- Other: 1.2%
Results

Latency: Mitigation Tactics

HIGHLIGHTS

- Respondents are predominantly focused on controlling and mitigating latency within the LAN/WAN edge as opposed to endpoints.
- There is a greater focus on mitigating latency across IT domains as opposed to within specific domains.

Q10: Please rate your agreement with each of the following statements as it pertains to your company.

My company focuses on controlling and mitigating latency **within** the LAN/WAN edge.

Factors: Strongly Disagree, Disagree, Agree, Strongly Agree

Q12: Please rate your agreement with each of the following statements as it pertains to your role.

I focus on minimizing latency **only within** my specific IT domain (i.e. only compute, only storage, only network).

Factors: Strongly Disagree, Disagree, Agree, Strongly Agree

I focus on minimizing latency **across** various IT domains (i.e. across compute, storage, and network).

Factors: Strongly Disagree, Disagree, Agree, Strongly Agree

n = 356
Results

Latency: Mitigation Tactics

HIGHLIGHTS

- 58% of respondents use infrastructure monitoring software and manual troubleshooting to mitigate application latency
- 49% of respondents run virtualized workloads on dedicated clusters to mitigate application latency
- For efficacy, infrastructure monitoring software and dedicated clustering were rated 6/13 and 4/13, respectively

Q14: What tactics does your organization use to mitigate application latency (Check all that apply)?

- Use of infrastructure monitoring software (i.e. Dell Foglight, Solarwinds, VMware vRealize Operations) and manual troubleshooting
- Running virtualized workloads on dedicated clusters
- Implementation of Hybrid (auto-tiering Flash & Hard Disk) storage arrays
- Implementation of Fibre Channel storage connectivity
- Implementation of low-latency network components
- Running Latency-Critical applications on dedicated physical infrastructure (i.e. non-virtualized infrastructure)
- Virtualizing but not mixing diverse workloads
- Use of workflow scripting or load balancing to balance resource utilization
- Use of application performance management/monitoring software (i.e. AppDynamics, New Relic, DynaTrace, AppNeta) and manual troubleshooting
- Implementation of All-Flash storage arrays
- Implementation of Software-Defined Networking (SDN) technologies
- Implementation of Network Function Virtualization (NFV) technologies
- Other

Efficacy Rank

- 58.43% #6
- 49.16% #4
- 43.26% #3
- 42.13% #1
- 40.17% #4
- 30.9% #7
- 30.9% #7
- 30.06% #10
- 25.84% #13
- 21.63% #1
- 17.7% #11
- 12.08% #7
- 62% Other

n = 356
**Analysis**

**Latency: Measurement, Mitigation, & Verticals**

**Latency: MSPs, Financials, & Healthcare**

The charts on the previous pages illustrate three noteworthy survey findings. Survey Question 11, “How does your company measure network latency?” was compared against an uncharted question (Question 10) in which 90.7% (n = 356) of respondents Agreed or Strongly Agreed that the avoidance and minimization of latency is important to their company. When this proportion is considered in light of Question 11, a full 32.3% of participants either do not measure latency or do not know if they measure latency. Given the reported importance of latency mitigation in Question 10, it is concluded that participants have either overstated the importance of latency mitigation, or that a sizable portion of respondents take a reactive approach to latency mitigation.

In Question 7, “Approximately what proportion of your production workloads are Latency-Critical (i.e. business needs cannot tolerate high latency levels)?” 22.8% of respondents (n = 356) identified a clear majority (60%-100%) of their workloads as being Latency-Critical. Of these (n = 81), 51.8% fell within the Managed Service Provider, Financial Services & Insurance, and Healthcare verticals.

The scenarios and applications contributing to this effect are rather intuitive. Service Providers, whose customers must contend with the speed of the Internet, cannot afford to harbor latency within their local network impacting service delivery. Financial Services, whose environments often consist of banking and trading applications reliant on ultra-low latency, are a logical fit. As Meaningful Use becomes a healthcare imperative, immediate access and availability of Electronic Medical Records necessitates latency-critical classification in those environments.

**Combatting Latency Across Domains**

IT culture is inextricably bound to the architecture it supports. The days of monolithic client-server architecture were marked by appropriately monolithic teams, each fixed on its domain and its domain alone. As IT has largely transitioned to an era marked by virtualization, cloud, and mobility, the data suggests that IT organizations have adjusted culturally also – to work across silos, reflective of their interdependence.

Our survey asked participants to rate their agreement, on a 4-point scale where 1 = Strongly Disagree and 4 = Strongly Agree, with the following statements:

- I focus on minimizing application latency within my specific IT domain (e.g. only compute, only storage, only network)
- I focus on minimizing application latency across various IT domains (e.g. across compute, storage, and network)

For all participants, there was a majority focused on minimizing application latency across various IT domains (i.e. compute, storage, and network) (83.1%, X = 3.49) as opposed to within their specific domain (71.8%, X = 3.15). When scoped to the 81 participants with a clear majority of Latency-Critical workloads, the mix shifted to 100% (X̄ = 4.00) and 76.2% (X̄ = 3.29), respectively.

At a high level, this data supports the notion that as the imperative of low latency increases, so does the imperative of cross-functional latency mitigation. While additional surveying would be required to explore precisely what this means, it is clear that the old pattern of passing problems down the stack no longer suffices. Our hypothesis is that in addition to cross-domain cooperation, more and more IT professionals have become generalists of necessity, which is to say, no longer is it sufficient to specialize in one limited area. IT professionals must expand their skill sets to span domains and keep with the pace of architectural change. Again, this hypothesis requires additional surveying.

**A Mismatch of Tactics**

A primary objective of this survey was to identify and rank the methods organizations use to mitigate latency. Our findings were rather pronounced across methods, however, the most notable finding was that the most effective tactics are not practiced by a majority of organizations. This
finding is most likely explained by budget, as the tactics deemed most effective also tend to be the most expensive.

Survey Question 14 asked Which tactics does your organization use to mitigate application latency (check all that apply)? and offered the following selections:

- Running Latency-Critical applications on dedicated physical infrastructure (i.e. non-virtualized infrastructure)
- Use of infrastructure monitoring software (i.e. Dell Foglight, Solarwinds, VMware vRealize Operations) and manual troubleshooting
- Use of application performance management/monitoring software (i.e. AppDynamics, New Relic, DynaTrace, AppNeta) and manual troubleshooting
- Use of workflow scripting or load balancing to balance resource utilization
- Running workloads on dedicated clusters
- Virtualizing but not mixing diverse workloads
- Implementation of low-latency network components
- Implementation of Software-Defined Networking (SDN) technologies
- Implementation of Network Function Virtualization (NFV) technologies
- Implementation of All-Flash storage arrays
- Implementation of Hybrid (auto-tiering Flash & Hard Disk) storage arrays
- Implementation of Fibre Channel storage connectivity
- Other (Please Specify)

Survey Question 15 asked For each of the practiced tactics you selected in Question 14, please rate their effectiveness at mitigating application latency within your organization.

The top tactics used to mitigate application latency (N=356):

1. Infrastructure Monitoring Software (58.4%)  
2. Dedicated Clustering (49.2%)  
3. Hybrid Storage (43.3%)  
4. FC Connectivity (42.1%)  
5. Low-Latency Networking (40.2%)

The most effective tactics used to mitigation application latency:

1. Low-Latency Networking ($X = 3.44$)  
2. All-Flash Arrays ($X = 3.42$)  
3. Hybrid Arrays ($X = 3.40$)  
4. FC Connectivity ($X = 3.39$)  
5. Dedicated Clustering ($X = 3.38$)

When scoped to participants with a clear majority of Latency-Critical workloads, the results were thus:

The top tactics used to mitigate application latency (n=81):

1. Infrastructure Monitoring Software (66.7%)  
2. Dedicated Clustering (65.43%)  
3. FC Connectivity (54.3%)  
4. Hybrid Storage (45.7%)  
5. Low-Latency Networking (40.7%)

The most effective tactics used to mitigation application latency:

1. Low-Latency Networking ($X = 3.44$)  
2. All-Flash Arrays ($X = 3.42$)  
3. Hybrid Arrays ($X = 3.40$)  
4. FC Connectivity ($X = 3.39$)  
5. Dedicated Clustering ($X = 3.38$)

Of particular note is that for both groups, the full participant population and the scoped sample, the usage of Infrastructure Monitoring Software was the number one tactic used to mitigate application latency. In both groups, however, this tactic failed to make the top tactics in terms of efficacy. Notable write-ins for Question 14 included the following:

- Automated Workload Migration
- Citrix XenApp + High Latency WAN Links
- Hyperconverged Infrastructure (Nutanix)
- 10 Gb LAN/VLAN + 40 Gb Storage Links
- Nothing
Containers are not new. In fact, their conceptual lineage dates back to the 1970s and chroot jail in UNIX, wherein an application’s processes and dependencies were isolated from the rest of the system.

In 2014, the genesis of Docker catapulted containers back into the spotlight, as a prospective and seemingly inevitable replacement for their heavier cousins, virtual machines. Since Docker made its first headline, numerous production-ready alternatives have surfaced for consideration.

Similar to how virtualization abstracts the operating system away from the hardware, containerization abstracts the application away from the operating system. This concept unlocks a world of new methods for quickly developing, deploying, and delivering applications. Containers are orders of magnitude smaller than VMs, bear a fraction of the memory overhead, and can be provisioned in a matter of seconds instead of minutes.

Our purpose in exploring containers as part of this survey was to anticipate and investigate three related phenomena: (1) The continued adoption of containers in both pre-production and production (2) The driving forces behind this adoption and (3) The emergence of new latency-related challenges in containerized environments.

Q16: Is your organization investigating using containers (i.e. Docker, CoreOS, LXC) for future deployment?

- Yes: 75%
- No: 21%
- We already use them: 4%

n = 354
Results

Containers: Adopters

HIGHLIGHTS

- 25% of respondents have already deployed or are investigating the deployment of containers
- 84% of respondents who are investigating the deployment of containers plan to deploy in 2016 or later
- Docker and CoreOS are the most popular container standards, with 59% and 24% adoption shares, respectively

Q17: When does your company plan on deploying containers?

- 31% Within 6 months
- 16% In 6 months to 1 year
- 16% In 1 year to 2 years
- 38% In more than 2 years

Q18: What container standard will your company adopt?

- 59% Docker
- 24% CoreOS
- 11% LXC
- 6% Other

n = 77

n = 71
HIGHLIGHTS*

- 64% of respondents have deployed or will deploy containers into their production environment.
- 58% of respondents are interested in containers in order to implement DevOps.
- 36% of respondents report Storage Latency as the greatest latency challenge in their container environment, however, 34% report not having any challenges.

Q19: In what environment has your organization deployed (or will deploy) containers?

- Production Environment: 24%
- Non-Production Environment: 64%
- I don't know: 13%

Q21: Which of the following is the greatest challenge in managing container latency?

- Storage Latency: 36.47%
- We do not have any challenges: 34.12%
- East-West Network Latency: 15.29%
- Code-Induced Latency: 8.24%
- Other: 5.88%

Q22: Why is your organization interested in containers?

- Implementing DevOps: 57.65%
- Accelerating Application Development Lifecycle: 48.24%
- Microservice Architecture: 41.18%
- Implementing CD/CI: 20%
- Other: 5.88%
Microservices is an application architecture most succinctly defined as *loosely-coupled service oriented architecture (SOA) with bounded contexts*. The term was first coined in 2005 by Dr. Peter Rogers, CEO of 1060 Research, at CloudComputing Expo during his presentation on Service-Oriented-Development on NetKernel. Specifically, Rodgers coined the term “Micro-Web-Services.”

Microservices, as opposed to monoliths, relate application components as a graph of independent service-evoked functions, rather than a series of persistent and dependent tiers. Microservices deliver numerous benefits, including modularity, full encapsulation, and isolated persistence. Operationally, they enable continuous development/continuous integration, as well as independent testing, deployment, and scalability. When evoked, each microservices component communicates with its siblings laterally over the network, using language-agnostic APIs. When a component fails or requires an update, that component (rather than an entire tier) gets cloned or updated.

Our survey questioned respondents about their experience with microservices for two reasons. First, the rise of containers in production was hypothesized conducive to and/or correlated with the implementation of microservices architecture. Second, one of the greatest criticisms of microservices has been their introduction of network latency.

Critics argue that microservices trade one type of complexity - code complexity - for another - operational complexity. These viewpoints notwithstanding, our survey sought to understand microservices as it exists in the wild, at least as it does in 2015.
Results

Microservices

HIGHLIGHTS

- 11% of respondents have deployed microservices in production
- 66% of respondents currently have no plans to deploy microservices
- 51% of respondents are interested in microservices in order to implement DevOps
- 41% of respondents report Storage Latency as the greatest latency challenge in their microservices environment, however, 31% report not having any challenges

Q24: How would you characterize the applications running (or that will run) using microservices? (Check all that apply)

- Production Backend: 61.54%
- Test/Dev Only: 48.72%
- Production Frontend: 38.46%
- Other: 2.56%

Q25: Which of the following is the greatest challenge in managing microservices latency?

- Storage Latency: 41.03%
- We do not have any challenges: 30.77%
- Code-Induced Latency: 17.95%
- East-West Network Latency: 7.69%
- Other: 2.56%

Q26: Why is your organization interested in microservices (Check all that apply)?

- Implementing DevOps: 51.28%
- Accelerating Application Development Lifecycle: 46.15%
- Implementing CD/CI: 23.08%
- Other: 5.13%
Containers: Still Early, and Not For Everyone

Our findings are consistent with prevailing commentary that despite a great deal of hype and publicity surrounding containers, very few organizations are actually using them. Question 16 asked participants, "Is your organization investigating using containers (i.e. Docker, CoreOS, LXC) for future deployment?" Of all respondents (n = 354), just 25% answered either ‘Yes’ (21%) or ‘We already use them’ (4%). An overwhelming 75% of respondents said No. While the promise and specific use cases of containers are well known, documented, and widely-accepted, the data suggest that at least for now, most organizations find their virtualized or bare metal deployments to be sufficient.

When scoped to the subset of respondents already using containers in their production environment, 42.9% (n = 14) have IT budgets of $5 million or greater and 50% have 250 or more hosts. Compared to the overall participant population, just 16% have budgets of $5 million or greater and 23% have 250 or more hosts. It is evident from the container-adopting sample that organizations with greater spending power are more prone to experiment with next-generation technologies like containerization. One must interpret this data and analysis with caution, as the sample size representative of container adopters is small.

Although at this time a majority of respondents are dismissive of containers, a similar survey in the future is likely to produce different results, especially since 84% of container contemplators plan to deploy in 2016 or beyond.

The momentum of central players like Docker and CoreOS, as well as the significant investments made by industry leaders VMware (Project Photon), Red Hat (Project Atomic), Microsoft (Hyper-V/Windows Server Containers), and others, indicate that container technology is widely regarded as the future. It simply remains in its infancy.

Docker First, and Foremost

59% of container contemplators (n = 71, those who responded ‘Yes’ to Q16) plan to adopt Docker. 71% of container adopters have chosen Docker. Clearly, the benefits of first mover advantage are playing-out for the container leader. Of particular challenge for every standard is differentiation in a category with so little room to differentiate. Containers, by their nature, are simple constructs. CoreOS’ Rocket has positioned itself as a purer, more secure alternative to Docker. Management frameworks such as Docker Hub are one means of differentiation, however, a burgeoning landscape of third party tools dilutes much of this advantage.

"lmctfy” was the only notable mention of non-Docker/CoreOS/LXC container standards under consideration.

Microservices: Mostly VM-Based, For Now

Just 11% of respondents (n = 348) are currently running microservices in production. Given that only 4% of respondents are currently using containers, it can be concluded that a majority of existing microservices deployments are running as virtual machines (VMs).

Similar to the trends observed with containers, a disproportionately high number of respondents that have deployed microservices are from companies with IT budgets of $5 million or greater (x̄ = 32.4%). Microservices, it is said, trade code complexity for operational complexity. A portion of this operational complexity is that discrete services are assigned to many small development teams who develop, refine, and own that service. It follows logically that the organizations most capable of operating in this fashion are those with the budgets to support the requisite developer headcount.

Storage Latency: Emerging Challenge?

While the primary driver for adopting both containers and microservices is Implementing Dev/Ops, we were surprised to find that the top challenge in mitigating both container- and microservices-related latency was fighting storage latency. The top criticism of these architectures is the east-west network latency they generate. We expected our survey to support this sentiment, however, given the low rates of adoption for both, we believe that future surveying when microservices are more mature will yield different challenge rankings.
About VMTurbo

VMTurbo’s Application Performance Control platform enables customers to manage cloud and enterprise virtualization environments to assure application performance while maximizing resource utilization. VMTurbo’s patented technology continuously analyzes application demand and adjusts configuration, resource allocation and workload placement to meet service levels and business goals. With this unique understanding into the dynamic interaction of demand and supply, VMTurbo is the only technology capable of controlling and maintaining an environment in a healthy state.

The VMTurbo platform first launched in August 2010 and now has more than 30,000 users, including many of the world’s leading money center banks, financial institutions, social and e-commerce sites, carriers and service providers. Using VMTurbo, our customers, including JP Morgan Chase, Salesforce.com and Thomson Reuters, ensure that applications get the resources they need to operate reliably, while utilizing their most valuable infrastructure and human resources most efficiently.

© 2015 VMTurbo, Inc. All Rights Reserved. All trademark names are the property of their respective companies.